Design and Implementation of an Online Article Reading System Based on NLP and Crawler

1^{st} tianxiang Zhang
dept. Nanchang University
Nanchang,China

***Abstract*—With the development of the Internet, more and more people choose to get information through the network, which makes a lot of articles presented in the form of webpage text on the Internet. Inevitably, a lot of repetitive text information also appears on the Internet, which leads to a decline in people's reading experience and also leads to the redundancy of storage space on the Internet. Based on the research of natural language processing, this paper proposes a text similarity algorithm based on multi-feature fusion. By deploying this algorithm, the articles in the library can be filtered quickly, and the highly similar articles can be marked, which reduces the possibility of users to browse the same articles twice.**

This paper also develops a set of online articles reading system which applies the text similarity algorithm. The system is based on the browser/server model, the background system is based on the Java language, with the popular SSM framework, and the development environment is Eclipse. The foreground system is based on the Html & vue.js framework, and the development environment are node.js & Microsoft Visual Studio Code. The system can realize the requirements of articles reading, user login, user registration, reading history and user's information change.

***Index Terms*—text similarity; Online article reading; system design**

I. Introduction

With the development of the Internet and the emergence of new media, more and more articles are published online, which greatly reduces the cost of reading for users [1]. However, this has also led to the emergence of a large number of highly similar or even repetitive articles [2]. For example, some websites extract articles directly from high-quality websites to increase traffic and place them on their own pages. There are even some websites, such as CSDN, that allow users to directly upload the content of articles within their own site. As long as the uploader declares the origin of the article on the page and uses search engines to search for keywords, multiple pages from the site will be listed in search results, but the article content is almost the same or even word for word, which also reduces the user's reading experience to a certain extent.

The news industry also has this problem. The same news article may be reprinted by multiple media, which are subscribed to by the same news aggregation platform, resulting in the same article appearing multiple times on the news aggregation platform. This also leads to the phenomenon of duplicate push when aggregating platforms push news to users.

This article designs an online article reading system based on text similarity, based on relevant technologies in natural language processing. This system performs similarity analysis on stored articles, marks and archives highly similar or even duplicate articles, reducing the occurrence of duplicate articles when users retrieve articles. At the same time, it provides users with the ability to quickly browse highly similar articles.

II. Related Works

A. Current research status at home and abroad

At present, there is a lot of research on article similarity in the industry, but it is mostly used for detecting academic misconduct and text classification. Academic misconduct detection will decompose and compare articles in detail, and each part of the article will undergo similarity analysis in the comparison library. The detection results are relatively detailed and accurate [3]. However, this algorithm typically consumes a significant amount of computing resources, consumes a significant amount of time, and is highly challenging to the hardware environment, resulting in high costs. Text classification tends to analyze the correlation between articles and the characteristics of the articles. Text classification commonly uses clustering algorithms to associate articles in a library, in order to achieve the goal of classifying articles with similar features [4]. However, its analysis results are difficult to apply to marking duplicate articles, and additional costs need to be paid for further processing of the results.

In China, in order to ensure the quality of academic papers and articles, many large academic websites have adopted academic misconduct detection systems, such as CNKI, to detect the similarity between articles to be stored and those in inventory (https://www.cnki.net) We use the CNKI academic misconduct detection system maintained by the Tongfang CNKI Research Integrity Management System Research Center, as well as the subordinate academic misconduct detection system for academic dissertations and scientific journals, as well as the Wanfang Data Knowledge Service Platform(http://www.wanfangdata.com.cn)Wanfang data literature similarity testing service used. In addition, some commercial companies also provide similar article originality checks/products, such as PaperPass under Beijing Zhichi Shuhui Technology

Co., Ltd. and Daya Paper Detection System provided by Superstar Daya.

The research on text classification algorithms started relatively early, dating back to the 1960s, when text was classified using artificially defined rules. After the 1990s, with the rapid development of the Internet and the rise of machine learning, researchers began to re study methods for text classification. In 1992, Lewis published a paper titled ”Presentation and Learning in Information Retrieval”, in which the author introduced a text classification system and tested it using his own dataset. In 1995, Vipnik proposed a support vector machine method based on statistical principles [5]. Then, classification models such as TFIDF emerged. In China, the application of text classification relies on a preprocessing technique, namely text preprocessing. After preprocessing, the text is language independent and can be classified using general methods. Due to the issue of Chinese language structure, one of the most important steps in text preprocessing is Chinese word segmentation. In the 1980s, Wang Xiaolong from Harbin Institute of Technology proposed the theory of minimum word segmentation, but this theory has a significant flaw, which is ambiguity. It wasn't until the 1990s that Guo Jin from Tsinghua University solved this problem using statistical language models.

When studying natural language processing, researchers have also discovered the important branch of text summarization. Text abstracts contribute a lot to calculating text similarity. The similarity of the abstract can be used to estimate the similarity of the original text. At this point, researchers were inspired by the Page Rank principle and applied it to the field of text processing, resulting in the Text Rank algorithm. When calculating the similarity of long texts, Text Rank can be an important indicator.

III. Related technologies

A. hanLP

HanLP is a collection of NLP Chinese tools. HanLP project author He Han (Hancs), based on the open source protocol Apache License 2.0, is now managed by Qing Island Dakuai Search Computing Co., Ltd. Management and Maintenance [6]. HanLP supports Java and Python, and can be introduced using the Maven plugin in the Java environment. And the hanLP library is ready to use and does not require configuration, making it convenient for developers to use.

HanLP features include word segmentation and part of speech tagging, dependency parsing, keyword extraction, and abstract extraction, perception machine lexical analyzer, Chinese name recognition, transliteration name recognition, phrase extraction, pinyin conversion, simplified and complex conversion. Among them, word segmentation is the most widely used function.

The segmentation function of HanLP also has subdivisions, including standard segmentation, NLP segmentation, index segmentation, N-shortest segmentation path segmentation, CRF segmentation, and speed dictionary segmentation, as well as support for user-defined dictionaries.

In this article, the word segmentation function in the hanLP toolkit is used.

B. Scrapy framework

Scrapy is a crawler framework based on the Python language. Through this framework, users can quickly create web crawlers with specific functions without worrying about implementation details, reducing the coding of underlying modules. The Scrap framework abstracts crawlers into several main parts: the Scrap Engine and the Scheduler, Downloader, Spider, Item Pipeline, Downloader Middleware , Spider Middleware. Among them, only Spider and Item Pipeline do not have default implementations that require user extensions, while other parts have default implementations that can be used without user writing.

IV. Text Similarity Algorithm for Multi feature Fusion

This article proposes a text similarity calculation method that integrates text features and vector features.

A. structural similarity

Structural similarity is used to represent the degree to which words co-occur in two paragraphs of text after word segmentation and removal of stop words. In this article, MorSim is used to represent structural similarity. Let's take a look at the following two paragraphs of text:

A: Adhere to the path of socialism with Chinese characteristics B: China, abbreviated as the People's Republic of China

In both A and B, the word ”China” appears, which is a co-occurrence of A and B. For any two paragraphs of text, the more co-occurrence words they have, the higher the probability that these two paragraphs of text are similar. Let's take a look at the following two paragraphs of text:

C: Guizhou Bank's Hong Kong stock market broke through on its first day of listing, with a drop of 7.66D: On the first day of listing, Guizhou Bank's Hong Kong stock price fell 7.26

In C and D, the common vocabulary includes Guizhou, Bank, Hong Kong Stock, Listing, First Day, and Burst. This takes up two paragraphs A large portion of the text, therefore, the structural similarity of C D is higher than that of A B.

In this article, formula (1) is used to calculate structural similarity.

$$MorSim(A,B) = \frac{2 * Com(A,B)}{Len(A) + Len(B)} \tag{1}$$

Com: Number of co-occurrence words

Len: Length (total number of words after word segmentation and removal of stop words)

Now divide A and B into words

A: Persist, China, Characteristics, Socialism, Road B: China, People's Republic of China, abbreviated as

The co-occurrence vocabulary of AB is 1, the length of A is 5, and the length of B is 3. According to this formula, the similarity between A and B is 0.2500.

Similarly, segmenting CD

C: Guizhou Bank, Hong Kong Stock, Listed, First Day, Burst, Decline D: Listing, First Day, Burst, Guizhou, Bank, Hong Kong Stock, Stock Price

Calculate the co-occurrence vocabulary and length of CD, and the similarity of CD is 0.6154. From the score, it can be seen that CD's score is significantly higher than AB's. Moreover, as the co-occurrence vocabulary in the two texts increases, their similarity will also increase. This is also in line with the reality that the more similar two paragraphs of text are, the inevitable increase in the same vocabulary in them.

B. *Word order similarity*

Word order similarity is a further processing of structural similarity, which determines the degree of similarity in the order of words in two paragraphs of text after obtaining co-occurrence words. In this article, OrdSim is used to represent word order similarity. Let's take a look at the following two paragraphs of text:

A: The Chinese team defeated the South Korean team B: The South Korean team defeated the Chinese team

It can be seen that AB is almost identical in structure, with a structural similarity of 1.0. However, semantically speaking, their meanings are opposite. Let's take a look at the following two paragraphs of text:

C: The women's volleyball team won the World Cup championship, celebrating the 70th anniversary of their country's birth. This is their fifth time winning the championship and also their tenth time winning a world-class competition. D: On the afternoon of the 29th, the women's volleyball team led by Lang Ping won the 13th World Cup with a record of 11 wins in 11 matches.

From a structural perspective, the similarity of CDs is not high, with a structural similarity of only 0.4444. However, from a practical perspective, the theme of these two texts is actually consistent, both referring to the matter of "women's volleyball team winning the championship".

From the above examples, it can be seen that the order of words also has a certain impact on text features. In AB, simply because the order is inconsistent, the meaning is completely opposite. In CD, although there are fewer co-occurrence words, the similarity between the two texts is greatly improved due to the consistent order of their co-occurrence words. Word order similarity, as a text feature, is also a factor that needs to be considered.

In addition, for words that appear multiple times in the text, they have a negative impact on similarity calculation, so they are removed during calculation.

In this article, formula (2) is used to calculate word order similarity

$$\begin{cases} 1-\frac{(Comonce(A,B))}{len-1}\cdots & len>1 \\ 0.5\cdots & len=1 \\ 0\cdots & len=0 \end{cases} \qquad (2)$$

Comonce: A term that only appears once in a co-occurrence vocabulary

Len: The length of ComOnce

Inv: Number of words in reverse order

The calculation of inv is somewhat unique. Assuming that the Comonce of EF has four words, numbered 1234 in the order of E and 2341 in the order of F. It can be seen that there is a smaller 1 after 4, and all other numbers are smaller than the number to their right, so the inv result of EF is 1.

The inv value of AB is 2, and the len value is 3. The inv value of CD is 0 and len is 4. So the OrdSim of AB is 0, and the OrdSim of CD is 1. As shown in V-A.

Comparison combination	inv	len	OrdSim
(A,B)	2	3	0.0
(C,D)	0	4	1.0

C. *AHP*

Now, there are already three eigenvalues that need to be fused into one overall eigenvalues. Use formula (3) as the fusion formula

$$StrSim = \alpha \cdot MorSim + \beta \cdot OrdSim + \gamma \cdot LenSim \qquad (3)$$

And the three parameters in formula (3) need to meet equation (4)

$$\alpha + \beta + \gamma = 1 \qquad (4)$$

The problem lies in the values of these three parameters. This article uses Analytic Hierarchy Process to solve this problem [7], [8].

Analytic Hierarchy Process is a method used to solve multi-element decision-making. In this solution, all decision related elements will be decomposed to form a decision tree, and quantitative analysis will be conducted based on this decision tree to ultimately obtain the decision result.

- Establish a hierarchical model structure
  Based on the Analytic Hierarchy Process (AHP) model, establish a hierarchical model structure as shown in Fig.1

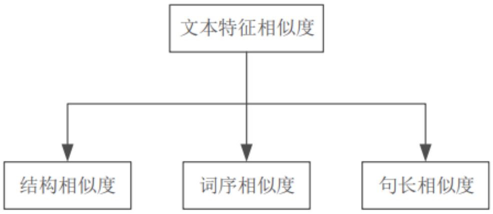

Fig. 1. Hierarchy Model Diagram

- Building a Comparison Matrix
  The parameters used in this article are shown in Table 3.2.

	structure	order features	Sentence length
structure	1	3	5
order features	1/3	1	5/3
Sentence length	1/5	3/5	1

According to the above table, the comparison matrix A can be obtained

$$A = \begin{bmatrix} 1 & 3 & 5 \\ \frac{1}{3} & 1 & \frac{5}{3} \\ \frac{1}{5} & \frac{3}{5} & 1 \end{bmatrix} \tag{5}$$

The comparison matrix formula is shown in formula (6).

$$AW = \lambda_{max} W \tag{6}$$

Where $\lambda_{max} = 3.00$, its feature vector is $\begin{bmatrix} 1.9565 & 0.6522 & 0.3913 \end{bmatrix}$, and after normalization

$$W = \begin{bmatrix} 0.625 & 0.218 & 0.130 \end{bmatrix} \tag{7}$$

V. Solutions and Evaluation

A. Experimental Result

The first part is data collection testing.

The data in this article comes from NetEase News, Sina News, and People's Daily. The collection period of articles is from May 30, 2020 to June 8, 2020. During the collection process, it was found that some articles were mainly based on images/videos, which were excluded during the collection process. The database ultimately included 7315 articles. The detailed collection results are shown in Table 5.4.

Article Source	NetEase	sina	People's Daily	overall
Total number	868	4922	1805	7595
articles collected	720	4878	1717	7315
excluded articles	3	2	4	8
collection failures	145	42	84	271
success rate	0.829493	0.991061	0.951247	0.963134

The second part is fusion parameter testing.

Firstly, for formula (3), test α and β The value of makes Sensim the most suitable for the actual situation.

When SenSim is greater than the threshold, it will be considered as an article with similar titles, and the next step of article content comparison is needed. When StrSim is less than the minimum value or greater than the maximum value, there is no need for Word2Vec vector comparison. When it is less than the minimum value, it will be considered as an article with similar titles, and when it is greater than the maximum value, it will be considered as an article with similar titles.

The third step is to test the processing speed of the system.

The data used in the test comes from historical collection data, with a total of 11558 articles. Among them, 2246 duplicate articles were detected overall.

Make a line diagram as shown in Fig.2. It can be seen that as the number of processed articles increases, the marginal cost of processing articles also increases, and the time required to process the same number of articles also increases.

VI. Conclusion and outlook

A. conclusion

Against the backdrop of rapid internet growth, online article reading is becoming increasingly popular. However, how to effectively reduce duplicate articles has always been a thorny issue.

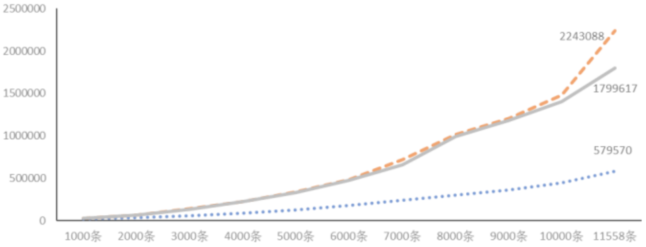

Fig. 2. Processing Time Line Chart

This article proposes a multi feature fusion text similarity algorithm by studying text similarity and observing article data on the Internet. Based on this algorithm, an online article reading system based on a crawler framework and natural language processing has been developed.

During the experiment, multiple sets of test parameters were used to test the text similarity algorithm. From the test results, it can be seen that this algorithm can ensure a certain recall rate and use less time to complete similarity calculation for all articles in the library, and label highly similar articles.

This system can achieve the function of crawling news article data from three large news aggregation platforms (Sina News, NetEase News, and People's Daily). Through the algorithm proposed in this article, news articles are processed, labeled with highly similar articles, and displayed in the form of web pages. The system also includes a series of functions such as user registration, user login, user information modification, article retrieval, and user reading history.

B. Shortcomings and Prospects

Although the algorithm proposed in this article can achieve a high recall rate, it still has certain requirements for system hardware and consumes much more time than a single text feature similarity comparison. Due to the time constraints of the graduation thesis, the development system also has problems with relatively single functions, simple system interfaces, and less refined web page styles.

In the following research, it is necessary to further optimize the algorithm, reduce hardware requirements, expand backend functions, and beautify the frontend pages.

References

[1] Danny Paskin. News publishing across platforms: Gatekeeping for print, web, facebook and twitter. *Newspaper Research Journal*, 39(4):376–388, 2018.

[2] and . . , 26(12):3030–3032, 2006.

[3] and . , , . , (12):82–83, 2015.

[4] and . . , 11(2):147–153, 1998.

[5] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[6] . Hanlp: . , (2):64–68, 2019.

[7] , , and . . , 36(8):65–68, 2019.

[8] and . . *:* , 32(5):45–49, 2017.