systems

MDPI

Article

Adapting Feature Selection Algorithms for the Classification of Chinese Texts

Xuan Liu [1], Shuang Wang [2], Siyu Lu [2], Zhengtong Yin [3], Xiaolu Li [4], Lirong Yin [5], Jiawei Tian [2] and Wenfeng Zheng [2,*]

1 School of Public Affairs and Administration, University of Electronic Science and Technology of China, Chengdu 611731, China; liuxuan@uestc.edu.cn
2 School of Automation, University of Electronic Science and Technology of China, Chengdu 610054, China; wangshuang4183@sefonsoft.com (S.W.); siyu.lu@std.uestc.edu.cn (S.L.); jravis.tian@std.uestc.edu.cn (J.T.)
3 College of Resource and Environment Engineering, Guizhou University, Guiyang 550025, China; ztyin@gzu.edu.cn
4 School of Geographic Science, Southwest University, Chongqing 400715, China; xliswu@swu.edu.cn
5 Department of Geography and Anthropology, Louisiana State University, Baton Rouge, LA 70803, USA; lyin5@lsu.edu
* Correspondence: winfirms@uestc.edu.cn

Abstract: Text classification has been highlighted as the key process to organize online texts for better communication in the Digital Media Age. Text classification establishes classification rules based on text features, so the accuracy of feature selection is the basis of text classification. Facing fast-increasing Chinese electronic documents in the digital environment, scholars have accumulated quite a few algorithms for the feature selection for the automatic classification of Chinese texts in recent years. However, discussion about how to adapt existing feature selection algorithms for various types of Chinese texts is still inadequate. To address this, this study proposes three improved feature selection algorithms and tests their performance on different types of Chinese texts. These include an enhanced CHI square with mutual information (MI) algorithm, which simultaneously introduces word frequency and term adjustment (CHMI); a term frequency–CHI square (TF–CHI) algorithm, which enhances weight calculation; and a term frequency–inverse document frequency (TF–IDF) algorithm enhanced with the extreme gradient boosting (XGBoost) algorithm, which improves the algorithm's ability of word filtering (TF–XGBoost). This study randomly chooses 3000 texts from six different categories of the Sogou news corpus to obtain the confusion matrix and evaluate the performance of the new algorithms with precision and the F_1-score. Experimental comparisons are conducted on support vector machine (SVM) and naive Bayes (NB) classifiers. The experimental results demonstrate that the feature selection algorithms proposed in this paper improve performance across various news corpora, although the best feature selection schemes for each type of corpus are different. Further studies of the application of the improved feature selection methods in other languages and the improvement in classifiers are suggested.

Keywords: text classification; feature extraction; classifier; algorithm; Chinese text

Citation: Liu, X.; Wang, S.; Lu, S.; Yin, Z.; Li, X.; Yin, L.; Tian, J.; Zheng, W. Adapting Feature Selection Algorithms for the Classification of Chinese Texts. *Systems* **2023**, *11*, 483. https://doi.org/10.3390/systems11090483

Academic Editors: Carlos de las Heras-Pedrosa, Francisco Javier Paniagua-Rojano and Dolores Rando-Cueto

Received: 11 August 2023
Revised: 15 September 2023
Accepted: 19 September 2023
Published: 20 September 2023

1. Introduction

Text is the most basic and common means to communicate in the Digital Media Age. To quickly and accurately obtain the target data among the massive electronic text data, classifying texts becomes the first step. Text classification is an associated technology that combines data mining and pattern recognition in the specific field of text analysis [1]. It establishes classification rules based on the features of the text. Automatic text classification based on machine learning classifies the target texts into one or more predefined categories, significantly reduces manual workload, and greatly improves work efficiency [2]. The step of textual feature selection greatly influences the classifier's performance. Commonly used feature selection algorithms include the mutual information algorithm (MI) [3]

and the CHI–square statistics algorithm (CHI) [4]. The feature selection algorithms generally use calculation functions to reduce the dimension of textual features and the difficulty of classification tasks [5]. A general processing flow of feature selection is the following: according to the characteristics of the text data set, select a suitable feature calculation function through a particular process, perform feature calculations on each term in each text in the data set to obtain quantitative results, and sort the results from largest to smallest. A certain number of feature items are selected as representatives of the original text data according to the threshold set in advance.

In recent years, the research on text classification for Chinese texts has achieved significant progress, especially in the fields of textual feature selection [6,7] and classifier construction [8]. However, existing feature selection algorithms always have some defects. For example, CHI generally overestimates low-frequency words, and MI always tends to select low-frequency features. To mitigate the problem of overestimating low-frequency words in feature extraction tasks by CHI and the problem of overestimating the category-specific information associated with the low-frequency words by MI, a balancing strategy is needed to improve the overall performance of text classification.

Furthermore, with the increasing networking of Chinese social life, Chinese text classification needs to deal with an increasingly large amount of text data. Different text types have different features, and feature selection methods must also be distinguished. Including multiple algorithms for different needs has become increasingly important [9,10]. However, such studies could seldom be found by far and are urgently needed.

In response to the above two gaps, this study proposes three different feature selection algorithms to improve the performance of text classification for Chinese texts and tests their performance in the classification of different types of Chinese texts. The three algorithms include a CHMI algorithm proposed by combining an improved CHI considering low-frequency words and an improved MI reducing the overestimation of the features carried by low-frequency words; a TF–CHI algorithm proposed by combining CHI and TF–IDF algorithm for more accurate weight calculations; and a TF–XGB algorithm to prescreen the feature words.

The rest of this study is organized as follows: Studies about existing text classification and feature selection methods in the literature are summarized in Section 2. The adjustments to the algorithms for feature selection are employed in Section 3, followed by the experiments on both SVM and NB classifiers in Section 4. The experimental results indicate the feasibility of the three feature selection algorithms, among which the TF–CHI feature selection algorithm demonstrates the best performance and significantly contributes to improving classification accuracy. Through the experiments conducted in the sections above, the final results consistently validate the feasibility and effectiveness of the proposed improvement methods.

2. Related Work

Before the 1960s, text classification was generally achieved by experts formulating rules and manual classification. In the Digital Media Age, the rapid growth of digital texts has created the need for automatic text classification. The development of automatic text classification mainly experienced three stages:

(1) Search for basic theory. In the 1950s, Dr. Luhn proposed the method of automatically creating article abstracts by word frequency statistics, which constituted the core idea of early text classification [11]. In 1960, Maron and Kuhn proposed a probabilistic indexing model for document indexing and searching in the library scene [12].

(2) Experimental exploration. Maron designed an experiment based on the 'Bayesian hypothesis' to realize automatic indexing of texts according to text keyword information [13]. The 'Vector Space Model' proposed by Salton and Wang represented those terms with text topic features in the text as feature vectors and transformed the problem of calculating the similarity of text into the issue of calculating the cosine of the included angle for the feature vector corresponding to the text [14].

(3) Application of machine learning. In 2000, scholars proposed a model that can learn through the distribution of words and the probability function of word sequences and achieved good results [15]. In 2008, scholars proposed a general deep neural network (DNN) when dealing with natural language process tasks [16]. After that, they offered a multi-functional learning algorithm and obtained a relatively unified neural network with word vectors [17]. In 2013, researchers from Google used the continuous skip-gram model to train the distributed representation of words and phrases and proposed a harmful sampling method that can replace hierarchical softmax [18]. In the same year, Brandon proposed for the first time a network model using a multilayer neural network, which has a more vital learning ability [19]. In 2014, Kim used convolutional neural networks (CNN) in text classification and achieved excellent results [20].

The research on automatic text classification for Chinese texts started relatively late. Shi and Bai first used a new convolutional recurrent neural network (CRNN) model to train a character recognition system [21]. Later, in 2018, Cao et al. [22] from Ant Group published a cw2vec algorithm using Chinese stroke information for feature selection. In 2019, Wan et al. [23] proposed a SABigram algorithm based on text structures, which can extract compound features from texts. Zhu and Yang [24] suggested a feature selection algorithm introducing the distribution expressiveness index between feature word categories. In 2022, with the use of leveraging adversarial training and contrastive learning, Pan et al. proposed a regular fine-tuning method based on the transformer model [25].

Automatic text classification generally includes several vital processes, namely text data preprocessing, text representation, feature selection, classification model training, classification performance evaluation, and other operations [26–28]. The key to text classification is how to preserve the complete text content features as much as possible in the feature selection operation. Commonly used feature selection algorithms include document frequency statistics [29], information gain algorithm [30], mutual information algorithm, and CHI–square statistical algorithm. However, these methods generally only consider whether a term exists in the text and ignore the frequency of the term in the text. For example, CHI has the problem of overestimating low-frequency words, and MI always tends to select low-frequency features.

To fill the gap, the researchers proposed the concept of weight calculation. The weight value of a feature word is a measure of its text representation ability. The feature weighting algorithm can calculate different feature weight values according to the distribution of category feature keywords in the data set text to achieve a better text classification effect [31]. The most commonly used feature weighting calculation method is the term frequency–inverse document frequency (TF–IDF) algorithm [32]. In addition, there are algorithms such as token flow control (TFC) and link-cut tree (LTC).

In actual practice, due to the significant differences in the characteristics of different types of texts, direct use of any of the above algorithms cannot guarantee a good classification effect. A common solution is to use the object text to vectorize the feature terms and their weight values obtained through screening. However, for different corpora, word frequency or the distribution ratio of terms between categories can play different roles. There is, therefore, a long-term need for improvements to the algorithm to adapt to different types of text.

3. Materials and Methods

3.1. Workflow

As shown in Figure 1, the primary process to adjust feature selection algorithms for different types of texts includes five steps. First, this study chooses the Sogou news corpus to build the training and test sets. Data preprocessing includes removing useless marks, text segmentation, removing functional words, and structured representation. Feature selection is carried out with four different algorithms on two different classifiers. The results are evaluated by comparing the classification indexes. Finally, the performance in different

fields is evaluated with a confusion matrix so that further suggestions for feature selection can be made.

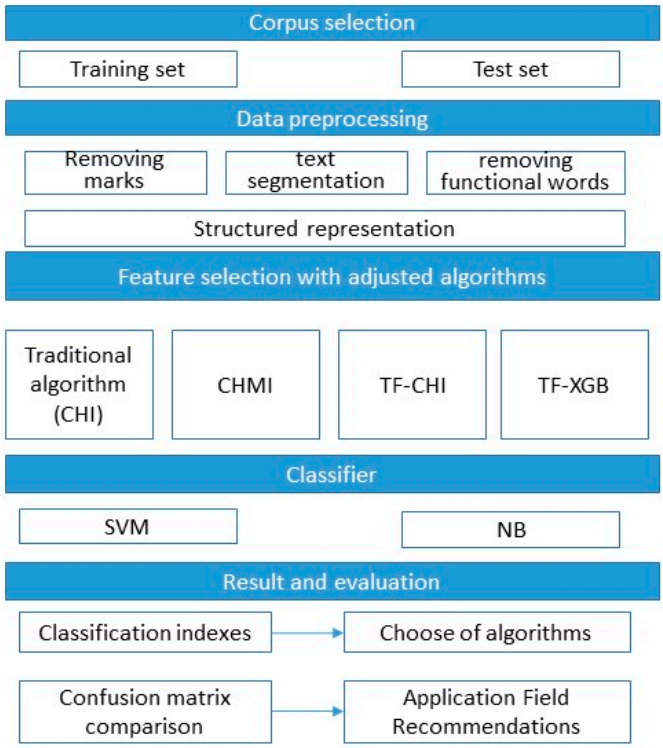

Figure 1. Workflow.

3.2. Corpus Selection

This study uses the Sogou news dataset and collates the corpus according to the needs. The Sogou news corpus is a corpus Sogou Labs uses for text classification. It contains text data in 10 categories: automobile, finance, IT (Internet technology), health, education, military, tourism, sports, culture, and recruitment. Each category contains 1990 articles. Text (https://www.kaggle.com/datasets/jarvistian/sogou-news-corpus, accessed on 1 September 2022). This article selects six categories with relatively distinctive corpus characteristics, namely automobile, finance, military, health, sports, and IT, as the data set used in this article.

The purpose of the experimental design of this study is to verify the effectiveness of the proposed feature extraction method. Due to time and resource constraints, we chose a smaller-scale data set for the experiment. Download 500 pieces of text data under each category to form a data set with a size of 3000. There was no overfitting in the 3000 corpus during the experiment, which met the preliminary verification requirements. Each piece of news is saved in the data format of "category", "theme", "URL", "content", and "source", as shown in Figure 2.

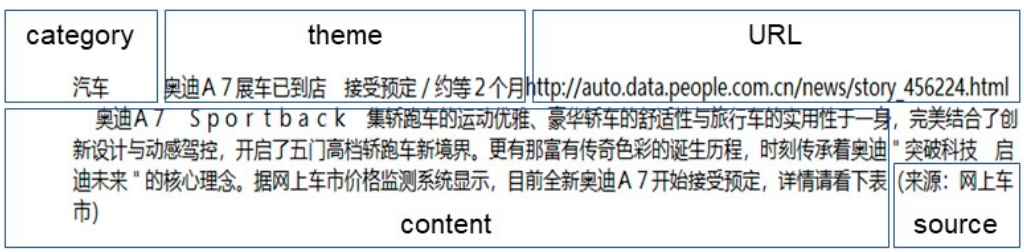

Figure 2. Example of news in Sogou news corpus.

The selected Sogou news was randomly extracted to form a training set and a test set at a ratio of 7:3.

3.3. Data Preprocessing

3.3.1. Removing Marks

The original text usually contains commonly used marks, such as emoticons, pictures, or links. They will increase the amount of calculation in the classification process and lower the accuracy of the classification results. This study implements batch processing of data sets through regular expression matching scripts.

3.3.2. Text Segmentation

There is a natural gap between any adjacent words in English and other Latin languages, but in Chinese, this gap does not exist. The basic principle of Chinese text word segmentation is to use the word segmentation algorithm to identify punctuation marks or certain Chinese words in sentences and add separators at these occurrence positions.

This study uses the Jieba library [33] in Python as a word segmentation tool. Jieba uses predefined dictionaries and word frequencies to find the most appropriate word segmentation combination in the sentence through dynamic programming.

3.3.3. Removing Functional Words

Many functional words in Chinese connect adjectives and nouns, adverbs and verbs, or adjacent sentences. Although functional words such as "的", "地", "而且", "还" enjoy a high frequency of use, they could hardly provide any discriminative information for text classification. This part refers to the Chinese stop word list that Zhou, Ya et al. [34] provided to set the stop word selection scheme.

3.3.4. Structured Representation

After text preprocessing, the obtained text data is like this: "我/热爱/围棋/篮球" (I/love/Go/basketball). Generally, it is necessary to express the text with a mathematical description through some method so as to become a language that the computer can recognize. This study uses Word2Vec [35] as the text representation model.

3.4. Adjusting Feature Selection Algorithms

3.4.1. A CHMI Algorithm Considering Word Frequency

(1) Improvement of CHI algorithm

CHI algorithm has good quantization ability for text features and is often used for text classification problems and used as one of the essential algorithms [36]. However, the algorithm only counts the number of words in the text and ignores the critical factor that the word has word frequency information in the text. An improved algorithm is required to choose more appropriate feature words due to the above problems. This paper proposes a word frequency factor based on the word item, whose size is equal to the ratio of the word item's frequency in category documents to its text frequency in whole text dataset documents. Its calculation equation is as follows:

$$\alpha(t_i, c_j) = \frac{n(t_i, c_j)}{n(t_i)} \tag{1}$$

In Equation (1), $n(t_i, c_j)$ refers to the frequency of occurrence of the word term t_i when the category result is c_j, and $n(t_i)$ refers to the text frequency of the word item t_i in the text in the entire text dataset document.

The size of the word frequency factor $\alpha(t_i)$ is determined by the frequency of occurrence of the word item t_i in a specific category of documents and the frequency of occurrence of this word item in all text dataset documents. The larger the value of the word frequency factor $\alpha(t_i)$, the higher the frequency that the word item t_i appears only

in this category document, and the smaller the value of the word frequency factor $\alpha(t_i)$, the less frequently the word item t appears in this category document, that is, it is easier to find in other category documents. By introducing the word frequency factor $\alpha(t_i)$, it is easier to find feature words that are more helpful for classification by using the CHI–square method.

(2) Improvement of MI algorithm

The basic idea of the MI algorithm is relatively straightforward. It can not only consider the correlation between the word items but also the intra-class distribution and relationship of the word items. However, it also has the problem of overestimating the amount of category feature information of low-frequency words. The traditional MI algorithm does not consider the situation in the frequency of word items that may appear in the text. In this study, an adjustment factor based on word items is proposed, whose size is equal to the ratio of the number of text with this word frequency to the total number of text in the category document, as shown in Equation (2):

$$D(t_i, c_j) = \frac{d(t_i, c_j)}{d(c_j)} \tag{2}$$

In Equation (2), $D(t_i, c_j)$ refers to the text frequency of the word item t_i in the document with the category result c_j, and $d(c_j)$ refers to the total text frequency in the document with the category result c_j.

The size of the adjustment factor $D(t_i, c_j)$ is determined by the frequency of the text in which the term t_i exists in the category document and the frequency of the text in which the term t_i does not exist in the category document. The larger the value of the adjustment factor $D(t_i, c_j)$, the higher the proportion of the text with the word item t_i in the document with the category result c_j. On the contrary, the smaller the value of the adjustment factor $D(t_i, c_j)$, the smaller the proportion of the text with the word item t_i in the document with the category result c_j. That is, such words are more likely to be low-frequency words in the category document. By introducing the adjustment factor $D(t_i, c_j)$, we can use the mutual information algorithm to eliminate better the feature words that may cause errors in classification.

(3) Formation of CHMI algorithm

Synthesizing the improved methods proposed above, an improved feature selection algorithm function is proposed as Equation (3):

$$CHMI(t_i, c_j) = \rho * \left[CHI(t_i, c_j) \times \alpha(t_i, c_j)\right] + (1 - \rho) \times \left[MI(t_i, c_j) \times D(t_i, c_j)\right] \tag{3}$$

In the above equation, $\rho \in (0, 1)$. Similarly, when dealing with multi-classification problems, the above equation can be improved as Equation (4):

$$\begin{aligned} CHMI_{Max}(t_i, c_j) = \rho \times &\left[CHI_{Max}(t_i, c_j) \times \alpha(t_i, c_j)\right] \\ &+ (1 - \rho) \times \left[MI_{Max}(t_i, c_j) \times D(t_i, c_j)\right] \end{aligned} \tag{4}$$

3.4.2. A TF–CHI Algorithm Considering Word Weights

(1) TF–IDF algorithm

To achieve better classification results, combining feature weight calculations for feature processing is often necessary before the feature results are applied for subsequent text classification training. The weight value of a feature keyword generally refers to the measurement of the text representation ability of each feature keyword. The TF–IDF algorithm, which takes into account both the word frequency and document frequency of feature keywords, is now a common feature weighting calculation method [32]. If the document frequency of a feature keyword is high, it means that the feature keyword appears

in many texts and could discriminate the categories. Thus, a lower weight value should be issued for this feature keyword, which is shown in Equation (5):

$$\omega = tf \times log\frac{N}{df+1} \tag{5}$$

In Equation (5), ω represents the weight result of a particular feature keyword, and tf is the word frequency; df is the document frequency, which specifically refers to the number of texts containing the feature keywords; and N refers to the number of texts in the training text dataset.

(2) The TF–CHI algorithm

TF–IDF algorithm generally works well. However, word frequency or the distribution of instances between categories can play different roles for different corpus data, and a better feature selection method can be found by optimizing related algorithms. The critical problem of the TF–IDF algorithm is the miscalculation of the weight for some words. For some words in the dataset, which are evenly distributed across every class of documents, the obtained weight result is too high; For words that only appear in documents of a specific category, the resulting weight is too small.

Taking the above disadvantages of the TF–IDF algorithm into consideration, this study plans to adjust the TF–IDF algorithm with the CHI–square statistical algorithm. The CHI–square statistical algorithm takes into account the distribution of words across categories, as shown in Equation (6):

$$CHI(t_i, c_j) = \frac{N(AD - CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{6}$$

$CHI(t_i, c_j)$ represents the CHI–square statistical value of term t_i and category c_j. A refers to the number of texts with term t_i in the document with category result c_j, B refers to the number of texts with term t_i in the document whose category result is not c_j in the training text dataset, C refers to the number of texts without term t_i in the document with category c_j, and D refers to the number of texts in which the term t_i does not exist in the document whose category result is not c_j in the training text dataset. N refers to the number of texts contained in the entire training text dataset, and $N = A + B + C + D$.

The CHI–square statistical algorithm quantifies textual features well [37]. This method assumes that the term t_i and the category c_j satisfy the CHI–square (χ^2) distribution of the first-order degree of freedom and calculates the correlation between the term and the category as the selection criteria. The CHI–square statistical algorithm can calculate the correlation of all terms in every text and then select the terms according to the correlation.

Combining both the TF–IDF algorithm and the CHI–square statistical algorithm, this study proposes a new algorithm for feature extraction—the TF–CHI algorithm. The calculation equation is proposed as Equation (7):

$$\omega(t_i, c_j) = tf \times log\frac{N}{df+1} \times CHI(t_i, c_j) \tag{7}$$

When using Equation (7) to calculate the weight results of terms in multi-text classification, the final result retains the maximum value of $\omega(t_i, c_j)$.

3.4.3. A TF–XGB Algorithm for Dimension Reduction

(1) The XGBoost algorithm

The gradient boosting algorithm is a typical implementation of ensemble learning [38]. It uses the basic principle of gradient descent method, uses cart classification regression tree as a weak learner, and ensures that the predicted loss value of $F(x)$ is gradually reduced before each weak learner is added through the idea of "gradual refinement", thus obtaining the minimum loss function of each iteration.

The XGBoost algorithm [39] is another improvement on the gradient boosting algorithm, which is mainly optimized from the level of the algorithm itself, reflected explicitly in the second-order Taylor expansion of the error part of the XGBoost loss function, which can make the obtained results more accurate. XGBoost algorithm can use the method of cyclic iteration to make the data processing in text classification have more advantages. The text classification results based on the XGBoost algorithm also generally have better classification results. However, in the process of Chinese text classification, the classification efficiency of this algorithm is often low, and sometimes, it is not easy to deal with feature words in high-dimensional feature space.

(2) The TF–XGB algorithm

To complete the prescreening of feature words for the classification process, this study uses the TF–IDF weight calculation method to filter out some words with low weight first and then uses the XGBoost algorithm to filter the remaining words to obtain the final feature word set. The specific steps include the following:

(1) According to the distribution of terms in the data set, Equation (5) of the TF–IDF algorithm is used to calculate the weight values of all terms, and then a unique index is generated for each term. This study then selects the most prominent m feature words and obtains their term indexes, orders them by weight, and saves them as a sequence $(1,\ 2,\ ...,\ m)$.
(2) According to the term index obtained in Step (1), this study obtains and generates a corresponding set of terms: $X_1, X_2, \cdots, X_m$ (from large to small).
(3) For all the words in the set of terms, the number of times each word in the decision tree is selected as the optimal partition attribute to separately count the importance of these words [40] until the optimal partition attribute times of all the words are obtained and recorded as $N_1, N_2, \cdots, N_m$ (from large to small).
(4) This study then uses Equation (8) to calculate the importance values of all the words, which are arranged in descending order and recorded as $c_1, c_2, \cdots, c_m$.

$$c_i = \frac{N_i}{\sum_{i=1}^{m} N_i} \tag{8}$$

(5) $c_1, c_2, \cdots, c_m$ are accumulated and calculated according to $c_1, c_1 + c_2, \cdots, c_1 + c_m$ method, and the results are recorded as $C_1, C_2, \cdots, C_m$.
(6) The term set S_k can be obtained, as shown in Equation (9). k is the threshold to choose feature words and could only be determined after multiple classification experiments on the same data set by manually evaluating the variation trend of the F_1-score of the classification result.

$$S_k = \{C_1, C_2, \ldots, C_k\}, \qquad k \leq m \tag{9}$$

3.5. Classifier Selection and Evaluation Setting

This study chooses two classifiers—the SVM algorithm and the NB algorithm—to evaluate the performance of the adjusted feature selection algorithms.

(1) Number of features for evaluation with SVM classifier

The text classification of the Sogou corpus is first carried out with the SVM algorithm as the classifier, and the kernel function is preferably determined as linear. The feature selection algorithms are CHI, CHMI, TF–CHI, and TF–XGB, and simulation experiments are carried out in this environment. First, in the first part of the experiment, the number of features is set to be 100–1500. The purpose is to verify the relationship between the text classification and the number of features under the four feature selection methods. According to the above purpose, the experiment is designed, and the F_1-score of the classification result is obtained through data mapping to obtain the relationship curve, as shown in Figure 3.

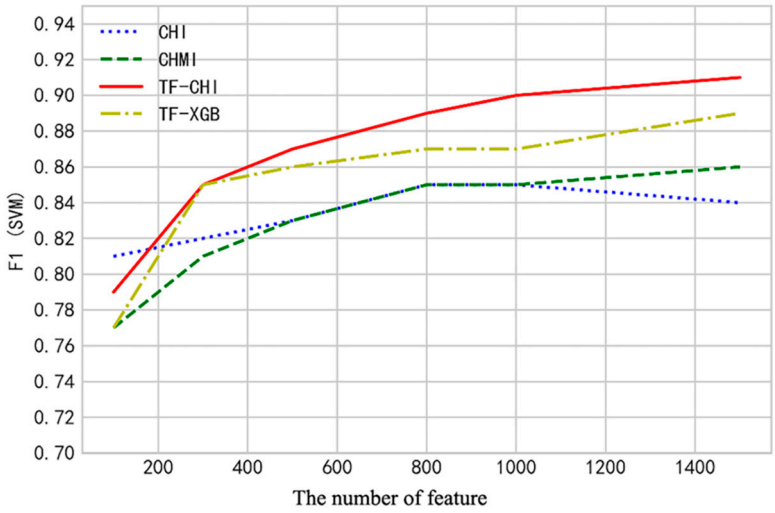

Figure 3. Relationship curve between the number of features and F_1-score under SVM.

According to Figure 3, the F_1-score increases with the increase in the number of features in the range of 100–1000. After 1000, the performance of the CHI algorithm starts to decrease gradually. So, this study sets 1000 as the feature number (threshold k) to compare the performance of feature selection algorithms with the SVM classifier.

(2) Number of features for evaluations with NB classifier

With the NB algorithm as the classifier, CHI, CHMI, TF–CHI, and TF–XGB are used as feature selection algorithms. Similar to the setting for the SVM classifier, the relationship between the text classification and the number of features is verified by setting the number of features to 100–1500. According to the classification result F_1-score obtained at this time, the following figure is prepared:

According to Figure 4, when the number of features is 100–1000, the F_1-score increases with the increase in the number of features. When the number of features reaches 1000, the F_1-score of CHI almost stagnates. Again, this study sets 1000 as the feature number to compare the performance of feature selection algorithms with the NB classifier.

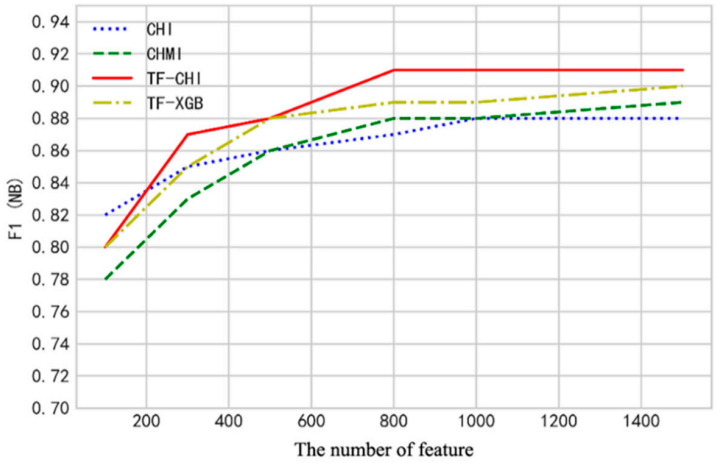

Figure 4. Relationship curve between the number of features and F_1-score under NB.

3.6. Evaluation of the Performance of Feature Selection Algorithms

First, the classification confusion matrix is introduced in Table 1, and the indexes are calculated accordingly. The confusion matrix is also applied when dealing with multi-class classification, where its own category is treated as "positive," and all other categories are treated as "Nnegative."

Table 1. Classification confusion matrix.

	Real Category (Positive)	**Real Category (Negative)**
Forecast category: (Positive)	TP	FP
Forecast category: (Negative)	FN	TN

Based on the classification confusion matrix, this study mainly uses two indexes, P and F_1-score, to evaluate the performance of feature selection algorithms. To obtain the F_1-score, the index of recall rate is also needed.

(1) Precision

Precision reflects the proportion of correctly classified results. Equation (10) is applied to calculate the precision:

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

(2) Recall Rate

Recall rate reflects the probability that the text whose real category is positive is still predicted to be positive after being classified. The way to calculate the recall rate is shown in Equation (11):

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

(3) F_1-score

The F_1-score is the harmonic mean of precision and recall. The larger the F_1 is, the more effective the method is. The calculation of F_1-score is shown in Equation (12):

$$F_1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{12}$$

4. Results

4.1. Classification Results and Comparison for SVM Classifier

Table 2 shows the results of the classification of the SVM classifier with different feature selection algorithms for various news categories. Obviously, the overall P and F_1-score values of the improved CHMI, TF–CHI, and TF–XGB algorithms have been significantly improved compared with the CHI algorithm. The average precision has been improved by 3%, 7%, and 5%, respectively, and the average F_1-score has been improved by 4%, 9%, and 6%, respectively.

Table 2. Classification results of the SVM classifier.

	Category / Evaluate	**Military**	**Sports**	**IT**	**Economy**	**Automobile**	**Health**	**Average**
Precision	CHI	66%	91%	87%	76%	93%	85%	83%
	CHMI	70%	**95%**	90%	79%	97%	**87%**	86%
	TF–CHI	88%	**95%**	**92%**	**84%**	99%	83%	**90%**
	TF–XGB	**89%**	94%	89%	79%	**100%**	77%	88%
F_1-score	CHI	76%	92%	75%	78%	90%	79%	81%
	CHMI	79%	93%	81%	82%	93%	83%	85%
	TF–CHI	**90%**	**95%**	**85%**	**86%**	**97%**	**87%**	**90%**
	TF–XGB	89%	95%	80%	82%	91%	83%	87%

Among the three improved algorithms, TF–CHI has a more significant improvement effect and can achieve an overall average precision of more than 90% and an F_1-score of 90%.

For the categories of military and automobile, TF–XGB outperforms TF–CHI. The precision values are 89% and 100%, respectively. For the health category, CHMI outperforms TF–CHI and TF–XGB algorithms. The precision is 87%, which is 2% higher than CHI, 4% higher than TF–CHI, and 10% higher than TF–XGB algorithms.

This study also employs the confusion matrices to show further the detailed performance of different feature selection algorithms for various categories. The confusion matrices obtained by different feature selection algorithms are shown in Table 3.

Table 3. Confusion matrix results of the SVM classifier.

			Real Category					
			Military	**Sports**	**IT**	**Economy**	**Automobile**	**Health**
Forecast category	Military	CHI	**138**	8	18	13	18	19
		CHMI	**136**	3	15	10	13	15
		TF–CHI	**140**	5	3	7	1	4
		TF–XGB	**138**	3	2	7	4	3
	Sports	CHI	3	**138**	4	2	1	2
		CHMI	3	**143**	2	1	0	2
		TF–CHI	5	**145**	1	0	0	1
		TF–XGB	4	**146**	1	0	0	1
	IT	CHI	1	1	**91**	9	0	3
		CHMI	2	1	**104**	4	2	2
		TF–CHI	0	0	**109**	7	0	2
		TF–XGB	1	1	**100**	6	2	2
	Economy	CHI	3	0	15	**115**	0	16
		CHMI	5	1	14	**119**	0	13
		TF–CHI	3	1	15	**127**	0	6
		TF–XGB	2	1	21	**123**	0	8
	Automobile	CHI	2	0	6	1	**130**	1
		CHMI	1	2	2	1	**136**	0
		TF–CHI	0	3	0	0	**140**	1
		TF–XGB	0	0	0	0	**134**	0
	Health	CHI	5	3	4	5	0	**111**
		CHMI	3	3	5	4	1	**122**
		TF–CHI	4	1	10	4	8	**138**
		TF–XGB	5	2	14	9	9	**138**

Bold numbers in Table 3 show the correctly classified results for each feature selection algorithm for each category. The number of test samples in each category is 150.

The three improved algorithms have the most significant effect on the promotion of health categories, and the number of correctly classified texts has increased by nearly 22 on average. The promotion effect of the military category is the worst, with an average increase of less than one.

When the TF–CHI algorithm is used as the feature selection algorithm, compared with CHI, the total number of correctly classified texts increases by 77. In addition, the number

of correctly classified texts in the four categories of health, IT, economy, and the automobile has increased dramatically by 27, 18, 12, and 10, respectively, while the improvement for the other two categories is not apparent.

When the CHMI algorithm is used as the feature selection algorithm, compared with CHI, the total number of correctly classified texts increases by 34. In addition, the number of correctly classified texts in the two categories of IT and health has increased significantly, increasing by 13 and 11, respectively, while the improvement for the other four categories is not obvious. The number of correctly classified texts in the military category has even been reduced by five.

When the TF–XGB algorithm is used as the feature selection algorithm, compared with CHI, the total number of correctly classified texts increases by 55. In addition, the number of correctly classified texts in the three categories of health, IT, and economy has increased dramatically by 27, 9, and 8, respectively, while the improvement effect on the other three categories is not obvious.

4.2. Classification Results and Comparison for NB Classifier

Similarly, the three feature selection algorithms are applied to classify various categories for the NB classifier. Table 4 shows the results of the classification of the NB classifier with different feature selection algorithms for various categories of news. Again, the overall average Precision and F_1 values of the three algorithms have been significantly improved compared with the CHI algorithm. The overall average precision has been improved by 3%, 6%, and 5%, respectively, and the overall average F_1-score has been improved by 4%, 9%, and 7%, respectively. Among the three improved algorithms, TF–CHI has the greatest improvement effect and can achieve an overall average precision of 91% and F_1-score of 91%.

Table 4. Classification results of the NB classifier.

	Category / Evaluate	Military	Sports	IT	Economy	Automobile	Health	Average
Precision	CHI	93%	93%	56%	77%	94%	91%	85%
	CHMI	97%	96%	60%	82%	97%	97%	88%
	TF–CHI	95%	94%	83%	83%	97%	94%	91%
	TF–XGB	96%	92%	81%	77%	99%	90%	90%
F_1-score	CHI	78%	94%	67%	82%	86%	84%	82%
	CHMI	85%	96%	72%	87%	88%	90%	86%
	TF–CHI	92%	96%	84%	87%	97%	91%	91%
	TF–XGB	91%	95%	80%	83%	97%	89%	89%

The confusion matrices obtained by different feature selection algorithms for the NB classifier are shown in the following Table 5.

Table 5. Confusion matrix results of the NB classifier.

		Real Category					
		Military	**Sports**	**IT**	**Economy**	**Automobile**	**Health**
Military	CHI	**111**	1	0	4	1	3
	CHMI	**126**	0	0	2	0	2
	TF–CHI	**132**	1	0	1	0	5
	TF–XGB	**143**	0	0	2	0	4

Table 5. *Cont.*

		Real Category					
		Military	**Sports**	**IT**	**Economy**	**Automobile**	**Health**
Sports	CHI	4	**143**	2	1	0	3
	CHMI	2	**147**	1	1	1	2
	TF–CHI	6	**147**	2	0	0	1
	TF–XGB	7	**148**	2	1	0	3
IT	CHI	22	2	**115**	8	32	17
	CHMI	11	3	**118**	5	29	14
	TF–CHI	4	1	**116**	8	6	4
	TF–XGB	3	1	**108**	9	8	4
Economy	CHI	11	0	17	**128**	0	10
	CHMI	9	0	11	**128**	0	8
	TF–CHI	5	0	15	**134**	0	7
	TF–XGB	8	1	21	**129**	1	8
Automobile	CHI	2	0	3	1	**117**	1
	CHMI	1	0	2	1	**120**	0
	TF–CHI	0	0	3	0	**143**	1
	TF–XGB	0	0	2	0	**141**	0
Health	CHI	4	4	1	3	0	**118**
	CHMI	1	2	0	1	0	**136**
	TF–CHI	3	1	2	2	1	**134**
	TF–XGB	5	0	5	4	0	**133**

Bold numbers in Table 5 show the correctly classified results for each feature selection algorithm for each category.

The three algorithms have the worst improvement effect on the category of IT, and the number of correctly classified texts in this category is reduced by more than one on average compared with CHI. They perform best in the military category, with an average increase of more than 26.

When the TF–CHI algorithm is used for feature selection, compared with the CHI algorithm, the total number of correctly classified texts increases by 87. The numbers of correctly classified texts in the three categories of military, automobile, and health have significantly increased by 34, 26, and 16, respectively. But the improvement for the other three categories is not obvious.

When the CHMI algorithm is used for feature selection, compared with the CHI algorithm, the total number of correctly classified texts increases by 42. In addition, the numbers of correctly classified texts in the two categories of health and military significantly increased by 18 and 14, respectively, while the improvement for the other four categories is not obvious. The number of correctly classified texts for economy even reduces to 0.

When the TF–XGB algorithm is used for feature selection, compared with the CHI algorithm, the total number of correctly classified texts increases by 69. In addition, the numbers of correctly classified texts in the three categories of military, automobile, and health significantly increased by 31, 24, and 15, respectively. The improvement for the other three categories is not obvious, and the number of correctly classified texts for the category of IT was even reduced by seven.

5. Discussion and Conclusions

This study is an attempt to improve the existing feature selection algorithm in different ways to adapt them for the classification of various categories of Chinese texts. Firstly, the word frequency factor is proposed to improve the sensitivity of CHI to low-frequency words. In view of the shortcomings of the MI algorithm, an adjustment factor is added, and finally, an improved new algorithm combining the two methods is formed as the CHMI algorithm. Secondly, according to the situation that the TF–IDF algorithm may obtain inaccurate weight results when calculating and processing the weight values of some words with a uniform distribution in the feature weight calculation section, the CHI algorithm is added to improve it to a TF–CHI. Finally, we also use the XGBoost algorithm to double-process the features processed by the TF–IDF algorithm to obtain a new TF–XGB algorithm. The above-improved algorithms are implemented on both SVM and NB classifiers, and the result shows that they have greatly improved the performance of the classifiers.

The excellent performance of the CHMI algorithm in sports and health categories may be attributed to the presence of numerous specific terms and frequent occurrences in these domains. As the CHMI algorithm is more sensitive to word frequency information, it is better suited for such domains. On the other hand, the TF–CHI algorithm exhibits outstanding performance in classifying news about the economy, where different industries like banking, insurance, and real estate have their specific industry-specific terms, and the distribution of terms among categories is uneven. TF–CHI addresses potential issues with traditional weight calculation that could lead to errors in the results, making it more suitable for the imbalanced distribution of the news about the economy. In the automobile domain, the TF–XGB algorithm outperforms other algorithms. The automobile theme can encompass various aspects, including technical specifications, market trends, environmental impacts, and consumer reviews. By filtering terms based on importance with TF–IDF and then applying XGBoost, the algorithm can consider multiple dimensions of automobile-related content, leading to more accurate classification. Overall, the choice of the appropriate feature selection algorithm depends on the specific characteristics and language usage patterns within each domain and affects the algorithm's performance in classifying texts from different categories.

Compared with English text, the processing of Chinese text has only more work for word segmentation, and there is little difference in feature selection and classification. While the specific research findings are confined in scope, the methodology itself possesses inherent scalability. Therefore, we believe that the feature extraction method proposed in this study can also be applied to classifying English texts after specific adjustments. In future research, we will devote ourselves to researching a mixed model with various feature selection algorithms built into the model. Through adaptive dynamic weight assignment, the best feature selection scheme is automatically matched during data processing to obtain more accurate results.

Both feature selection and classifier construction are the key steps of text classification and define the effectiveness of an automatic text classification method. In this study, we set SVM and NB—the most frequently used algorithms—as the classifier. However, further discussion could be carried out to choose and improve the classifier. Such improvement could be a different setting for the global and local kernel functions for SVM, the introduction of modern neural network (NN) methods as the classifier, or allowing multi-label classification. The NN method is a machine learning method that simulates the neural network of the human brain. The basic structure of the NN method is more complex, and the data processing is more prosperous. In the future, we would also try to adjust the classifiers with new methods for better performance of text classification.

Author Contributions: Conceptualization, L.Y. and W.Z.; methodology, X.L. (Xuan Liu), S.W. and S.L.; formal analysis and investigation, Z.Y., X.L. (Xiaolu Li) and S.W.; writing—original draft preparation, X.L. (Xuan Liu), J.T. and S.L.; writing—review and editing, L.Y., X.L. (Xuan Liu), S.L. and W.Z.; software, X.L. (Xiaolu Li) and Z.Y.; data curation, X.L. (Xuan Liu) and S.W.; visualization, J.T.; resources, X.L. (Xiaolu Li) and Z.Y.; supervision and project administration, W.Z.; funding acquisition, W.Z. and X.L. (Xiaolu Li). All authors contributed to the study design. All authors have read and agreed to the published version of the manuscript.

Funding: This study received support from the Sichuan Science and Technology Program (2023YFSY0026, 2023YFH0004) and the Sichuan Social Science Major Project (SC22ZDCY09).

Data Availability Statement: The data used in this experiment can be obtained from the following address: https://www.kaggle.com/datasets/jarvistian/sogou-news-corpus, (accessed on 1 September 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, X.; Shi, T.; Zhou, G.; Liu, M.; Yin, Z.; Yin, L.; Zheng, W. Emotion classification for short texts: An improved multi-label method. *Humanit. Soc. Sci. Commun.* **2023**, *10*, 306. [CrossRef]
2. Sebastiani, F. Machine learning in automated text categorization. *ACM Comput. Surv.* **2002**, *34*, 1–47. [CrossRef]
3. Jiang, A.-H.; Huang, X.-C.; Zhang, Z.-H.; Li, J.; Zhang, Z.-Y.; Hua, H.-X. Mutual information algorithms. *Mech. Syst. Signal Process.* **2010**, *24*, 2947–2960. [CrossRef]
4. Lancaster, H.O.; Seneta, E. Chi-Square Distribution. In *Encyclopedia of Biostatistics*; John Wiley & Sons: Hoboken, NJ, USA, 2005. [CrossRef]
5. Bai, L.; Li, H.; Gao, W.; Xie, J.; Wang, H. A joint multiobjective optimization of feature selection and classifier design for high-dimensional data classification. *Inf. Sci.* **2023**, *626*, 457–473. [CrossRef]
6. Liu, X.; Zhou, G.; Kong, M.; Yin, Z.; Li, X.; Yin, L.; Zheng, W. Developing Multi-Labelled Corpus of Twitter Short Texts: A Semi-Automatic Method. *Systems* **2023**, *11*, 390. [CrossRef]
7. Bai, R.; Wang, X.; Liao, J. Extract semantic information from wordnet to improve text classification performance. In Proceedings of the International Conference on Advanced Computer Science and Information Technology, Miyazaki, Japan, 23–25 June 2010; pp. 409–420. [CrossRef]
8. Shi, F.; Chen, L.; Han, J.; Childs, P. A data-driven text mining and semantic network analysis for design information retrieval. *J. Mech. Des.* **2017**, *139*, 111402. [CrossRef]
9. Wang, W.; Yan, Y.; Winkler, S.; Sebe, N. Category specific dictionary learning for attribute specific feature selection. *IEEE Trans. Image Process.* **2016**, *25*, 1465–1478. [CrossRef]
10. Szczepanek, R. A Deep Learning Model of Spatial Distance and Named Entity Recognition (SD-NER) for Flood Mark Text Classification. *Water* **2023**, *15*, 1197. [CrossRef]
11. Luhn, H.P. The automatic creation of literature abstracts. *IBM J. Res. Dev.* **1958**, *2*, 159–165. [CrossRef]
12. Maron, M.E.; Kuhns, J.L. On relevance, probabilistic indexing and information retrieval. *J. ACM* **1960**, *7*, 216–244. [CrossRef]
13. Maron, M.E. Automatic indexing: An experimental inquiry. *J. ACM* **1961**, *8*, 404–417. [CrossRef]
14. Salton, G.; Wong, A.; Yang, C.-S. A vector space model for automatic indexing. *Commun. ACM* **1975**, *18*, 613–620. [CrossRef]
15. Bengio, Y.; Ducharme, R.; Vincent, P. A neural probabilistic language model. In Proceedings of the 13th 2000 Neural Information Processing Systems (NIPS) Conference, Denver, CO, USA, 29 November–4 December 1999.
16. Collobert, R.; Weston, J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; pp. 160–167. [CrossRef]
17. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
18. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems, Carson City, NV, USA, 5–10 December 2013.
19. Barakat, B.K.; Seitz, A.R.; Shams, L. The effect of statistical learning on internal stimulus representations: Predictable items are enhanced even when not predicted. *Cognition* **2013**, *129*, 205–211. [CrossRef]
20. Kim, Y. Convolutional neural networks for sentence classification. *arXiv* **2014**, arXiv:1408.5882.
21. Shi, B.; Bai, X.; Yao, C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2298–2304. [CrossRef]
22. Cao, S.; Lu, W.; Zhou, J.; Li, X. cw2vec: Learning Chinese word embeddings with stroke n-gram information. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018. [CrossRef]
23. Wan, C.; Wang, Y.; Liu, Y.; Ji, J.; Feng, G. Composite feature extraction and selection for text classification. *IEEE Access* **2019**, *7*, 35208–35219. [CrossRef]

24. Zhu, M.; Yang, X. Chinese texts classification system. In Proceedings of the 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT), Kahului, HI, USA, 14–17 March 2019; pp. 149–152. [CrossRef]
25. Pan, L.; Hang, C.-W.; Sil, A.; Potdar, S. Improved text classification via contrastive adversarial training. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 22 February–1 March 2022; pp. 11130–11138. [CrossRef]
26. Zhang, M.-L.; Zhou, Z.-H. A review on multi-label learning algorithms. *IEEE Trans. Knowl. Data Eng.* **2013**, *26*, 1819–1837. [CrossRef]
27. Onan, A.; Korukoğlu, S.; Bulut, H. Ensemble of keyword extraction methods and classifiers in text classification. *Expert Syst. Appl.* **2016**, *57*, 232–247. [CrossRef]
28. Kang, M.; Ahn, J.; Lee, K. Opinion mining using ensemble text hidden Markov models for text classification. *Expert Syst. Appl.* **2018**, *94*, 218–227. [CrossRef]
29. Azam, N.; Yao, J. Comparison of term frequency and document frequency based feature selection metrics in text categorization. *Expert Syst. Appl.* **2012**, *39*, 4760–4768. [CrossRef]
30. Omuya, E.O.; Okeyo, G.O.; Kimwele, M.W. Feature Selection for Classification using Principal Component Analysis and Information Gain. *Expert Syst. Appl.* **2021**, *174*, 114765. [CrossRef]
31. Vora, S.; Yang, H. A comprehensive study of eleven feature selection algorithms and their impact on text classification. In Proceedings of the 2017 Computing Conference, London, UK, 18–20 July 2017; pp. 440–449. [CrossRef]
32. Qaiser, S.; Ali, R. Text mining: Use of TF-IDF to examine the relevance of words to documents. *Int. J. Comput. Appl.* **2018**, *181*, 25–29. [CrossRef]
33. Sun, J. Jieba Chinese Word Segmentation Tool. 2012. Available online: https://github.com/fxsjy/jieba (accessed on 1 September 2022).
34. Yao, Z.; Ze-wen, C. Research on the construction and filter method of stop-word list in text preprocessing. In Proceedings of the 2011 Fourth International Conference on Intelligent Computation Technology and Automation, Shenzhen, China, 28–29 March 2011; pp. 217–221. [CrossRef]
35. Zhang, C.; Wang, X.; Yu, S.; Wang, Y. Research on keyword extraction of Word2vec model in Chinese corpus. In Proceedings of the 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS), Singapore, 6–8 June 2018; pp. 339–343. [CrossRef]
36. Shah, F.P.; Patel, V. A review on feature selection and feature extraction for text classification. In Proceedings of the 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 23–25 March 2016; pp. 2264–2268. [CrossRef]
37. Zhai, Y.; Song, W.; Liu, X.; Liu, L.; Zhao, X. A chi-square statistics-based feature selection method in text classification. In Proceedings of the 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 23–25 November 2018; pp. 160–163. [CrossRef]
38. Liang, D.; Yi, B. Two-stage three-way enhanced technique for ensemble learning in inclusive policy text classification. *Inf. Sci.* **2021**, *547*, 271–288. [CrossRef]
39. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 14–18 August 2016; pp. 785–794. [CrossRef]
40. Sagi, O.; Rokach, L. Approximating XGBoost with an interpretable decision tree. *Inf. Sci.* **2021**, *572*, 522–542. [CrossRef]