Scholar Publishing Group
Machine Learning Theory and Practice
https://doi.org/10.38007/ML.2022.030401
ISSN 2790-0983 Vol. 3, Issue 4: 1-8

Hybrid Leapfrog Algorithm in Feature Selection Optimization of Text Classification

Shiwei Chu*

Forestry College of Beijing Forestry University, Beijing, China

chushiwei2022@bjfu.edu.cn

**corresponding author*

Keywords: Hybrid Leapfrog Algorithm, Text Classification, Feature Selection, Optimal Application

Abstract: The research of hybrid leapfrog algorithm has always been a hot spot. The performance of the classifier model can be improved by improving and optimizing the model. This paper proposes a fusion text semantic structure pattern recognition system based on hybrid leapfrog and support vector machine. The leapfrog algorithm is studied to improve the speed and accuracy of text classification feature selection. This paper mainly uses experimental design and data comparison to illustrate the performance of different algorithms in text classification. The experimental results show that MSFLA-FCM algorithm has great advantages in terms of average fitness value, the maximum fitness reaches 1.518, and the effect of text classification is significantly improved.

1. Introduction

With the development of computer technology, in terms of data processing, we have been able to classify according to user input information. This paper mainly aims at the hybrid jump algorithm (PCA) to solve the problems in text classification. At present, two classification methods based on the combination of training set and test set have been proposed to distinguish the behavior characteristics of fish schools. At present, the commonly used methods mainly include artificial neural algorithm, statistical learning and knowledge map based methods.

There are a lot of theoretical achievements on hybrid leapfrog algorithm and its application in text classification feature selection optimization. For example, some scholars said that the Hybrid Frog Leaping Algorithm (SFLA) is an intelligent optimization algorithm that imitates the foraging behavior of frog groups, and it has the advantages of few parameters, simple structure, strong robustness, etc. [1-2]. Some scholars believe that leapfrog algorithm initializes the population through reverse learning mechanism, and then uses normal cloud operator to solve the better value around the global optimal individual and sub group optimal individual [3-4]. In addition, some scholars have proposed a fuzzy C-means clustering algorithm based on mixed leapfrog [5-6]. Therefore, the application of leapfrog algorithm and its improvement in feature selection of text classification is a more novel topic and has practical value.

This paper first studies the text classification technology, analyzes its framework, and analyzes the steps of text classification. Secondly, the theory of the hybrid leapfrog algorithm is expounded, and the advantages of the algorithm are proposed. Then the application of the hybrid leapfrog algorithm in text classification is briefly discussed. Finally, through experiments, we study the role of different algorithms in text classification, and draw relevant conclusions.

2. Optimization Algorithm of Text Classification Feature Selection Based on Hybrid Leapfrog

2.1. Text Classification Technology

Text classification technology is an interdisciplinary subject. At present, text classification has been successfully applied in various fields, such as intelligent retrieval, gene analysis, digital library, etc. Understanding the general process and key steps of text classification technology will help us understand the characteristics of text classification more deeply, and at the same time, we can propose some feature selection methods more specifically. Its basic goal is to automatically classify unknown texts into corresponding categories according to the text set of known categories. The general process is as follows: analyze the text set of known categories and construct the corresponding classification model, that is, classifier. Then, text of unknown categories is automatically classified into predefined categories according to the classifier. The process of text classification is equivalent to the process of set mapping in the mathematical field. In this process, the text sets to be classified of unknown categories are mapped to the corresponding category sets one by one according to the predefined category information [7-8]. The basic framework of text classification is shown in Figure 1:

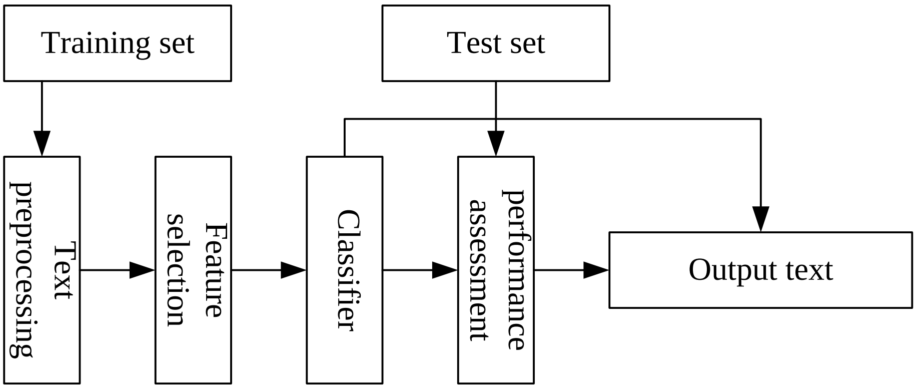

Figure 1. The basic framework for text classification

The text classification model can be divided into two processes, one is the training process the other is the classification process. In the training process, the classifier is constructed by text preprocessing, feature selection and feature weighting for the text set of known categories (training set), and the classifier is tested and evaluated using the text set of unknown categories (test set). Text preprocessing is mainly to convert text into a form that meets the requirements of the text classification system, and eliminate information irrelevant to text classification. Effective text preprocessing can make text classification fast and effective [9-10].

Text preprocessing usually includes removing stop words and text marks. Each corpus has its fixed storage format, and these tags used to describe the text format are generally irrelevant to the text content, such as numbers, punctuation marks, spaces, images and other information, and even some garbled code. These tags do not carry any useful information, do not help in the text classification process, and even affect the efficiency of text classification. Therefore, before classification, it is usually necessary to preprocess the corpus.

Text is composed of a series of strings without fixed structure. During text classification, the content of the text cannot be understood and processed by the computer. Therefore, it is necessary to extract the text from the complex text structure and convert it into a form that can be understood by the machine [11-12].

2.2. Hybrid Leapfrog Algorithm

Hybrid Frog Leaping Algorithm (SFLA) is a method that regards the frog in the population as the hidden solution of the optimization problem, and uses the cooperation and competition in the process of frog hunting to find the optimal solution in the population. The essence of this algorithm is a group mountain climbing algorithm. SFLA mainly solves the problem of multi-objective optimization. The hybrid leapfrog method is a global minimization rule search technique. It does not depend on the objective function but only on the algorithm itself. SFLA regards each frog as an individual of the population, and multiple subgroups constitute the whole population. When the individuals of the sub group are updated, it is necessary to use the best and worst individuals of the sub group to generate a new individual (as a jump). If the fitness of the parent individual is worse than that of the new individual, replace it [13-14].

In the local search stage, for each subpopulation, only the frog with the worst fitness value is updated each time during its evolution. First, under the guidance of the best frog M_x in the sub population, the update method based on local optimization M_{v} is used to update:

$$G_{\mathrm{i}} = R*(M_x - M_v), M_v' = M_v + G_{\mathrm{i}} \tag{1}$$

In the above formula, G_{i} represents the step size of leapfrog jumping, and R is a random real number evenly distributed between 0 and 1. The update method is based on the global optimization M_{v}.

$$G_{\mathrm{i}} = R*(M_x - M_v), M_v' = M_v + G_{\mathrm{i}} \tag{2}$$

The core step of SFLA implementation is to interweave global search and local search. Global search can effectively jump out of the problem of local optimal solution. Local search can make the search speed faster, and can quickly find the extreme value in the specified space. The main idea of improving SFLA is to divide the whole population into a subpopulation by randomly selecting frogs to perform grouping, and each group contains b frogs. For each group of frogs, judge whether it is the best frog by comparing the fitness of frogs in the subpopulation. Each frog in the population will be affected by another frog. Through evolution, each frog will be optimized to the best solution [15-16].

Hybrid hop algorithm is a global search and optimization problem. In the process of classification, several features are arranged in a certain order. Although this processing method can reduce the number and complexity of training sets, improve the operation speed and accuracy. The hybrid leapfrog algorithm is a new classification method based on the combinatorial optimization problem, which has the characteristics of global search ability and local optimization. This model can integrate strings, words and other information. In the training process, we convert the objective function into the sub generation edge. This method improves the solving efficiency and saves time. But there are also some defects. First of all, it cannot solve the problem of strong relationship dependency between complex data points. It cannot describe a class or a combination of classes as a whole. Secondly, the requirements for objective function parameters are high. Finally, in the classification process, a large amount of sample data is needed to support its calculation and optimize performance indicators [17-18].

2.3. Application of Hybrid Leapfrog Algorithm in Text Classification

The hybrid leapfrog algorithm classification is mainly based on the training set to classify, and a training text contains a lot of data information and corresponding feature points. This paper adopts an automatic partition rule for this model. This method can not only ensure the quality of randomly generated results, but also avoid the calculation error caused by human factors. At the same time, the least square method is used after the optimal solution is obtained through statistical analysis to meet the problem solving requirements. By satisfying the weight range under certain conditions to process and generate a classifier, the efficiency and performance of the algorithm can be improved. This paper mainly studies the application of the hybrid leapfrog algorithm in text classification, and gets a better solution by adjusting the parameters of the model. Take the trained initial value data as the test variable. Then the corresponding threshold value is calculated according to the corresponding difference function formula in the experimental results.

Based on the characteristics of the hybrid leapfrog algorithm, this paper firstly analyzes three key technologies commonly used in the field of text classification-split recognition, artificial neural network and entity analysis. Then, on the classification training module, an improved maximum entropy search optimization model is proposed, which is suitable for solving the optimal solution and minimum variance conditions. Text information is obtained through preprocessing. According to the fit function analyzed from the original data, the attribute vector values and global threshold intervals corresponding to the sample points are calculated, and the weight distribution among all categories in each cluster domain after grouping is obtained. The classification results are obtained by using statistical features for classification training. Compared with the experimental results, the hybrid leapfrog algorithm has a high similarity in the training process.

3. Text Classification Experiment Design

3.1. Experimental Platform

The algorithm used in the experiment is programmed by MATLAB 9.0, and the experimental environment is P6/3.0GHz/4.0GB PC. This experiment is carried out on a PC configured as follows.

Processor: Intel (R) Core (TM) i7
Memory: 12G
Operating system: Microsoft Windows 13 Home Edition
Development environment: PyCharm

Development language: Python implements IG, CHI, MI, FSTM and WMI algorithms

3.2. Test Data Set

The experimental data set is downloaded from the search engine. It includes five projects: finance, entertainment, sports, science and medicine. After the document is cleaned, the Chinese lexical analysis system ICTCLAS is used to extract the feature words and the weight of the feature words is calculated by tf idf. FCM, SFLA-FCM and MSFLA-FCM were used to cluster the data.

3.3. Evaluation Indicators

This paper focuses on the function of hybrid leapfrog algorithm in text classification. Therefore, this paper mainly selects the precision, recall and query time of text classification as indicators. Both recall and precision are obtained when the text is divided accurately. In this paper, FCM, SFLA-FCM and MSFLA-FCM algorithms are used to compare and obtain relevant data.

4. Analysis of Experimental Results

4.1. Performance Index Analysis

ICMFW in this paper represents IG, CHI, MI, FSTM and WMI algorithms respectively. By analyzing the precision, it can be concluded that the WMI algorithm is 1 percentage point higher than other feature selection algorithms on average. By analyzing the recall ratio, it can be concluded that WMI algorithm is the best method for feature selection. FSTM and IG algorithms are basically the same, and both are better than MI and CHI. See Table 1 for details:

Table 1. Text classification criteria datasets classify overall metrics on feature selection algorithms

	Precision ratio	Recall ratio	Test time
I	0.881	0.865	10
C	0.872	0.855	13
M	0.891	0.837	10
F	0.882	0.864	10
W	0.899	0.890	8

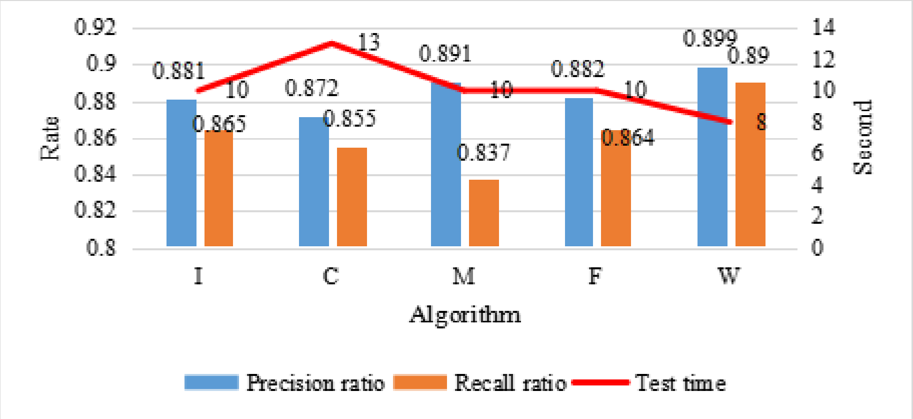

Figure 2. Text classification criteria datasets classify overall metrics on feature selection algorithms

As shown in Figure 2, we can see from the shape trend in the figure that the WMI algorithm introduces word frequency and does not completely eliminate all low-frequency words. WMI algorithm is superior to MI algorithm and FSTM algorithm, and the experiment also verifies the rationality and effectiveness of the improved algorithm.

4.2. Performance Comparison of Three Algorithms

The comparison between the optimal value of the algorithm's fitness and the algorithm's convergence time is shown in Table 2. Each algorithm is calculated 40 times in this paper, and the average value is taken. We can see from the table that each algorithm takes less than one minute. This makes the experiment time less.

Table 2. Performance comparison of the three algorithms

	Maximum fitness(10^{-5})	Average fitness(10^{-5})	Time
FCM	1.402	1.402	16
SFLA-FCM	1.486	1.473	38
MSFLA-FCM	1.518	1.511	30

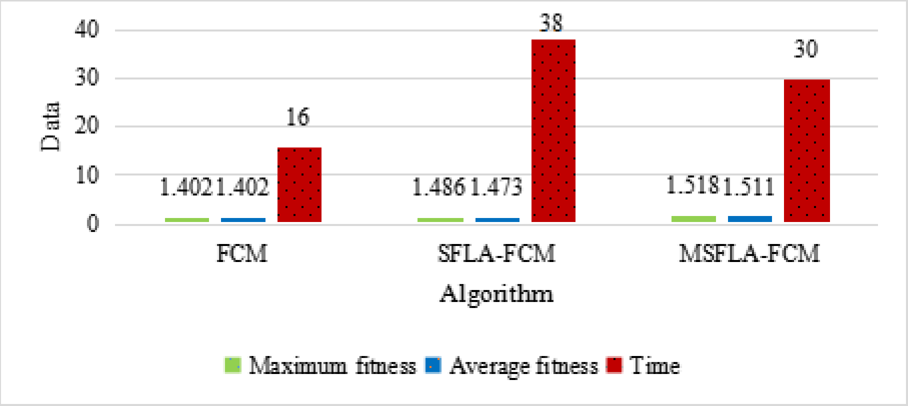

Figure 3. Performance comparison of the three algorithms

As shown in Figure 3, although MSFLA-FCM has comparative advantages over the other two algorithms in terms of clustering accuracy and the ability to find that texts belong to multiple classes, due to the introduction of the local search and global search processes of the hybrid leapfrog algorithm, the algorithm is longer than the FCM clustering algorithm in terms of time, which is also a disadvantage of the algorithm. However, compared with SFLA-FCM, the evolutionary time has been relatively shortened due to the corresponding improvement of the hybrid leapfrog algorithm.

5. Conclusion

In this paper, when studying the hybrid leapfrog algorithm, the selection of text classification features is discussed, and the corresponding theory is given. Through analysis and comparison, it is found that there are some differences between the target points and training data (including horizontal and vertical directions) based on the mixed frog jump. According to this property, it can be divided into two categories. Based on the hybrid leapfrog algorithm, the text classification problem is improved by introducing word vectors, statistical feature parameters and random search methods in machine learning. This article is affected by the incomplete content and other factors, and fails to successfully verify its other performance and application value.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Assia Belherazem, Redouane Tlemsani: Boosting Convolutional Neural Networks Using a Bidirectional Fast Gated Recurrent Unit for Text Categorization. Int. J. Artif. Intell. Mach. Learn. 12(1): 1-20 (2022). https://doi.org/10.4018/IJAIML.308815

[2] Muhammad Azeem Sarwar, Mansoor Ahmed, Asad Habib, Muhammad Khalid, M. Akhtar Ali, Mohsin Raza, Shahid Hussain, Ghufran Ahmed: Exploiting Ontology Recommendation Using Text Categorization Approach. IEEE Access 9: 27304-27322 (2021). https://doi.org/10.1109/ACCESS.2020.3047364

[3] Ankita Dhar, Himadri Mukherjee, Niladri Sekhar Dash, Kaushik Roy: Text Categorization: Past And Present. Artif. Intell. Rev. 54(4): 3007-3054 (2021). https://doi.org/10.1007/s10462-020-09919-1

[4] Emin Borandag, Akin Öz çift, Yesim Kaygusuz: Development of Majority Vote Ensemble Feature Selection Algorithm Augmented with Rank Allocation to Enhance Turkish Text Categorization. Turkish J. Electr. Eng. Comput. Sci. 29(2): 514-530 (2021). https://doi.org/10.3906/elk-1911-116

[5] V. Srilakshmi, K. Anuradha, Chigarapalle Shoba Bindu: Stochastic Gradient-CAViaR-Based Deep Belief Network for Text Categorization. Evol. Intell. 14(4): 1727-1741 (2021). https://doi.org/10.1007/s12065-020-00449-x

[6] Maximiliano Garc ía, Sebasti án Maldonado, Carla Vairetti: Efficient n-Gram Construction for Text Categorization Using Feature Selection Techniques. Intell. Data Anal. 25(3): 509-525 (2021). https://doi.org/10.3233/IDA-205154

[7] Liriam Enamoto, Li Weigang, Geraldo P. Rocha Filho: Generic Framework for Multilingual Short Text Categorization Using Convolutional Neural Network. Multim. Tools Appl. 80(9): 13475-13490 (2021). https://doi.org/10.1007/s11042-020-10314-9

[8] Walid Cherif, Abdellah Madani, Mohamed Kissi: Text Categorization Based on a New Classification by Thresholds. Prog. Artif. Intell. 10(4): 433-447 (2021). https://doi.org/10.1007/s13748-021-00247-1

[9] Mohammad Alhawarat, Ahmad O. Aseeri: A Superior Arabic Text Categorization Deep Model (SATCDM). IEEE Access 8: 24653-24661 (2020). https://doi.org/10.1109/ACCESS.2020.2970504

[10] Huda Abdulrahman Almuzaini, Aqil M. Azmi: Impact of Stemming and Word Embedding on Deep Learning-Based Arabic Text Categorization. IEEE Access 8: 127913-127928 (2020). https://doi.org/10.1109/ACCESS.2020.3009217

[11] Eniafe Festus Ayetiran: An Index-Based Joint Multilingual/Cross-Lingual Text Categorization Using Topic Expansion via BabelNet. Turkish J. Electr. Eng. Comput. Sci. 28(1): 224-237 (2020). https://doi.org/10.3906/elk-1901-140

[12] Fatima-Zahra El-Alami, Said Ouatik El Alaoui, Noureddine Ennahnahi: Deep Neural Models and Retrofitting for Arabic Text Categorization. Int. J. Intell. Inf. Technol. 16(2): 74-86 (2020). https://doi.org/10.4018/IJIIT.2020040104

[13] Edward Kai Fung Dang, Robert Wing Pong Luk, James Allan: Context-Dependent Feature Values in Text Categorization. Int. J. Softw. Eng. Knowl. Eng. 30(9): 1199-1219 (2020). https://doi.org/10.1142/S021819402050031X

[14] V. Srilakshmi, K. Anuradha, Chigarapalle Shoba Bindu: Optimized Deep Belief Network and Entropy-Based Hybrid Bounding Model for Incremental Text Categorization. Int. J. Web Inf. Syst. 16(3): 347-368 (2020). https://doi.org/10.1108/IJWIS-03-2020-0015

[15] Maciej Pachocki, Anna Wróblewska: Categorization of Persons Based on Their Mentions in Polish News Texts. J. Autom. Mob. Robotics Intell. Syst. 14(2): 42-49 (2020). https://doi.org/10.14313/JAMRIS/2-2020/19

[16] Mouhoub Belazzoug, Mohamed Touahria, Farid Nouioua, Mohammed Brahimi: An Improved Sine Cosine Algorithm to Select Features for Text Categorization. J. King Saud Univ. Comput. Inf. Sci. 32(4): 454-464 (2020). https://doi.org/10.1016/j.jksuci.2019.07.003

[17] Behzad Naderalvojoud, Ebru Akcapinar Sezer: Term evaluation metrics in imbalanced text categorization. Nat. Lang. Eng. 26(1): 31-47 (2020). https://doi.org/10.1017/S1351324919000317

[18] Mohamed Seghir Hadj Ameur, Riadh Belkebir, Ahmed Guessoum: Robust Arabic Text Categorization by Combining Convolutional and Recurrent Neural Networks. ACM Trans. Asian Low Resour. Lang. Inf. Process. 19(5): 66:1-66:16 (2020).https://doi.org/10.1145/3390092