Universe Scientific Publishing

ORIGINAL ARTICLE

Discussion on Text Classification Algorithm Based on Genetic Algorithm and Probability Theory

Ling Sun[*]

Lanzhou Resources & Environment Voc-Tech College, Lanzhou 730021, Gansu, China. E-mail: xue195@163.com

Abstract: This article mainly studies the text classification algorithm based on genetic algorithm and probability theory, and improves the speed and accuracy of text classification by using the related knowledge of genetic algorithm and probability theory. Preliminary assignment of feature items is carried out through TF algorithm, and then special non-substantial words are shielded. Using L-E operator for weighting calculation can make the better results converge faster. Using genetic algorithm, using crossover operator, mutation operator and establishing a suitable objective function, speed up the retrieval speed and improve the efficiency of obtaining the best results. Using a hybrid algorithm can eliminate the interference of synonyms and non-characteristics.

Keywords: Genetic Algorithm; Probability Theory; Text Classification; Bayesian Formula

1. Weight calculation

1.1 TF algorithm

TF represents the number of times a feature item appears in the text. The higher the frequency of occurrence, the more weight is assigned to the feature item, and the lower the frequency of occurrence, the smaller the weight is assigned. This method is simple and easy to implement, and has high efficiency. It is generally applicable to long texts and is the first commonly used method for text classification. However, in practical applications, there are sometimes low-frequency words, but they are very important to the text. If the TF algorithm is used, it will not be given a large weight, which may cause important information to be ignored.

1.2 IDF algorithm

When calculating the weight, if only the TF algorithm is used, the forbidden words will interfere with the feature extraction. The IDF algorithm can reduce the influence of forbidden words on feature item extraction. The IDF algorithm is an inverted sorting algorithm, which calculates the frequency of a feature item in the entire article collection. The more times it appears, the lower its distinguishing effect on similar texts. If a feature item is contained in all texts, its weight is zero. The IDF algorithm believes that the feature items that appear in a large number of texts are less important than the feature items that only appear in a few texts. Therefore, the IDF algorithm can help extract more important and accurate feature items from the text.

1.3 Word frequency statistics and L-E linear index weighting factor

When performing word frequency statistics, count the number of occurrences of each feature item in each paragraph and title, and mark the feature items that appear in the title and the beginning and end of the paragraph and count the number of times. Next, the keywords that are evenly distributed in the article are weighted. The standard for this uniform distribution should be that the keywords appearing in each natural segment are similar in frequency and there is no big fluctuation. Use L-E linear exponential weighting factor for analysis:

$$T = n_1 \times T_1 + n_2 \times T_2 + n_3 \times T_3 + n_4 \times T_4$$

doi: 10.18686/ahe.v5i2.3348

In the formula, T represents the total number of times the keyword appears in the article. T_1 is the total number of times the keyword appears in the article. T_2 is the number of times the keyword appears in the title. T_3 is the number of times the keyword appears in the paragraph. The number of occurrences at the beginning and the end of the paragraph. T_4 is the number of occurrences of the feature item in the paragraph of the entire article. The value of n_1 is determined by the length of the article, the shorter the article, the greater its value. The value of n_2 is determined by the ratio of the title to the length of the article, the smaller the ratio, the greater its value. the value of n_3 is the number of paragraphs and words of the article how much is determined. The value of n_4 is determined by the convergence speed of the widely distributed feature items.

1.4 Total probability formula and Bayesian formula

Definition Let S be the sample space of experiment E , and $B_1, B_2, B_3, \cdots, B_n$ be a set of events of E. If, $B_iB_j = \Phi, i \neq j, i, j = 1,2, \cdots, n$, $B_1 \cup B_2 \cup \cdots \cup B_n = S$, then $B_1, B_2, \cdots, B_n$ are called a division of the sample space.

Theorem suppose the sample space of test E is, A is the event of E , $B_1, B_2, \cdots, B_n$ is a division of E, and, then$P(B_i) > 0 (i = 1,2, \cdots, n)$, $P(A) = P(A \mid B_1)P(B_1) + P(A \mid B_2)P(B_2) + \cdots + P(A \mid B_n)P(B_n)$ is called the total probability formula.

Theorem Suppose the sample space of test E is S , A is the event of E , and $B_1, B_2, \cdots, B_n$ is a division of E , then

$$P(B_i \mid A) = \frac{P(A \mid B_i)P(B_i)}{\sum_{j=1}^{n} P(B_j)P(A \mid B_j)} (i = 1,2, \cdots, n)$$

This is called the Bayesian formula. Explanation: i and j are all subscripts, and the sum is 1 to n. The genetic algorithm has four parameters that need to be explained and calculated in advance, which are population size, mutation probability, crossover probability, and number of iterations. Among them, when calculating the crossover probability and the mutation probability, the relevant knowledge of probability theory can be used to solve the problem.

2. Feature item extraction based on parallel genetic algorithm

2.1 Binary encoding

Make each bit in the chromosome correspond to a word in the feature word set. When the feature word is extracted, its corresponding chromosome position is 1, and when it is not extracted, it is 0. Using this binary code for genetic algorithm can ensure that even if the set of feature words is large, it will not take up too much space.

2.2 Genetic operators

The selection operator is used to pair the current population to breed the next generation. The selection operator should conform to the principle of randomness and follow the genetic law of objective organisms. Secondly, crossover operator and mutation operator are carried out.

2.3 Fitness function

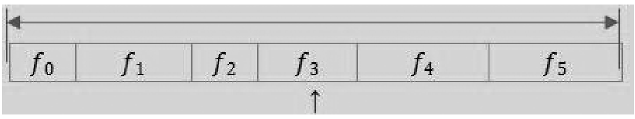
Euclidean distance is used to measure the similarity of text. One of the most important is to calculate the expression of text similarity between chromosomes formed by different keywords. Use the following methods to solve the calculation problem:

(1) Expand the feature items extracted by the two chromosomes, and obtain their intersection, so that the dimension of the text vector represented by them is consistent.

(2) Use these two chromosomes to represent the same text respectively. Pay special attention to the text representation. For a chromosome, if a keyword is not extracted in the chromosome, then the text vector it represents corresponds to 0. At the same time, in order to prevent chromosomes from aberrations, the chromosomes are set to the recessive threshold of traits. The trait at position 1 in the chromosomal gene sequence is set as a dominant trait, and the position at 0 is a recessive trait. In chromosomes, neither dominant traits nor recessive traits can be less than 35% . This can effectively prevent chromosome aberrations.

The fitness function can help us make choices in genetic algorithms. For example, when selecting individuals in a population, we can usually use fitness ratios for selection. The working principle is as follows.

$$\text{Overall fitness value } F = \sum_{i=0}^{N-1} f_i$$



$r \in [0,F)$

The uniformly distributed random numberr $\in [0, F)$ specifies which individual to choose through the following parameter minimization:

$$i \leftarrow \underset{n\in[0,N)}{\operatorname{argmin}}\left\{r < \sum_{i=0}^{n} f_i\right\}$$

Where N is the number of individuals, and f_i is the fitness value of the i -th individual.

2.4 Probability selection

Probability selection is a variant of fitness function selection. The working principle of fitness ratio selection has been mentioned above. The working principle of probability selection is roughly similar to the principle of fitness ratio selection, but the probability selection and fitness ratio selection are different. One point is that there is no need to sort the population, and the selection probability of individual i obeys the binomial distribution:

$$P(i,k) = \binom{n}{k} P(i)^{k}(1-P(i))^{n-k}$$

Where n is the size of the population, and k is the number of individuals that need to be selected from the population. The running complexity of the implemented probability selector is $O(n + \log(n))$ instead of $O(n_2)$, due to the naive method: binary (index) search is performed on the summation probability array.

3. Perform text classification

After excluding the influence of synonyms in the text, the final selected feature items are assigned to the text. The final output data includes the read content and main words of each paragraph of the article; keywords and the frequency of their appearance in the title, beginning and end of the paragraph; the paragraphs of the feature item distribution and the number of occurrences; the probability of the keywords appearing after weighting; the feature vector of the reference; through the genetic cross-mutation processing of the chromosome, the judgment of the text classification.

4. Conclusion

This article mainly discusses text classification algorithms based on genetic algorithm and probability theory, and analyzes text preprocessing, feature item extraction, feature item extraction based on parallel genetic algorithm, and text classification. On the basis of certain theoretical and practical achievements in the field of text classification, this paper studies the use of genetic algorithms and knowledge of probability theory to extract and extract text feature items, which can make text classification more efficient. In today's society, a large amount of information is constantly emerging. Research on text classification algorithms can improve the speed and efficiency of text classification, accelerate the speed of information retrieval, and better meet people's needs. It is hoped that this article can enrich the related theoretical results and provide a certain reference for related theoretical and practical research.

References

1. Zheng G. The law of algorithm and the algorithm of law. China Law Review 2018; (02): 66-85.
2. David Belot. Probability Graphic Model. Beijing: People's Posts and Telecommunications Press; 2018.
3. Li B. The practical role of machine learning. Beijing: People's Posts and Telecommunications Press; 2017.
4. Li T, Du F. Research on fuzzy portfolio selection model based on background risk. Yinchuan: Ningxia Sunshine Publishing House; 2016.