A Method for Massive Scientific Literature Clustering Based on Hadoop

Beijing University of Posts and Telecommunications

WenChuan Yang, Qiuhan Zhao, Rui Hua

Beijing University of Posts and Telecommunications

Abstract

With the development of science and technology and a large numbers of advanced vocabularies, the traditional classification of disciplines cannot meet the current needs of the subject division of scientific literature. At the same time, the clustering of the scientific literature put forward more requirements to the efficiency of the methods and the corresponding software and hardware facilities. In this paper, text features are extracted based on the TF-IDF method and the features of scientific literature. In Hadoop distributed environment, text clustering is carried out through Canopy-Kmeans algorithm, which achieved clustering of the massive scientific literature. As a result, our method proposed in this paper has improved key indicators compared to previous algorithms and greatly improved the efficiency of clustering.

Methodology

The method section is mainly divided into three parts. The first part describes how to extract the feature of Chinese scientific literature. The second part introduces the clustering algorithm used in this paper. The third part carries on the further information extraction to the cluster result, and extracts the summary for each classification.

1 Scientific Literature Features

Nouns are important for Chinese text clustering, especially in scientific literature. Formal scientific literature has information such as titles, abstracts and keywords. The information plays a very important role in the clustering of texts. The information contained in titles is a summary of the entire literature.

Moreover, keywords of the literature can generally represent the features of a scientific literature. However, keywords which are used to represent the features of the text is also limited. Although there are two pieces of literature with similar content, different keyword extractions cannot be clustered into the same cluster though. So, it is better to extract features by combining keywords and content.

2 Clustering Algorithm

In this paper, we use Canopy-Kmeans algorithm. First, we use canopy algorithm to initialize k value. This process can be seen as a quick and easy classification. According to the initial k, we propose k-means algorithm to optimize and complete the final clustering. Additionally, we implement the distribution of Canopy-Kmeans algorithm through Hadoop as shown in Fig 1.

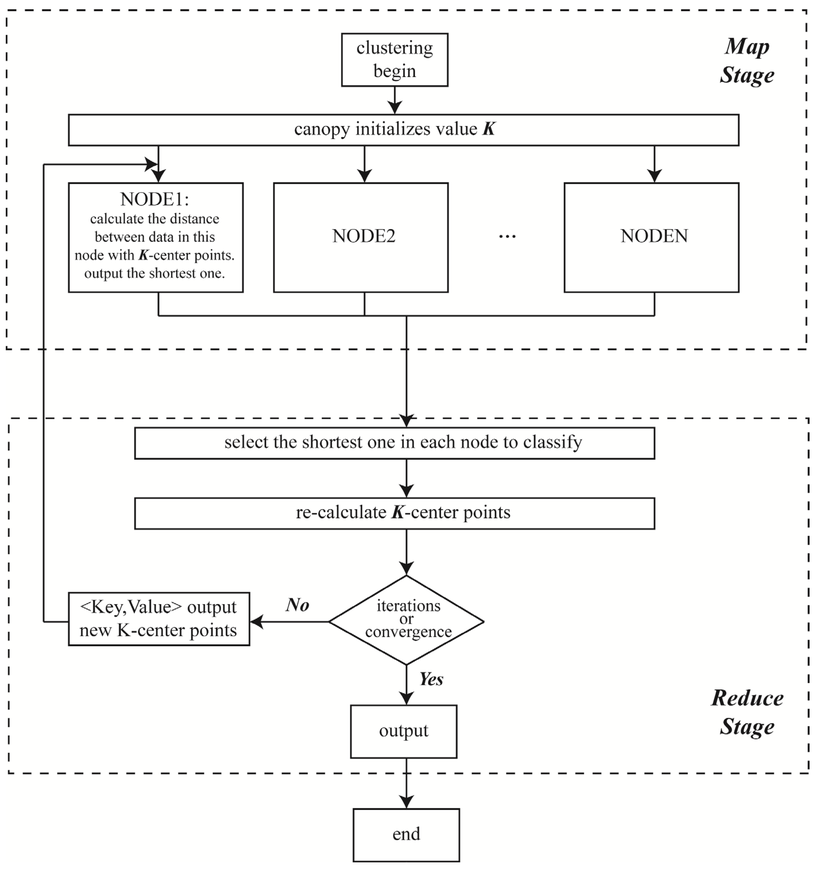

Fig. 1. MapReduce Structure Canopy k-means

3 Clustering Result Information Extraction

After obtaining the clustering results of scientific literature, forming a muti-text summary for each class will enable us to obtain more useful information. Through the study of text structure of scientific literature and Chinese expression habits, this paper proposes a statistics-based summary generation method. Specifically, the Chinese text sentence weights are assigned according to the following table.

So, we can perform distributed extraction of the text in the same class after clustering. For all scientifc literature in each class, after we score sentences, we sort the sentences according to the sum of weight, and extract the few sentences with the highest weight to generate a muti-text summary for a class.

TABLE I. WEIGHTS AND EXPLANATIONS OF PARAMETERS

Parameters	Explanation	Weight
Title Feature	The more words contained in the title, the higher the weight. Each contains more than one headline word, and the titleFeature increases by 1.	1.5
Keyword Feature	TF-IDF can be used to score the feature words contained in the article. The keywordFeature is a sum of the weight score. The	0.5
Sentence Length	We set the sentence length of 20 is an ideal value. If length of sentence is greater than 20, the weight is decreased.	1.0
Sentence Position	After formatting the paragraphs, 10 different weights from 0.03 to 0.18 are set according to the sentence position.	2.0

Experiments

For the data in this paper, we have calculated the acceleration ratios of Cnaopy-kmeans and K-means algorithm. It can seen from Fig 2, as the number of nodes increases, the acceleration ratio of different algorithms also gradually increases. Since Canopy-Kmeans performs initial coarse clustering to initialize the K value, it is more efficient than k-means which random the K value.

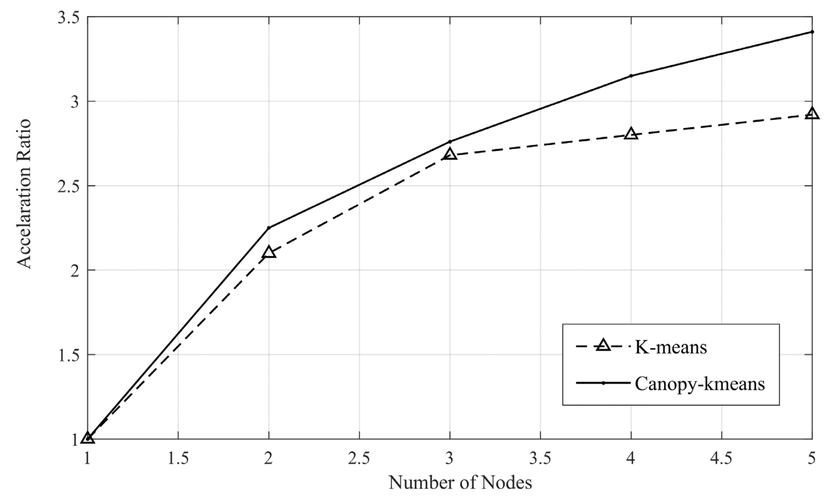

Fig. 2. Acceleration Ratio

TABLE II. COMPARISON OF ALGORITHM RESULTS

Evaluation	This paper	Bayes	KNN
Recall %	82.53	78.16	35.42
Precision %	83.94	74.57	78.80
F1 value	82.72	76.43	62.14
Running time	4273s	16232s	2332s

From the above table, it can be seen that the algorithm adopted by this paper is excellent in most of indicators. Generally, the method in this paper has achieved good results in terms of accuracy and efficiency for massive scientific literatures. And more, extraction of multiple text abstracts from a class of scientific literatures, it greatly improves the efficiency of getting information.

Conclusion

In this paper, we raise a method to cluster scientific literature and extract multiple text abstracts. The method combined features of scientific literature and TF-IDF has a certain improvement in most of indicators over some previous algorithms, and efficiency is also impressive through Hadoop.

We introduced in detail the implementation of automatic clustering of massive scientific and technical literature through this paper. According to the order of operations, we obtained modules from sample collection module, the preprocessing module, the feature generation module, the clustering module and the information extraction module. The implementation process is described in detail, including the technical details of designing the system, the configuration of the environment, and the setting of algorithm parameters. In addition, the preliminary results obtained by each module are displayed, and statistical analysis is performed on the operating efficiency of the system's algorithm. Finally, information extraction module extract summaries from clustering results for every classification.