Parallel Classification Method of Shapelet for Large-scale Time Series

Cao Yang[1]

China University of Mining and Technology, China

[1]caoy@cumt.edu.cn

Abstract—Time series shapelet are subsequences of time series that can maximally represent a class, the quality of shapelet set is the key of time series classification algorithms based on shapelet, which have high accuracy and good interpretability mostly. However, due to the large number of shapelet candidate sets that need to be traversed in the process of calculating shapelet, its drawback of high time complexity makes it difficult to use traditional methods in large-scale data sets. Therefore, in order to be able to improve shapelet algorithm on the time complexity of large-scale data set and reservation of calculation accuracy meanwhile, in this paper, a new method was suggested to change the traditional shapelet algorithm with parallel computing, through the combination of clustering and sampling method, making the large time series data set into several small samples. Then get their candidate set of shapelet by parallel computing, finally through the merge algorithm candidates were calculated according to the original data set into the most discrimination shapelet collections. Fifteen large scale UCR data sets are selected in the experiment to verify the algorithm. Through comparison experiments, it can be shown that this method can greatly reduce the training time on most time series data sets, and effectively improve the classification accuracy of the time series classification algorithm based on shapelet.

Keywords—Time series classification; Shapelet; Large Time Series; Parallel computing

I.MOTIVATION

Time series data widely exist in every aspect of production and life. It can reflect the internal running state of things by a sequence of data that changes with fixed time. Different from the traditional classification problem, the data points of the time series are in order relationship. The data state at a certain moment has little effect on the time series classification result, and Accurate extraction of the features of time series segments and selection of the corresponding classifier according to the features are the key to determine the time series classification effect.

Shapelet was first proposed by Eamonn Keogh et al., it is a subsequence pattern that can distinguish different time series categories to the greatest extent. By another word, a feature based on the local shape of the time series sequence [1].Shapelet's superiority in time series classification lies in two aspects :(1) faster classification. Shapelet classification algorithms compare subsequences and are more efficient than algorithms that compare entire sequences.(2) it's explicable. Shapelet embodies the differences between categories. Figure 1 is an example of shapelet that described the effect of gun-point [2] data set on classification by using shapelet. Figure 1-a and 1-b respectively represent the corresponding time series of gun lifting and no gun lifting. The shapelet sequence fragment in figure 1-b can clearly illustrate the difference between the two actions.

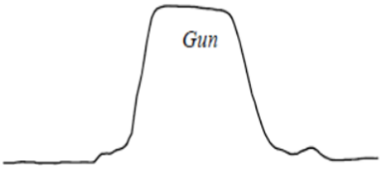

Figure 1-a the gun position of time series

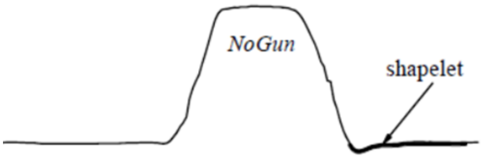

Figure 1-b the no gun position of time series

Traditional shapelet algorithm uses the brute force calculation strategy to obtain the candidate set of shapelets, and measures representativeness in all of the time series subsequence in the candidate set, this method can obtain a relatively superior classification accuracy. Classification methods based on shapelet can be divided into three categories: Shapelet Discovery Algorithms that embed shapelet Discovery Algorithms into binary decision trees [3], Shapelet Transform Algorithms that separate shapelet extraction from classifier construction [4][5] and Shapelet Learning Algorithms that acquire shapelet through machine learning theory [6] (shapelet learning Algorithms).

However, since all subsequences in the data set are evaluated as shapelet candidate sets, the computational complexity of shapelet is high (O (n^2m^4)), where n represents the number of sequences and m represents the length of sequences. It makes it difficult to apply shapelet directly to practical problems when it was just proposed. Therefore, methods to improve the computational efficiency of shapelet have attracted extensive attention from researchers. Ye and Keogh is the first that put forward two accelerated strategies, Subsequence Distance calculate before abandoning strategy (Subsequence short Early Abandon) and allow the Entropy of the Pruning strategy (Admissible Entropy Pruning [1]), the former on shapelet Distance calculation, when the current Distance is greater than the minimum Distance threshold, give up the calculation process; and latter bound of information gain is obtained by simple prediction calculation, then a part of candidate set of shapelet is cut in advance. These two methods can steadily accelerate the search speed of shapelet without reducing the accuracy, and they are two orders of magnitude faster than the brute force search strategy.

Fast shapelet selection method based on SAX symbolization was proposed to accelerates shapelet selection from the perspective of sequence data representation. It converts each time series instance into SAX word segment [3], greatly simplifying the subsequence search process of time series. Then perform a random mask progress to find the top-k SAX words with the highest information gain on this basis, finally restore the original sequence to obtain the final shapelet set. This approach reduces the complexity to O(nm^2).However, these methods to improve the acquisition efficiency of shapelet are still difficult to achieve satisfactory classification effect on large-scale data sets. Based on above-mentioned background, this article is based on mapreduce computing architecture, put parallel computing design into the traditional shapelet algorithm through the combination of clustering and resampling method[7], making the original large time series data set into small samples, each small sample will be get their candidate set of shapelet by parallel computing on cluster, and through merging algorithm it is calculated according to the original data set the most discrimination shapelet collection.

In experiment part, 15 large-scale UCR data sets are selected to verify the proposed algorithm, and the experimental conclusion shows the effectiveness of this algorithm.

II. RELATION WORKS

2.1 shapelet accelerated search strategy

Here, we divide shapelet accelerated search strategy into three categories. Meanwhile, because shapelets is a feature extraction method, it needs to be combined with classification algorithm to realize the classification of time series. In this paper, ST-HESCA, a heterogeneous integrated classification method with the best performance, is also summarized.

2.1.1Optimized search strategy The literature [8] proposed Pruning Shapelets with Key Points (Pruning Shapelets with Key Points, PSKP). PSKP firstly found the Key Points in the time series according to the standard deviation generated by each change of data Points in the time series, then extracted candidate Shapelet with these Key Points, and finally classified the time series with decision tree based on optimal Shapelet. In literature [9], during the initial screening of shapelet, candidate sets of shapelet with different endpoints from the same starting point were put together to reduce repeated operations by sharing distance information. However, Mueen[5] et al. designed a matrix to cache the distance calculation results to realize the reuse of distance value, and filtered out some candidate shapelet through triangle inequality.

2.1.2 simplify sequence morphology In Literature [10] shapelet candidate is discovered in the word space of sequence segmentation polymerization approximation (piecewise aggregate approximation, PAA), PAA here is a piecewise average characteristics of processing method. on this basis, a method was designed to get the calculation of the optimal shape, finally processed logistic regression to adjust shapelet learning model. after fast shapelet method based on SAX was proposed, another symbolic method was put forward to solve the problem of shapelet search, SFA method [11] replaced SAX with Fourier transform and proved that it was more robust to noise samples

2.1.3 narrow down the search scope In literature [11] time series were extracted from the training data set, and use the furthest subclass segmentation method to determine the sampling time sequence local deviation point (the local farthest deviation points, LFDPs), and choose between the two adjacent LFDPs subsequence as candidate shapelet. In this way, the number of candidate shapelet is greatly reduced, which significantly reduces the time complexity. Literature [13] proposes a random model generation algorithm based on shapelet classification, which generates shapelet classification tree through rapid sampling, so that only a very small part of shapelet space can be evaluated to produce a high-precision classification model in a short time.

These methods reduce the time complexity of finding shapelet in various aspects. However, the acquisition algorithm based on shapelet still needs long training time on large-scale data sets.

2.2 heterogeneous ensemble classification method ST-HESCA based on shapelet transformation

Shapelet is a feature extraction method, which should be combined with the classification algorithm to realize the classification of time series. For HESCA benchmark Classification algorithm is adopted in The experiment in this paper, The Heterogeneous Ensembles of Standard Classification Algorithms, The proposed in [14], compared with The traditional integrated Classification method, this method paid more attention to The diversity of The base classifiers, in order to achieve this purpose it USES The composition of The integration of Heterogeneous model, Heterogeneous integration approach stronger generalization ability than The original integration methods, it is currently The best overall Classification in time series Classification effect of integrated Classification strategy, The following is a brief introduction about its algorithm composition.

HESCA is composed of eight classifiers, two of which themselves are integrated classification algorithms: random forest (500 trees) and rotating forest (50 trees), respectively, and the remaining six are k-nearest neighbors, naive bayes, C4.5 decision trees, support vector machines with linear and polynomial basis function kernel, and bayesian networks. In this way, probabilistic, instance-based and tree-based classifiers are included to ensure the generalization ability of the ensemble classifier as a whole. The accuracy estimate of each constituent algorithm was obtained by 10-fold CV training, which was used as a weighted index for the later prediction test set category. In the literature [15], Bagnall and Lines for a contrast experiment of classification algorithm, the current mainstream time series experiment data set sampling UCR standard time series data, win on precision, respectively is the HIVE - COTE [16], the FLAT - COTE [17] and shapelet transform algorithm, shapelet transformation of these algorithms is shapelet transform implemented HESCA (ST - HESCA), at the same time this method is also used in HIVE - COTE, Flat-cote and become the core algorithm of shapelet classification module.

The experimental comparison in this paper is ST - HESCA algorithm, because it is currently shapelet classification method of the highest average precision. in order to ensure the integrity of their shapelet set, in shapelet calculation section, ST - HESCA USES the method of literature [18], it used one vs all coding scheme to simplify the evaluation calculation, through more frequent early give up strategy to speed up the execution procedure, and improve the precision of multi-class classfication. This makes the process more efficient than brute force shapelet calculation, but overall, the efficiency is still too slow to be used to process large-scale time series data sets.

In this paper, parallel computing is used to relieve the computing pressure of finding shapelet. Hadoop, a widely used distributed computing platform, is considered to be used in this paper. It is usually used to form a cluster on servers or cheap PCs to process batch computing for large-scale data. At the same time, the spark cluster computing environment was not selected for this article because there is less need for iterative computation. The key to parallel design of traditional shapelet computing methods lies in: how to realize reasonable partition of the original data set, generate multiple small sample sets that retain the original data distribution, so as to realize independent parallel computing of shapelet subset, and on this basis combine the results of the subset to obtain the global shapelet optimal set. Based on the above ideas, the method in this paper includes three main steps: 1. Cluster and random sampling the original data set to construct a small sample set on each calculation node; 2. Using Mapreduce parallel computing strategy, calculate shapelet candidate on each sample set and restore shapelet to the corresponding cluster according to the cluster mark of subsequence, and remove the redundancy of shapelet in each cluster to obtain the final optimal solution based on the whole.3. Transform and classify the test data according to the obtained optimal set of shapelet. Next, the specific content of the algorithm will be introduced.

III. RELATED DEFINITIONS AND SYMBOL REPRESENTATION

Definition 1.time series and subsequence:

The length of the time series T is *n*, and the length of its sub-sequence *S* is *l*. here l<n. T can be represented as $t_1,t_2,\ldots,t_{n.}$ and $S=t_i,t_{i+1},\ldots t_{i+l-1}$.here $1 \leq i \leq m+l-1$. Every two data points in the time series have the same time interval

Definition 2.distance between time series:

For two time series of equal length X and Y, let their length be m, then on the premise of using Euclidean distance as the metric standard, the distance calculation formula of two sequences is

$$\mathrm{dist}(\mathrm{x},\mathrm{y})=\sqrt{\frac{1}{\mathrm{m}}\sum_{i=1}^{m}(\mathrm{x_i}-\mathrm{y_i})^2} \quad (1)$$

The calculation process of Shapelet involves the calculation of the distance between unequal time series. Here, the shortest distance calculated is defined as the distance of subsequence. The realization process is to slide the short sequence on the long sequence.

$$\mathrm{sub}dist(x,y)=\min(dist(x,y_{|x|})) \quad (2)$$

Here, the long sequence is *y*, the short sequence is *x*, and y $_{|x|}$ represents the distance between the length of *y* sequence corresponding to each slide of *x* and the subsequence of equal length *x*, which is actually the minimum distance between the two sequences

Definition 3.information gain IG

Given a splitting strategy, a data set D is divided into two independent subsets D_1 and D_2, and the entropy before and after splitting is E(D) and $E(\hat{D})$, respectively. $E(\hat{D})$ is obtained from the weighted average entropy of each subset.

Definition 4.shapelet

Shapelet is a tuples (s, τ) made up by time series sequence *s* and split threshold τ. τ represents the threshold of distance. Shapelet splits data set D into two disjointed subsets, including:

$$D_{left}=\{x:x\in D, subdist(s,x)\le d_{th}\} \quad (3)$$

$$D_{right}=\{x:x\in D, subdist(s,x)> d_{th}\} \quad (4)$$

set $N_1=|D_{left}|$, $N_2=|D_{right}|$, The information gain of a shapelet is

$$I(s,d_{th})=E(D)-\hat{E}(D)=E(D)-(\frac{N_1}{N}E(D_{left})+\frac{N_2}{N}E(D_{right})) \quad (5)$$

Where, E(D) is the information entropy of data set D, and N is the number of time series samples in data set D.

IV. PARALLEL CLASSIFICATION METHOD OF SHAPELET

This method for large-scale time series of parallel shapelet classification algorithm includes three stages: pretreatment, Mapreduce parallel computing and classification.

4.1 preprocessing stage

(1) Cluster the original training data: Through clustering, the sequences with high similarity degree in the original time series data set can be grouped, and the similar sequences in each cluster can generate the collection of shapelets with similar distinction ability. therefore, Obtaining data samples from different clusters and assigning them to different calculation nodes to extract the collection of shapelets, which can effectively evaluate the classification ability of such sequences for small sample sets. literature [19] verified the feasibility of k-means algorithm using Euclidean distance measurement in time series clustering and can provide an efficient and relatively accurate clustering result .

(2) Random sampling constructs small sample sets: Each cluster was randomly sampled with replacement, and the results after each cluster sampling are integrated and constructed into small

sample sets, Given that such an operation may miss parts of the sequence containing highly differentiated shapelet, random sampling will be performed multiple times. the small sample collection will be the input data for parallel computing. The pseudo-code of this stage is shown in algorithm 1

algorithm 1 *Preprocess()*

```
Input: k-value of k-means, Sampling frequency n, sampling rate P%, path of training data train_url.
Output: N sampled data files。
01)Instancestrain = newInstances(train_url)
02)SimpleKMeans KM = new SimpleKMeans(k)
03)KM.buildClusterer(train)
04)Forl=0 to n
05)     For m=0 to k
06)         New Instances temp
07)         Temp .add(RandomSample(KM.getcluster(m),p))
08)     end For
09)    File file = new File(train_url+l+"aftersample.arff");
10)    BufferedWriter.write(temp.toString())
11)End for
12)End
```

Algorithm 1 is a preprocessing phase of the data processing procedure. First, read the training dataset to k means clustering (Lines 01-03), then make a sampling on each cluster, this cluster random sampling results are stored in the temporary object instance *temp*. every time after sampling procedure, create a new data sample in the original path, numbering them in order of sampling sequence, After *n* times of sampling procedure, this preprocessing process is finished. The need of time complexity in this process can be neglected

4.2Parallel computing stage on Mapreduce

Process shapelet calculation on multiple small samples, and all obtained shapelet candidate set are restored to the original data set after removing redundant. Last they are screened out to be high discrimination shapelet set, this part is the core of suggested method, It is also the most computationally intensive part of the whole method. we put this step into the map/reduce on the distributed data processing platform, each small sample as an independent data files uploaded to the HDFS, These samples are processed and summarized by parallel shapelet computation to obtain the shapelet candidate sets.

Map phase

In reprocessing phase, multiple small samples are uploaded to the HDFS data in the form of files, HDFS block mechanism will make each file into a data block (when the size of sample data is less than the default block size 128m). in the map phase, HDFS generate the number of data slices consistent with the number of files, and then system will make maptask distribution to each node on compute cluster, finally These maptasks are mapped to different data nodes, thus reached the algorithm of parallel computing, and the compute of different nodes do not interfere with each other. The input data of each map here is < file name and file content in the form of <key,value>. The map function completes shapelet query on sequence data of small samples, outputs local shapelet collections of each small sample, and stores them into the Context object in the form of multiple <shapeletcontent ,IG value>

Reduce phase

In reduce phase age, final shapelet candidates are summarized from shapelet candidates of all small samples. After de-redundancy operation, they are classified according to the cluster origin of the corresponding sequence of shapelet, and rearranged in each cluster according to the IG value of shapelet. The specific codes for the global optimal solution of shapelet are shown in algorithm 2 and algorithm 3

```
algorithm 2Map(Textflie_name, Textfile_content, Context context)
Input: <flie_name,file_content>
Output: < shapelet_content,IG_value>Context
Begin:
01) Instances train_sample=ClassifierTools.loadData(flie name)
02) New ShapeletsMap
03)    For all Instance in train_sample do
04)       seriesShapelets =SearchAllShapelets(Instance)
05)       seriesShapelets=seriesShapelets.removeSelfSimilar()
06)       ShapeletsMap.add(seriesShapelets)
07)    End for
08) ShapeletsMap.sortbyclass()
09) ShapeletsMap.removeExcess()
10) SampleShapelets=getShapeletsformmap(ShapeletsMap)
11)    For all Shapelet in SampleShapelets do
12)       Comtext.write(Shapelet.content,Shapelet.qualityValue)
13)    End for
14) End
```

In algorithm 2, the first line reads the file sequence, where *Instances* object is the sequence data set, and the *subsequent* Instance object is each sequence Instance in the data set. The second line creates a new map object to store shapelet according to the class mapping relationship. The two functions in line 4-5 respectively carry out all possible shapelet search on each sequence instance and remove redundant shapelet operations. Line 8-14 indicates that the final shapelet obtained from each instance is put into the *shapeletmap*. After the local optimal shapelet selection, the local optimal solution is loaded into the *Comtext* object for map function output.

```
algorithm 3Reduce (Text shapelet, Text IG_value, Context context)
Input: < shapelet Key, IG value>Context
Output:< shapelet Key, IG value> Context
 Begin:
01) String Train_name=Key.getserisename()
02) New ShapeletsMap
03) for all key and value do
04) ShapeletsMap.add(key,value)
05) End for
06) ShapeletsMap.setseries(Train_name)
07) ShapeletsMap.sortbyclass()
08) ShapeletsMap.removeExcess()
09) FinalShapelets=getShapeletsformmap(ShapeletsMap)
10) For all Shapelet in FinalShapelets do
11) Comtext.write(Shapelet.content, Shapelet.qualityValue)
12) End for
13) End
```

In algorithm 3, line2 create a new store shapelet *ShapeletsMap* map object mapping relations according to the class, get the name of the training data in the first line, all < key, value > Context are written to the map object (line 03-05), organize the results of all small samples and restore them into the original cluster according to the cluster mark, then sort and remove redundancy(line 06-08),finally execute Reduce function and retain the output on HDFS(line09-13).

4.3 classification phase

Finally, we set the local application to standby until the distributed application completes, then the local program starts and reads the shapelet data set calculated by the mapreduce program on HDFS, then the local program shapelet transforms the original training data, in the end, The transformed data set is handed to HESCA algorithm for classification and the final classification result is obtained, the final classification results, This also represents the end of the whole classification process of the suggested method.

V. EXPERIMENTS AND RESULTS

5. 1 experimental environment

The experimental platform is a cluster composed of 8 virtual institutions. The CPU model is Inter(R)core(TM) i7-8750h CPU @2.20ghz, the memory is 24G, and the hadoop version is 3.0.2

5. 2 Data set and parameter determination

Experimental data were obtained from 15 data sets in the common data set of UCR time series as experimental objects. The length of data set size and class number sequence is shown in table 1

Table 1 The details of experimental data

No	Data set name	Train data size	Series length	Class numble	Tesstdata size
1	car	60	577	4	60
2	ChlorineConcentration	467	166	3	3840
3	Computers	250	720	2	250
4	DistalPhalanxOutlineCorrect	600	80	2	276
5	ElectricDeviceOn	639	360	2	369
6	Ham	109	431	2	105
7	Herring	64	512	2	64
8	Meat	60	448	3	60
9	Plane	105	144	7	105
10	ProximalPhalanxOutlineCorrect	600	80	2	291
11	ProximalPhalanxOutlineAgeGroup	400	80	3	205
12	SLeaf5	485	128	2	640
13	SwedishLeaf	500	128	15	625
14	SyntheticControl	300	60	6	300
15	Trace	100	275	4	100

The parameters used in this method are mainly in the pre-processing stage, including the sampling rate *p*, the number of clusters *k* and the number of samples *n*. Here, the first step of pre-processing is to cluster the original training data according to the sequence of pre-processing. The experiment in this paper uses k-means, but other clustering algorithms are also applicable. in the description of the preprocessing stage we can find that the number of small samples obtained by the final pretreatment is only related to the sampling rate *P*, and The total amount of data submitted to the distributed system for processing is (*train_num*p*n*), Parallel processing (train_num*p) of time series instances can be carried out in the case that the number of calculated nodes do not exceed *n*. The maximum operation efficiency can be obtained when the sampling number n is less than the number of operation nodes in the cluster. *P* value has a significant impact on the size of multiple small samples obtained. The appropriate sampling rate should reduce redundant samples and ensure that the sample group after multiple sampling has the ability to represent the original distribution structure of training data. The selection of k value will affect the effect of clustering, which is the characteristic of k-means algorithm, Whether k value will affect the final shapelet set quality after a series of processing needs to be determined experimentally.Under such background, Five groups of typical time series data were selected from the experimental data for parameter testing, and the representative in terms of sequence length, training set size and the number of classes .The results are shown in figures 2 and 3

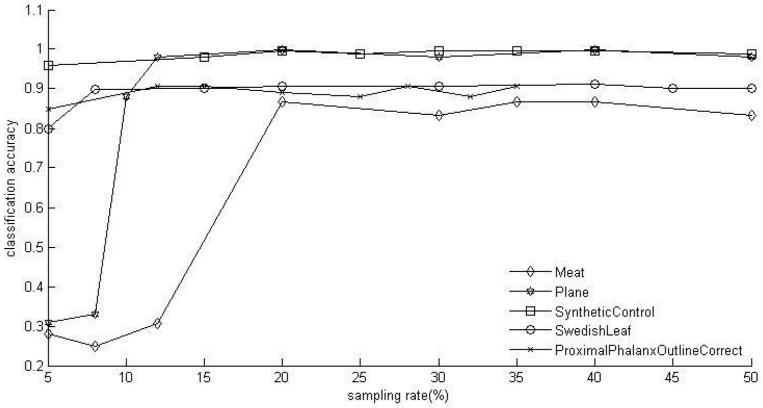

Figure 2.Influence of sampling rate on classification accuracy in different data sets

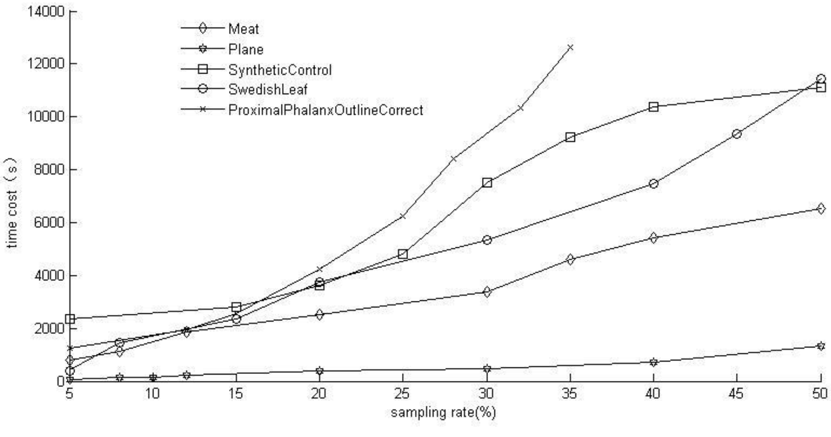

Figure 3.Influence of different data sets on classification time in different data sets

From figure 2, the sampling rate has great influence on the value of the final classification results, when the sampling rate value is too low, the sample data can’t restore the distribution structure of training data, When the sampling rate reaches the threshold, the classification result tends to be stable, a relatively stable amplitude fluctuation prove that the small sample has good representativeness.

As can be seen from the figure3, the sampling rate and time consumption are positively correlated, which indicates that when determining the sampling rate, in order to ensure the acquisition of small representative samples, the sampling rate can be appropriately increased to ensure that the sampling rate reaches an appropriate level. This upper limit is determined by the training data itself. Figure 2 also shows that the magnitude of the impact of different data sampling rate is different, such as data ProximalPhalanxOutlineCorrect refinanced at a low sampling rate can maintain the overall classification accuracy of a higher level, which can reflect a lot of similar sequences in this dataset in a certain extent, allowing for a low sampling rate to calculate shapelet, at the same time take the data sampling rate is too high will increase significantly classification under the condition of time, so the figure given in its highest sampling rate was 35% after many parameters testing, will set the sample rate as formula (6) of experiment in this paper

p=5/ The number of instances of the minority class (6)

For example, the minimum number of class instances in the Trace training data set is 21, and the sampling rate p value is 5/21=0.23. The purpose of this processing is to restore the inter-class and intra-class relations of the original training data in a small sample as far as possible.

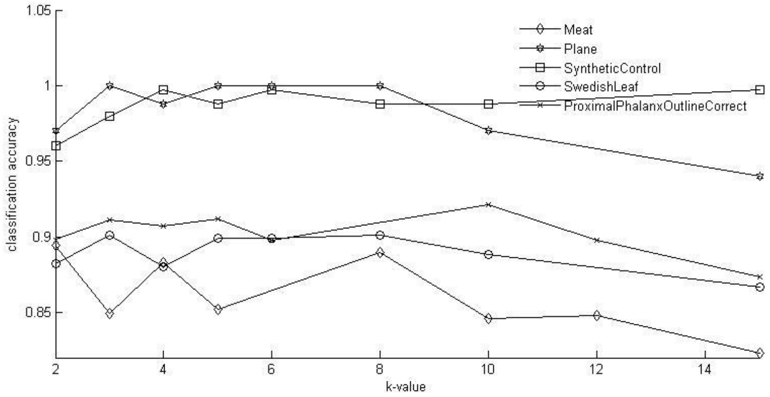

Figure 4.Influence of kvalue of k-means on classification accuracy in different data sets

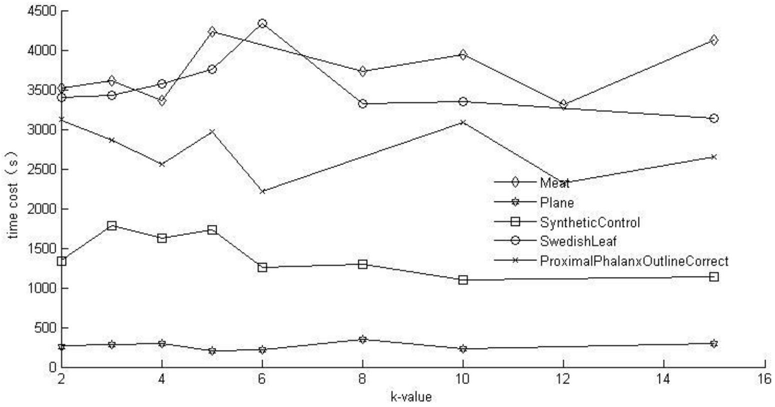

Figure 5.Influence of kvalue of k-means on classification time in different data sets

Figure 4 and figure 5 shows the five data of k value changes impact on the classification accuracy of time and curve, it can be seen from the diagram that the k value of the set of classification time had no obvious effect in terms of classification accuracy, different k value will also affect the classification accuracy on a certain extent, it is worth noting that when k is over 10,classification accuracy have a a small decline, k value can be too high to reach good clustering effect, result in the pretreatment process to completely random sampling, finally in this experiment, set

$$k = \left\lceil \sqrt{class\_number} \right\rceil$$

Finally, the number of classification n is set as 8, which is related to the number of nodes running in the distributed computing environment adopted in this experiment. It is a parameter that needs to be flexibly selected according to the distributed computing environment, and the selected parameter should be able to maximize the operation ability of the cluster.

5.3 experiments and results

Table 2.Comparison of experimental time

Data set name	ST-HESCA	Suggested method	Increase ratio
car	21515	13242	38.5%
ChlorineConcentration	17919	12558	29.9%
Computers	15357	14368	6.4%
DistalPhalanxOutlineCorrect	42437	3873	90.9%
ElectricDeviceOn	25874	1238	95.2%
Ham	31904	2794	91.2%
Herring	19766	1157	94.1%
Meat	8259	3386	59.0%
Plane	896	285	68.2%
ProximalPhalanxOutlineCorrect	30263	2564	91.5%
ProximalPhalanxOutlineAgeGroup	12190	232	98.1%
SLeaf5	48227	1151	97.6%
SwedishLeaf	72749	3757	94.8%
SyntheticControl	3415	172	95.0%
Trace	13054	1285	90.2%

Table 2 and table 3 shows the method respectively and ST - HESCA algorithm on the 15 sets of data classification accuracy and running time, two set of method of parameter is the default parameters of ST-HESCA algorithm. The increase ratio calculation formula is (*ST-HESCA takes – suggested algorithm takes*)/*ST-HESCA takes* in table 2, In order to visually display the improvement effect of the method in this paper, the data with higher accuracy in the two methods in table 3 are shown in bold. In this experiment, the time of ST-HESCA refers to the total time for training data shapelet to calculate shapelet transformation and for classification test data to obtain classification results. The time of suggested algorithm refers to the whole time of the training data preprocessing, parallel calculation of shapelet, and the use of the classification algorithm HESCA after the transformation to obtain the classification results. The main difference between the two methods lies in the different extraction strategies of shapelet.

By comparing the running time of all data, it can be seen that the method in this paper can steadily improve the running efficiency of shapelet transform classification algorithm. It is not difficult to find that there are significant individual differences among different data sets in the promotion effect of different data by observing the promotion amplitude and the corresponding data.

Table 3.Comparison of experimental accuracy

Data set name	ST- HESCA	Suggested method
car	0.900	**0.917**
ChlorineConcentration	**0.700**	0.695
Computers	0.604	**0.664**
DistalPhalanxOutlineCorrect	0.754	**0.779**
ElectricDeviceOn	0.596	**0.607**
Ham	0.676	**0.705**
Herring	**0.641**	0.625
Meat	0.850	**0.883**
Plane	1.000	1.000
ProximalPhalanxOutlineCorrect	0.890	**0.907**
ProximalPhalanxOutlineAgeGroup	0.829	**0.868**
SLeaf5	0.984	**0.989**
SwedishLeaf	**0.902**	0.899
SyntheticControl	0.993	0.993
Trace	**1.000**	0.990

In terms of accuracy, it can be seen that the classification accuracy of the method in this paper is basically the same as that of the original algorithm, which can reverse exceed the original algorithm on some data sets, and maintain an acceptable level when the accuracy is lower than that of the original algorithm. In order to explore whether the accuracy of the method in this paper and the ST-HESCA algorithm has been significantly improved, We then performed the Wilcoxon Signed-Rank Test, which is a non-parametric test of paired samples

5.4 Wilcoxon Signed-Rank Test

Results are shown in table 4, where sample 1 and sample 2 respectively represent the classification accuracy of HESCA algorithm and the method in this paper on each data set, and sample difference represents the difference between signed paired sample data.

Table 4.First step Calculation of sign rank test

sample1	sample 2	Difference value of samples
0.900	0.917	-0.017
0.700	0.695	0.005
0.604	0.664	-0.060
0.754	0.779	-0.025
0.596	0.607	-0.011
0.676	0.705	-0.029
0.641	0.625	0.016
0.850	0.883	-0.033
1.000	1.000	0.000
0.890	0.907	-0.017
0.829	0.868	-0.039
0.984	0.989	-0.005
0.902	0.899	0.003
0.993	0.993	0.000
1.000	0.990	0.010

Table 5.last step Calculation of sign rank test

Absolute Difference value	The rank of Difference	Rank with sign
0.017	7.5	-7.5
0.005	2.5	+2.5
0.060	13	-13
0.025	9	-9
0.011	5	-5
0.029	10	-10
0.016	6	+6
0.033	11	-11
0.017	7.5	-7.5
0.039	12	-12
0.005	2.5	-2.5
0.003	1	+1
0.010	4	+4

Table 4 count the difference of two kinds of matching data for operation,, then sort the difference, sorting will remove the value 0, and divide the same difference numerical rank value, the final sign rank test result can be seen in table 5.

rank total value is 13.5, the sum of obtained double-end confidence p-value is 0.02524, which means there are more than 95% probability to prove that suggested method on the 15 sets of data accuracy is significantly higher than ST-HESCA algorithm, it shows that the candidate set of shapelet extracted by this method can not only meet the requirement of improving the collection speed of shapelet, but also guarantee the quality of acquired shapelet.

VI. CONCLUSIONS

In this paper, aiming at the low efficiency of traditional shapelet calculation method in large-scale data set, a parallel shapelet classification algorithm is designed based on mapreduce computing architecture, which can calculate shapelet candidate sets on small sample sets on multiple clusters in parallel, and merge it into the optimal shapelets set for the whole set, for feature mapping and transformation of data sets in the experimental part, algorithm performance verification was carried out for 15 large-scale data sets of UCR. From the results, it can be seen that the method suggested in this paper can keep the high accuracy of shapelet classification algorithm on the basis of speeding up the operation efficiency, and even improve the accuracy on some data sets.

Acknowledgement This work was supported by National Natural Science Foundation of China (No.61876186), the Youth Science Foundation of China University of Mining and Technology(No. 2013QNB16)

Disclosure statement The authors declare that there is no conflict of interests regarding the publication of this manuscript

REFERENCES

[1]*Ye L, Keogh E. Time series shapelets: a new primitive for data mining[C]//Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009: 947-956*

[2]*CHEN Y K E, HU B, ET AL. The UCR Time Series Classification Archive [M].http://www.cs.ucr.edu/~eamonn/time_series_data/. 2015.*

[3]*Rakthanmanon, T., & Keogh, E. (2013, May). Fast shapelets: A scalable algorithm for discovering time series shapelets. In proceedings of the 2013 SIAM International Conference on Data Mining (pp. 668-676). Society for Industrial and Applied Mathematics.*

[4]*Zhang, Z., Zhang, H., Wen, Y., & Yuan, X. (2016, September). Accelerating time series shapelets discovery with key points. In Asia-Pacific Web Conference (pp. 330-342). Springer, Cham.*

[5]*Mueen, A., Keogh, E., & Young, N. (2011, August). Logical-shapelets: anexpressive primitive for time series classification. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1154-1162). ACM.*

[6]*Grabocka, J., Schilling, N., Wistuba, M., & Schmidt-Thieme, L. (2014, August). Learning time-series shapelets. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 392-401). ACM.*

[7] *J.A. Sáez, B. Krawczyk, M. Wozniak ´ , Analyzing the oversampling of different classes and types of examples in multi-class imbalanced datasets, Pattern Recognit. 57 (2016) 164–178.*

[8]*Li, G., Yan, W., & Wu, Z. (2019). Discovering shapelets with key points in time series classification. EXPERT SYSTEMS WITH APPLICATIONS, 132, 76-86.*

[9]*Xing, Z., Pei, J., Yu, P. S., & Wang, K. (2011, April). Extracting interpretable features for early classification on time series. In Proceedings of the 2011 SIAM International Conference on Data Mining (pp. 247-258). Society for Industrial and Applied Mathematics.*

[10]*Fang, Z., Wang, P., & Wang, W. (2018, April). Efficient Learning Interpretable Shapelets for Accurate Time Series Classification. In 2018 IEEE 34th International Conference on Data Engineering (ICDE) (pp. 497-508). IEEE.*

[11]*Schäfer, P., & Högqvist, M. (2012, March). SFA: a symbolic fourier approximation and index for similarity search in high dimensional datasets. In Proceedings of the 15th International Conference on Extending Database Technology (pp. 516-527). ACM.*

[12]*Ji, C., Zhao, C., Liu, S., Yang, C., Pan, L., Wu, L., & Meng, X. (2019). A fast shapelet selection algorithm for time series classification. Computer Networks, 148, 231-240.*

[13]*Gordon, D., Hendler, D., & Rokach, L. (2015). Fast and space-efficient shapelets-based time-series classification. Intelligent Data Analysis, 19(5), 953-981.*

[14]*Large J, Lines J, Bagnall A (2017) The Heterogeneous Ensembles of Standard Classification Algorithms (HESCA): the Whole is Greater than the Sumofits Parts (pp. 1-31), URL http://arxiv.org/abs/1710.09220, 1710.09220.*

[15]*Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 31(3), 606-660.*

[16]*Lines, J., Taylor, S., & Bagnall, A. (2016, December). HIVE-COTE: The hierarchical vote collective of transformation-based ensembles for time series classification. In Data Mining (ICDM), 2016 IEEE 16th International Conference on (pp. 1041-1046). IEEE*

[17]*Bagnall, A., Lines, J., Hills, J., & Bostrom, A. (2015). Time-series classification with COTE: the collective of transformation-based ensembles. IEEE Transactions on Knowledge and Data Engineering, 27(9), 2522-2535.*

[18]*Bostrom, A., & Bagnall, A. (2017). Binary shapelet transform for multiclass time series classification. In Transactions on Large-Scale Data-and Knowledge-Centered Systems XXXII (pp. 24-46). Springer, Berlin, Heidelberg.*

[19]*Anh, D. T., & Thanh, L. H. (2015). An efficient implementation of k-means clustering for time series data with DTW distance. International Journal of Business Intelligence and Data Mining, 10(3), 213-232..*