This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

A Big Data Clustering Algorithm for Mitigating the Risk of Customer Churn

Wenjie Bi, Meili Cai, Mengqi Liu and Guo Li

***Abstract*—As market competition intensifies, customer churn management is increasingly becoming an important means of competitive advantage for companies. However, when dealing with big data in the industry, existing churn prediction models cannot work very well. In addition, decision makers are always faced with imprecise operations management. In response to these difficulties, a new clustering algorithm called Semantic Driven Subtractive Clustering Method (SDSCM) is proposed. Experimental results indicate that SDSCM has stronger clustering semantic strength than Subtractive Clustering Method (SCM) and fuzzy c-means (FCM). Then a parallel SDSCM algorithm is implemented through a Hadoop MapReduce framework. In the case study, the proposed parallel SDSCM algorithm enjoys a fast running speed when compared with the other methods. Furthermore, We provide some marketing strategies in accordance with the clustering results, and a simplified marketing activity is simulated to ensure profit maximization.**

***Index Terms*—Axiomatic Fuzzy Sets, MapReduce, Semantic Driven Subtractive Clustering Method, Subtractive Clustering Method.**

I. Introduction

NOWADAYS, expanding market shares has become more and more tough for service industry, such as telecommunications industry, as the competition is fierce and market is increasingly saturated [1]. Thus, these companies pay more attention to the existing customers so as to avoid customer churn. Customer churn refers to the loss of customers who switch from one company to another competitor within a given period [2]. Industrial practice has shown that customer churn can lead to huge economic losses and even hurt the company's public image. Hence, customer churn management is extremely important especially for the service industry.

There exist many effective ways in the literature for handling customer churn management problem. Analytical methods mainly include statistical models, machine learning and dada mining [3]. Castro and Tsuzuki [4] propose a frequency analysis approach based on k-nearest neighbors machine learning algorithm for feature representation from login records for churn prediction modeling. Au et al. [5] propose a new data mining algorithm, called data mining by evolutionary learning (DMEL), to handle classification problems. Moreover, it is applied to predict churn under different churn rates with telecom subscriber data. Decision tree, neural network and K-means are selected by [6] as main techniques to build predictive models for telecom customer churn prediction. Their empirical evaluation indicates that data mining techniques can effectively assist telecom service providers to make more accurate customer churn prediction. Verbraken et al. [7] formalize a cost-benefit analysis framework and define a new expected maximum profit criterion. This general framework is then applied to the customer churn problem with its particular cost-benefit structure. Recently, based on a boosting algorithm, a robust churn prediction model has been successfully applied in churn prediction in the banking industry [2]. Although these methods can deal with customer churn problem efficiently, we should also notice that they are limited to process small structured data, such as account data, call details data, etc., all of which are less than ten thousand records. However, with the widespread adoption of smart phones and growth in mobile internet, companies today have accumulated unprecedented amounts of data sources. Take China Telecom as an example, as of July 2015, the system generates 10.5 trillion user-domain data records, and the corresponding data storage amount is hundreds of terabytes per day. The huge volume of data has the typical features of big data, and is hence referred to as "telco big data", which are the data to be analyzed in this paper and include call detailed records, Internet traffic logs, user profiles, location updates, social networking information so on and so forth. On account of big data have 55% possibility to bring the most value to operators in the area of customer retention, companies are eagerly seeking big data analytics solutions to solve customer churn problem for the sake of turning the data into valuable business insights.

In fact, the use of industrial big data for customer churn management has caught researchers' eyes because traditional methods are not engineered for the type of big, dynamic and unstructured data. For instance, Cloudera introduces some big

Manuscript received August 12, 2015. Accepted for publication February 25, 2016. This research was supported by National Natural Science Foundation of China (Grant nos. 71371191, 71210003, 71471057，71372019), PSFC NO.2015JJ2194 and Innovation Driven Planning of Central South University NO. 2015CX010. Paper no. TII-15-1272. *(Corresponding author: Guo Li.)*

W. J. Bi and M. L. Cai are with the School of Business, Central South University, Changsha, 410083 China. (e-mail: beenjoy@126.com; cai_meili@163.com).

M. Q. Liu is with the School of Business Administration, Hunan University, Changsha 410083, China (e-mail: liumengqi1976@163.com).

G. Li is with the School of Management and Economics, Beijing Institute of Technology, Beijing 100081, China (phone: +86-13811751697; fax: +86-10-68912483., e-mail: liguo@bit.edu.cn)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

data use cases for telcos including customer experience management, network optimization, operational analytics and data monetization. And Huang et al. [8] empirically demonstrate that telco big data make churn prediction much easier through 3V's perspectives. Actually, the difficulty for customer churn management lies in selecting analytical algorithms which allow them to cope with industrial big data [9]. However, there exists few literature studying big data clustering algorithms. Thus, with the purpose of developing the efficient algorithm in view of customer churn management, we first notice that there are two major challenges when utilizing telco big data:

1. It is difficult to give an explicit definition for the fuzzy concept – "customer churn". For example, how to define customers of telecom operators have lost? Is it elimination of telephone numbers? In arrears for more than three months (post-paid customers)? Or no calling within three months (pre-paid customers)? Moreover, with the telco big data, identifying the characteristics of churning customers is more challenging because a customer's decision of whether to churn is also affected by the social networking information, such as comments of friends, recommendations of a celebrity, etc. Therefore, how to efficiently mine those unstructured social networking data, and thus identify the churning customers? That is, how to define the "customer churn" on the basis of social networking information?

2. The existing analytical methods do not work very well in dealing with big data. Firstly, as mentioned earlier, in order to fully define "customer churn", we need to efficiently mine the unstructured social networking data. Because the relational database which stores traditional data effectively cannot process the unstructured data, so do the analytical methods. Therefore, we need to propose a new distributed computing method which is able to process the data stored in a specific infrastructure. Secondly, the volume of telco big data has reached a TB level. For example, in telco big data, business support systems (BSS) and operations support systems (OSS) sources are around 2.3TB new coming data per day, such as real time billing and call details information, unstructured information like textual complaints, mobile search queries and trajectories. Thus, indentifying the churning customers is extremely difficult. As the traditional analytical methods are more likely to encounter performance bottlenecks when conducting customer churn analysis, the new analytical method require to be as accurate as possible [10]. Therefore, to maximize the value of telco big data, proposing an efficient and appropriate parallel algorithm is a big challenge.

To solve the first problem, we introduce Axiomatic Fuzzy Sets (AFS) to generalize the definition of customer churn [11]. The key idea of AFS method is that several simple concepts or attributes can express many complex concepts by AFS algebra and AFS structure [12]. Moreover, it can deal with the preferences attributes, which reflects the preference extent of a customer influenced by social networking. For instance, let $X = \{x_1, x_2, x_3, x_4, x_5\}$ be a set of five customers. $M = \{\mathrm{m}_1, \mathrm{m}_2, \mathrm{m}_3, \mathrm{m}_4\}$ is a set of attributes, where m_1 = elimination of telephone numbers, m_2 = without phone calls in three months, m_3 = without data traffic in three months, m_4 = in arrears for more than three months. For a fuzzy concept $\eta = \mathrm{m}_1 + \{\mathrm{m}_2, \mathrm{m}_3\} + \mathrm{m}_4$, the semantic significance of η is: "persons who have eliminated telephone numbers" or "persons without phone calls and data traffic in three months" or "persons in arrears for more than three months", which represents "customers who are more prone to churn".

As for the second challenge, motivated by Subtractive Clustering Method (SCM) and AFS [11], [13], we propose a new big data clustering algorithm called Semantic Driven Subtractive Clustering Method (SDSCM). Although the distributed K-means algorithm is popular, the clustering results are likely to be imprecise if the initial parameters are valued improperly [14]. In comparison, SCM is usually used to generate more precise input parameters for K-means based on raw data, including cluster centroids and clustering number [13]. Nevertheless, there exist many uncertain parameters in SCM, which leads to the clustering inaccuracy. Hence, to improve its accuracy, the parameters of SCM are determined automatically in SDSCM. Furthermore, a parallel SDSCM algorithm is implemented through a Hadoop MapReduce framework for real-time and efficient data analysis.

The three main contributions of this paper are as follows. Firstly, we propose a new algorithm called SDSCM, which improves clustering accuracy of SCM and K-means. Moreover, this algorithm decreases the risk of imprecise operations management by using AFS. Secondly, to deal with industrial big data, we propose a parallel SDSCM algorithm through a Hadoop MapReduce framework. Thirdly, in the case study of China Telecom, the results show that the parallel SDSCM and parallel K-means have high performance, when compared with traditional methods.

The rest of this paper is organized as follows. In section II, urgent problems faced by companies are described and some basic theories are introduced. In section III, a new clustering algorithm called SDSCM is proposed. In section IV, some evaluation indexes are introduced and experiments are conducted on standard data sets to compare the performance of SDSCM, SCM and Fuzzy C-means (FCM). In section V, modifications of SDSCM and K-means with MapReduce are presented. In section VI, the distributed clustering methods are implemented to address the problem of customer churn faced by China Telecom. The last section is for conclusion and future research.

II. BACKGROUND

A. Problem Description

This paper provides new methods to help the company better mitigate the risk of customer churn, and hence to gain higher profits. It mainly studies customer churn problem in service industry under the big data environment. Specific problems include:

1. How to define the fuzzy concept "customer churn"

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

comprehensively?

2. Since many parameters affect the accuracy of SCM, how to obtain their values properly based on raw data?

3. How to evaluate the semantic strength of algorithms? And how to design a new clustering algorithm by integrating semantic fuzzy concept with SCM, which performs better than FCM and SCM?

4. To deal with big data sets effectively, how can we modify the new algorithm be to a parallel one?

5. How to use the clustering results to guide the customer churn management of the company? And which customers' cluster is the marketing target?

To tackle these problems, we first need some basic theories.

B. Axiomatic Fuzzy Set (AFS)

AFS is an effective way to describe the fuzzy concept. The membership functions and their logic operations are determined by original data and facts instead of intuition [15]. Moreover, several simple concepts or attributes can express many complex concepts by using AFS algebra and AFS structure [12]. The AFS structure and the membership function are defined as follows.

Let (M,τ,X) be an AFS structure, where X is the universe of discourse, and M is a set of attributes or concepts on X. $\tau(x,y)=\{m|m\in M,(x,y)\in R_m\}\in 2^M$, where R_m is a preference relation. For $A\subseteq M$ and $x,y\subset X$, the following symbol is defined [11]:

$$A^{\tau}(x)=\{y|y\in X,\tau(x,y)\supseteq A\}. \quad (1)$$

Let X and M be sets, (M,τ,X) be an AFS structure and (M,σ,m) be a measure space, where m is a finite and positive measure with $m(X)\neq 0$. Then, $(M,\tau,\mathrm{X},\sigma,\mathrm{m})$ is called as a sim-cognitive field. If $\underline{A_k}(\{x\})\in\sigma$, $\forall k=1,2,...,n,x\in X$, $\sum_{k=1}^{n}A_k$ is a measurable concept of (M,τ,X) under σ whose membership function is defined as follows [11]:

$$\mu_{\sum_{k=1}^{n}A_k}(x)=SUP_{1\le i\le n}(m(\underline{A_i}(\{x\})/m(X)),\forall x\in X. \quad (2)$$

Membership μ represents the degree that x belongs to the concept, which means the higher μ is, the closer x is to the concept.

Assume $m(\mathrm{A})=|\mathrm{A}|$ ($|\mathrm{A}|$ is the number of elements in set A). The fuzzy set membership function of equation (2) is completely determined by sub-preference relations of the simple concepts in M.

For example, let $X=\{x_1,x_2,x_3,x_4,x_5\}$ be a set of five customers. $M=\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3,\mathrm{m}_4,\mathrm{m}_5,\mathrm{m}_6\}$ is a set of attributes, where m_1 = Peak Calls, m_2 = Off-peak Calls, m_3 = Weekend Calls, m_4 = National Calls, m_5 = International Calls, m_6 = loyalty. The sub-preference relation of m_6 is expressed as $x_1>x_2=x_5>x_4>x_3$, where $x_1>x_2$ means x_1 is more loyal than x_2.

TABLE I
DESCRIPTIONS OF ATTRIBUTES

	Peak Calls	Off-peak Calls	Weekend Calls	National Calls	International calls
x_1	73	31	1	105	0
x_2	54	9	14	77	2
x_3	57	32	6	95	1
x_4	25	21	1	47	9
x_5	57	6	20	83	0

For a fuzzy concept $\eta=\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3\}+\{\mathrm{m}_4,\mathrm{m}_5\}+\mathrm{m}_6$, the semantic significance of η is "persons with many calls during peak time and off-peak time and weekend" or "persons with many national and international calls" or "high loyal persons", which represents "customers who are less prone to churn".

According to equation (2), $\eta(\mathrm{x}_i)$ and the membership $\mu_\eta(x_i)$ of each customer can be calculated as follows.

$$\eta(\mathrm{x}_1)=\{\mathrm{x}_4\}\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3\}+\{\mathrm{x}_5\}\{\mathrm{m}_4,\mathrm{m}_5\}+\{\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5\}\{\mathrm{m}_6\}$$
$$\eta(\mathrm{x}_2)=\varnothing\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3\}+\varnothing\{\mathrm{m}_4,\mathrm{m}_5\}+\{\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5\}\{\mathrm{m}_6\}$$
$$\eta(\mathrm{x}_3)=\{\mathrm{x}_4\}\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3\}+\{\mathrm{x}_5\}\{\mathrm{m}_4,\mathrm{m}_5\}+\varnothing\{\mathrm{m}_6\}$$
$$\eta(\mathrm{x}_4)=\varnothing\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3\}+\varnothing\{\mathrm{m}_4,\mathrm{m}_5\}+\{\mathrm{x}_3\}\{\mathrm{m}_6\}$$
$$\eta(\mathrm{x}_5)=\varnothing\{\mathrm{m}_1,\mathrm{m}_2,\mathrm{m}_3\}+\varnothing\{\mathrm{m}_4,\mathrm{m}_5\}+\{\mathrm{x}_3,\mathrm{x}_4\}\{\mathrm{m}_6\}$$
$$\mu_\eta(x_1)=m(\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)/m(\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)=0.8$$
$$\mu_\eta(x_2)=m(\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)/m(\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)=0.6$$
$$\mu_\eta(x_3)=m(\mathrm{x}_4,\mathrm{x}_5)/m(\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)=0.4$$
$$\mu_\eta(x_4)=m(\mathrm{x}_3)/m(\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)=0.2$$
$$\mu_\eta(x_5)=m(\mathrm{x}_3,\mathrm{x}_4)/m(\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4,\mathrm{x}_5)=0.4$$

It is obvious that x_1 has the largest membership value, which means x_1 is closest to the semantic concept. That is, x_1 is loyal and less prone to churn. On the contrary, x_4 has the smallest membership value, which means x_4 is prone to change brands and has a high probability to churn.

C. Subtractive Clustering Method (SCM)

In this subsection, we introduce SCM for computing the cluster centroids, which belongs to unsupervised learning and can quickly determine the number of clusters and cluster centroids based on the raw data [16].

In SCM, each data point is considered as a potential cluster centroid and the potential is computed as

$$M_1(x_i)=\sum_{j=1}^{n}\exp\left(-\frac{\|x_i-x_j\|^2}{(\tau_1/2)^2}\right), \quad (3)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

where τ_1 is the neighbor radius which influences the scope of a cluster centroid. The larger τ_1 is, the greater its impact will be. Thus, the data point with maximum mountain function is the first centroid. Then update the mountain function of each data according to equation (4).

$$M_l(x_i) = M_l(x_i) - M_{l-1}^{*} \exp\left(-\frac{\| x_i - x_l^{*} \|^2}{(\tau_2/2)^2} \right) , \quad (4)$$

where τ_2 is the influencing weight of the last cluster centroid. Data points near the first cluster centroid will have greatly reduced potential, and thus unlikely to be the next cluster centroid. To avoid getting close cluster centroids, according to [13], in general, $\tau_2 = 1.5\tau_1$.

D. K-means

K-means is the simplest one among all these clustering methods [17], [18]. Hence, we introduce it here for computing the clusters. Note that the clustering results are likely to be imprecise if having improperly valued initial parameters in K-means. On the contrary, SCM can generate more precise input parameters based on raw data, including cluster centroids and clustering number [13], [19]. Consequently, in this paper, the parameters generated by SCM pass to K-means so as to improve the accuracy of K-means.

The algorithm of K-means starts with initialized k cluster centroids. Then, data are iteratively assigned to the nearest cluster and the new centroids of k clusters are re-calculated until the termination conditions are reached.

III. Semantic Driven Subtractive Clustering Method (SDSCM)

For solving the first two problems in section II, we integrate AFS and SCM, which forming the new algorithm, SDSCM. The innovation points of SDSCM are:

1. It can fully express semantic signification of fuzzy concept.
2. It can automatically determine the neighbor radius and the weight coefficient of SCM.
3. It sets a termination condition on the basis of an earlier study reasonably.

The procedure of SDSCM is shown in the Fig.1.

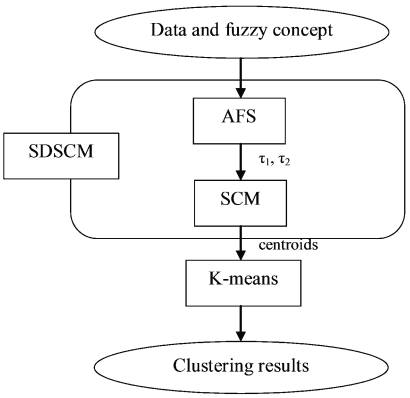

Fig. 1. Flow chart of clustering methods

Firstly, we use AFS to select related attributes for expressing the fuzzy concept by its membership function and logic operations. Then, according to the calculated membership, we determine the neighbor radius and the weight coefficient of SCM automatically. Thirdly, we use SCM to compute the cluster number and centroids by selecting and updating mountain functions. In this paper, we integrate SCM and AFS as SDSCM. Finally, we use K-means to calculate the clusters with the cluster centroids obtained by SDSCM. The details of SDSCM algorithm are shown as follows.

A. The Algorithm of SDSCM

TABLE II
NOTATION USED BY SDSCM

Symbol	Description
η	The fuzzy concept.
x_i	$x_i \in X$ and x_1^F is the first centroid in parameter determination. x_1^* is the first centroid in SCM.
$\mu_\eta(x_i)$	The membership of x_i
p_i	The sum of absolute differences of $\mu_\eta(x_i)$, and p_1^F is the minimum.
d_i	The Euclidean distance between the first cluster centroid and the other data points. $\bar{d}$ is the mean.
τ_1	The neighbor radius.
τ_2	The weight coefficient.
l	The cycle index.
M_i	The mountain function and M_l^* is the maximum in the l cycle.
ε	The termination condition.

Algorithm 1: the algorithm of SDSCM

Step 1: According to the fuzzy concept η given by the user, use equation (5) to compute the membership of x_i.

$$\mu_\eta(x) = SUP(m(\eta)/m(X)), \forall x \in X . \quad (5)$$

Step 2: Compute the sum of absolute differences of $\mu_\eta(x_i)$ as

$$p_i = \sum_k^n \left| \mu_\eta(x_i) - \mu_\eta(x_k) \right| . \quad (6)$$

Step 3: Select the minimum value of p_i as the first cluster centroid.

$$p_1^F = \min\{p_i\} , \quad (7)$$

$$x_1^F = x_i . \quad (8)$$

Step 4: Compute the Euclidean distance between the first cluster centroid and other data points. The neighbor radius τ_1 is the variance of these distances which influences the scope of a cluster centroid.

$$d_i = | x_i - x_1^F |^2 , \quad (9)$$

$$\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i , \quad (10)$$

$$\tau_1 = \frac{1}{n}\sum_{i=1}^{n} (d_i - \bar{d}) . \quad (11)$$

Step 5: To avoid getting close cluster centroids, set the weight coefficient $\tau_2 = 1.5\tau_1$ [13].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

After determining the parameters automatically, we use the algorithm of SCM to compute the cluster centroids [13].

Step 6: Let $l=1$ and compute the mountain function of x_i.

$$M_l(x_i)=\sum_{j=1}^{n}\exp\left(-\frac{\| x_i-x_j \|^2}{(\tau_1/2)^2}\right). \quad (12)$$

Step 7: Select the maximum mountain function.

$$M_l^*=Max_i[M_l(x_i)]. \quad (13)$$

Meanwhile, let x_i be the first centroid x_1^*.

$$x_1^*=x_i. \quad (14)$$

Step 8: Let $l=l+1$, and update the mountain function of each data vector according to

$$M_l(x_i)=M_l(x_i)-M_{l-1}^*\exp\left(-\frac{\| x_i-x_l^* \|^2}{(\tau_2/2)^2}\right). \quad (15)$$

Step 9: Select the data associated with larger $M_l(x_i)$ to be the second centroid, and execute ***step 6*** repeatedly until

$$M_l^*<\varepsilon M_1^* \quad (16)$$

is satisfied, where ε is a positive constant less than 1. When the ratio is smaller than ε, the iteration stops [16].

Step 10: Finally, output the cluster centroids.

In equation (6), Small p_i means the membership of x_k and x_i are almost same for the fuzzy concept η, that is, x_k and x_i belong to the same cluster with high probability. Therefore, the data point with minimum p_i is chosen as the first cluster centroid. This approach is similar to SCM where cluster centroids are selected using maximum mountain functions. Outliers may occur given the instability of raw data. Therefore, we calculate τ_1 by equations (9), (10) and (11) because variance reflects the dispersion degree of data. Moreover, variance is more stable than Range. The smaller the variance is, the more the points near x_i, and the smaller the τ_1 is. According to [13], $\tau_2=1.5\tau_1$. Hence, we determine τ_1 and τ_2 completely by the raw data and semantic concept without artificial intervention.

B. Time Complexity Analysis

Assume that the data set has n data points. Each data point has m attributes. In addition, the data set is divided into l clusters. To calculate the time complexity of the SDSCM, we should consider four main steps, those are, step 2, 6, 8 and 9. The time complexities of step 2 and 6 are same and equal to $O(mn^2)$. The time complexity of step 8 is $O((l-1)mn)$ and that of the step 9 is $O(lmn)$. Based on these results, the time complexity of SDSCM is $O(mn^2)$.

IV. EVALUATION INDEXES AND EXPERIMENTAL RESULTS

This section evaluates the performance of SDSCM. Firstly, we introduce some classical evaluation indexes. Then, we propose new indexes for evaluating clustering strength of algorithms. Thirdly, we conduct two experiments on standard data sets to make a comparative analysis among SDSCM, SCM and FCM.

A. Evaluation Indexes

Generally, classical evaluation indexes for analyzing and evaluating clustering results include Cluster Separation (SPT), Cluster Compactness (CMP) and Evaluation Validity (EVA) [20]. Furthermore, to evaluate the semantic strength of algorithms, we propose new evaluation indexes called Semantic strength (SS) and Semantic strength expectation (SSE). The notations related to these indexes are shown in Table III.

TABLE III
NOTATION USED BY THE EVALUATION INDEXES

Symbol	Description
v_i	The centroid of i^{th} cluster and $\|\|v_i\|\|$ is the number of data points belonging to the cluster with centroid v_i
v_j	The centroid of j^{th} cluster.
x_j	The data point.
N	The total number of the data points.
μ_{ij}^m	The membership of the j^{th} datum, and generally $m=2$
C	The number of clusters.
η	The fuzzy concept.
$\mu_\eta(x_j)$	The membership of x_j based on η.

$$SPT=\min\left\|v_i-v_j\right\|^2. \quad (17)$$

SPT reflects the degree of separation among clusters. The larger the SPT is, the higher the clustering quality might be.

$$CMP=\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{C}\mu_{ij}^m\left(x_j-v_i\right)^2. \quad (18)$$

CMP is used to measure the tightness of each cluster. The smaller the CMP is, the better the cohesion of the clusters might be.

CMP and SPT evaluate a cluster's compactness and its difference with other clusters, respectively. A good clustering result should have a smaller CMP and a larger SPT. However, they cannot fully reflect the overall clustering quality. Thus, we introduce EVA further.

$$EVA=\frac{CMP}{SPT}. \quad (19)$$

Obviously, the smaller the EVA is, the higher the overall clustering quality might be.

Note that when the three indexes are not consistent, EVA is chosen as the major evaluation index.

We define semantic strength (SS) as the average membership of each cluster based on a given fuzzy concept:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

$$SS(\mathrm{v}_i) = \frac{\sum_{x_j \in v_i} \mu_\eta(x_j)}{\| \mathrm{v}_i \|}. \quad (20)$$

Thus, the higher the SS is, the higher the membership of cluster might be, and the closer the cluster and fuzzy semantic concept meets, which indicates a stronger semantic strength. Conversely, the smaller the SS is, the weaker the semantic strength might be.

Moreover, we define semantic strength expectation (SSE) as the maximum product of clustering accuracy and semantic strength:

$$SSE = \max_{j \in I}(E \times SS(C_j)) \quad , \quad (21)$$

where E refers to the accuracy of clustering. And E is calculated by comparing the experimental results of the algorithm with the ground truth in the test set. In this paper, the data set is divided into 60% training data and 40% checking data. The higher the SSE is, the closer the cluster results to the fuzzy semantic concept are, and the stronger the effect of the semantic strength might be. Conversely, the smaller the SSE is, the weaker the effect of semantics is.

B. *Experiment results of Iris data set.*

Iris data set is well known since Fisher used it in several statistical experiments [21]. It contains three clusters, each of which represents a type of iris and has 50 elements. The samples are distributed in three clusters. One cluster is linearly separable from the other two, and the other two have some overlap.

In view of the Iris data set, the parameters of SDSCM are input as follows:

$M = \{\mathrm{m}_1, \mathrm{m}_2, \mathrm{m}_3, \mathrm{m}_4\}$. m_1 is sepal length. m_2 is sepal width. m_3 is petal length. m_4 is petal width. $\eta = A_1 = \mathrm{m}_1 + \mathrm{m}_2 + \mathrm{m}_3 + \mathrm{m}_4$.

The parameters of SCM are input as follows:

1) The neighbor radius $\tau_1 = 0.87$.

2) The weight coefficient $\tau_2 = 1.305$.

3) The terminal condition $\varepsilon = 0.00001$.

The parameters of FCM are input as follows:

1) Fuzzy factor $m = 2$.

2) The number of clusters $C = 3$.

3) The terminal condition $\varepsilon = 0.00001$.

Specific procedures for the three algorithms are verified with MATLAB 7.11.0. Simulation results are shown in Table IV.

TABLE IV
EVALUATION INDEXES FOR THE IRIS DATA SET

Algorithm	E	CMP	SPT	EVA	SSE
SDSCM	96%	0.0382	0.6441	0.0593	0.4034
SCM	89%	0.0365	0.6038	0.0605	0.3891
FCM	100%	0.0419	0.6403	0.0655	0.3484

It can be seen from Table IV that under the fuzzy semantic concept, CMP of SDSCM is smaller than that of FCM, but larger than that of SCM, which indicates that clustering coherence and effect of SDSCM are better. Furthermore, SSE of SDSCM is larger than that of SCM and FCM. That means the results produced by SDSCM are closer to the fuzzy semantic concept and the semantic effect of clustering is better. Nevertheless, FCM has the highest accuracy.

C. *Experiment results of Wine data set.*

Wine data set is the chemical results of three different wine made in the same place in Italy. It has a good clustering structure, containing 178 samples and 13 numeric attributes. It is divided into three clusters with different sample sizes.

The three algorithms used for the Wine data set are the same as those used for the Iris data set. Experiment results are shown in Table V.

TABLE V
EVALUATION INDEXES FOR THE WINE DATA SET

Algorithm	E	CMP	SPT	EVA	SSE
SDSCM	95.16%	0.1246	0.5796	0.2150	0.1568
SCM	93.40%	0.1781	0.6984	0.2550	0.0972
FCM	94.94%	0.1613	0.6308	0.2557	0.0791

As shown in Table V, with the fuzzy semantic concept, CMP and EVA of SDSCM are smaller than those of SCM and FCM. The results are similar to those of the Iris data set which indicate that clusters clustered by SDSCM have better internal compactness but weaker distinction from each other. Meanwhile, the semantic strength of SDSCM is strongest.

D. *Semantic analysis of SDSCM.*

It can be seen from the above experiments that SDSCM has larger semantic strength expectation than the other two algorithms. That means clustering results of SDSCM are closest to the desired results. Fig. 2 and Fig 3 show the obtained semantic strength.

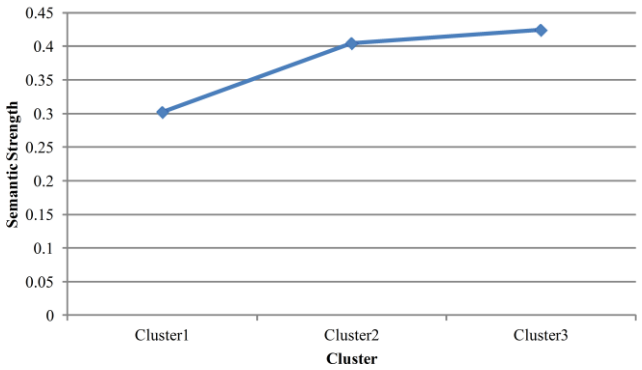

Fig. 2. Semantic strength for Iris data set

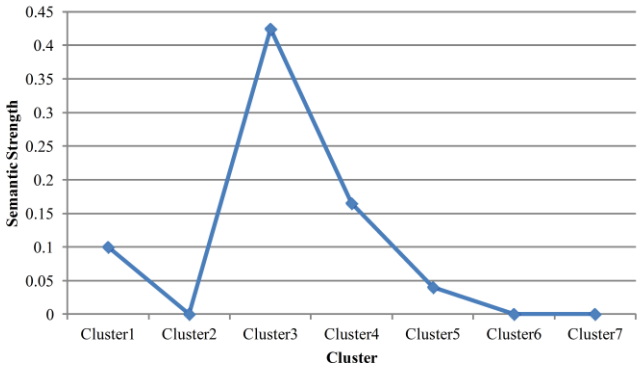

Fig. 3. Semantic strength for Wine data set

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

Fig. 2 and Fig 3 illustrate that the third cluster of Iris data set and the fourth cluster of Wine data set have the largest SS, separately. Hence, users can select these two clusters as prior targets of practical applications.

V. IMPLEMENTATION OF SDSCM AND K-MEANS WITH MAPREDUCE

In section IV, the experiment results show that SDSCM has the best clustering quality and strongest semantic strength. Therefore, SDSCM is a good choice to process industry big data. In order to solve the forth problem in section II, in this section, we design a parallel SDSCM algorithm which is implemented with the MapReduce programming model. Also, we introduce the parallel K-means [18].

The procedure of implementing the clustering methods in big data analytics is shown in Fig.4.

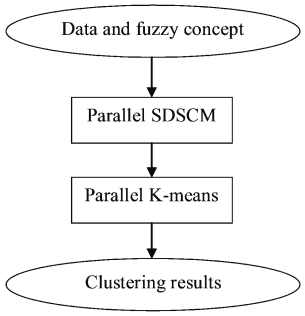

Fig. 4. Flow chart of clustering methods

A. Deploying SDSCM with MapReduce

In this subsection, we design a parallel SDSCM algorithm for the Hadoop MapReduce framework. The MapReduce is a programming model [22]. A map function is executed for each key/value pair of the input, and generates a set of intermediate key/value pairs. Then the reduce function merges all intermediate values associated with the same intermediate key.

In parallel SDSCM, we denote the distances between data points as a matrix. The row of the matrix represents a data point and the column represents the attribute of a data point. Suppose that m data points with n attributes are needed to be processed, so two matrices A and B with m rows and n columns can be obtained. For the convenience of calculating the distance between data points in MapReduce, A is set to be same as B. Then the distances between data points are the difference of data in the same rows of A and B. The distance matrix is denoted as D. Provided that A and B are large, matrices can be partitioned for parallel computing and distributed systems. Therefore, the parallel SDSCM can be realized with a MapReduce framework. The algorithm contains five steps.

Algorithm 2: the algorithm of parallel SDSCM

Step 1: compute distance matrix.

Input: two $m \times n$ matrices A and B

Output: the distance matrix D

The map function:

Read each A_{ij} in A and B_{ij} in B where $i = 1, 2, ...m, \quad j = 1, 2, ...n$. The map function emits $< (i,k), (A, j, A_{ij}) >, < (k,i), (B, j, B_{ij}) >$ pairs respectively and $k = 1, 2, ...m$.

The combine function:

The combine function collects all the outputs of the map function. Merge $< (i,k), (A, j, A_{ij}) >$ and $< (k,i), (B, j, B_{ij}) >$ associated with the same key as new $< key, \{data\} >$ pairs. Then transfer the new key/value pairs to the reduce function.

The reduce function:

First, sort $< key, \{data\} >$ pairs in descending order by j and save them in two different lists A' and B'. Second, compute the square difference of A_{ij} in A' and B_{ij} in B' associated with the same i and j. Third, sum the results d. Finally, output the $< (i,k), d >$ pairs where d is the data of the distance matrix D.

Step 2: parameter determination

Input: the fuzzy concept η, the distance matrix D

Output: the neighbor radius τ_1

The map function:

First, read A_{ij} of A, B_{ij} of B and compute membership function μ_i^A and μ_i^B according to equation (1) separately. Second, save them in two different lists A'' and B''. Third, emit $< i, \mu_i^A, \mu_i^B >$ pairs.

The reduce function:

First, compute p_i according to equation (2) and generate $< i, p_i >$ pairs. Second, by bubble sorting, minimum p_i denoted as p_i^F is obtained. Then the corresponding data d_i^F in D are obtained. Third, Compute τ_1 according to equations (5), (6) and (7). Finally, emit $< i, \tau_1 >$ pairs.

Step 3: initialize mountain function.

Input: the distance matrix D and neighbor radius τ_1

Output: the mountain function of each data point M_i

The map function:

Read the data of D and each datum stores as a $< (i,k), d >$ pair. Split the pairs and form new $< i, d >$ pairs.

The combine function:

The combine function collects all the outputs of the map function. Merge d of $< i, d >$ associated with the same i as new $< key, \{data\} >$ pairs. Then transfer the new key/value pairs to the reduce function.

The reduce function:

Sum the data associated with the same key according to equation (8). Then the mountain function of each data point M_i is obtained. By bubble sorting, the maximum mountain

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

function M_1^* and its corresponding data x_1^* are obtained. Finally, the reduce function emits $< i, M_i >$ pairs.

Step 4: update mountain function

Input: the last mountain function of each datum, the row of the last cluster centroid and the neighbor radius τ_1

Output: the updated mountain functions M_i

The map function:

Read the corresponding i and d of the last cluster centroid x_{l-1}^* and store as the $< (l-1, i), (d, M_i) >$ pairs. Split the pairs and compute the new updated mountain functions according to equation (11). Emit the new $< i, M_i >$ pairs.

The reduce function:

Write the updated $< i, M_i >$ pairs into the files.

Step 5: get cluster centroids

By bubble sorting, the maximum mountain function M_l^* is obtained. If equation (12) is satisfied, a set of cluster centroids $\{x^*\}$ is obtained. Otherwise, the MapReduce programming model is reconfigured and new cluster centroids is obtained.

Eventually, we get the cluster centroids using big data SDSCM with MapReduce, which is the input of big data K-means.

B. Time Complexity Analysis

Provided that there are Q nodes participating in computing in Hadoop system and each node can complete w Map tasks. The time complexity of the first step is $O(mn^2 / Qw)$ and that of the second step is $O(n^2 / Qw)$. The time complexity of the third step is $O(n / Qw)$ and that of the fourth step is $O((l-1)n / Qw)$. The time complexity of the fifth step is $O(ln / Qw)$. Therefore, the time complexity of parallel SDSCM is $O(mn^2 / Qw)$.

In conclusion, compared with SDSCM, the modified big data SDSCM improves time efficiency in theory.

C. Deploying K-means with MapReduce

In this subsection, we introduce a parallel K-means algorithm for the Hadoop MapReduce framework [23]. When processing large scale data sets with K-means algorithm, the first step is to partition data points to Q subgroups with equal size. The K-Means algorithm based on MapReduce is as follows.

Algorithm 3: the algorithm of parallel K-means

Input: the data points $\{x_i\}$ and l cluster centroids $\{x^*\}$

Output: members of each cluster

The map function:

First, calculate the distances between $\{x_i\}$ and $\{x^*\}$. Second, find the nearest centroids with each data points. Finally, emit $< x^*, x_i >$ pairs.

The reduce function:

Update $\{x^*\}$ by a mean calculation and obtain the new cluster centroids $\{x^{*\prime}\}$. If $\{x^*\}$ equals to $\{x^{*\prime}\}$, then the clusters are obtained. Otherwise, emit $< x^{*\prime}, x_i >$ pairs as the inputs of the next MapReduce job in accordance with the next iteration.

VI. CASE STUDY OF CHINA TELECOM

In order to solve the fifth problem in section II, in this section, we implement parallel SDSCM and parallel K-means with MapReduce in the case of China Telecom for dealing with customer churn.

A model for developing a customer churn management framework is established as shown in Fig.5 [1].

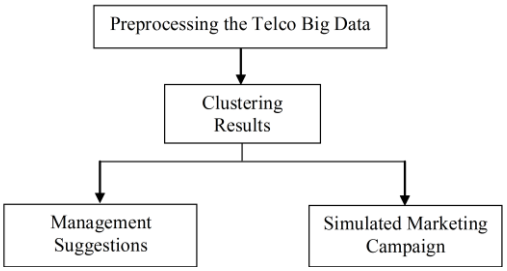

Fig. 5. The churn management framework

A. Preprocessing the Telco Big Data

China Telecom has been gradually transformed into data-oriented company and has accumulated a huge volume of valuable data on subscriber behaviors, service usage, and network operations. Thus, these accumulated data are identified as "telco big data" in this paper. As preprocessing of large input data sets has been confirmed to increase the plausibility and accuracy of the churn forecasts, we should first define the fuzzy concept and preprocess the telco big data [24].

In this case, "customer churn" is the semantic fuzzy concept needed to be defined clearly. After analyzing the business activities, this concept is defined in the following three ways: elimination of telephone numbers, in arrears for more than three months (post-paid customers) or no calling within three months (pre-paid customers).

Next, we preprocess the telco big data. It is worth noting that we design the big data clustering algorithm from the perspective of telco big data, therefore, it is applicable for all such data. And in this section, for verifying the effectiveness of the algorithm, we specific analyze the data from business support systems (BSS) and operations support systems (OSS), which constitute almost 97% of telco big data. Generally, most customer behavior features are extracted from BSS, including call detailed records (minutes of international calls, counts of national calls, etc.), billing records (cost of total calls, account balance, etc.), demographic information (name, age, address, etc.), package/handset (Phone charge plan, recharge value), purchase history, compliant records and customer levels (VIP or non-VIP) [8]. And the data volume in BSS is around 24GB per day. While OSS is a computer system which supports management functions such as network configuration, fault management, etc. The data from OSS mainly include circuit

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

switch (CS, evaluating the call connection quality), packet switch (PS, describing users' mobile web behaviors), and measurement report (MR, estimating users' approximate trajectories). And the data volume in OSS is around 2.2TB per day. Also, we use web crawler to obtain some social networks data.

We use the multi-vendor data adaption module to change tables to the standard format. Then, we import them to big data platform by using Extract-Transform-Load (ETL). Thirdly, we store these raw tables in Hadoop distributed file systems (HDFS), where most data are stored in regular tables or sparse matrix. And finally, we choose the features to be analyzed based on Hive/Spark SQL and some learning algorithms, such as PageRank, label propagation, etc. Furthermore, we have shown some basic features in Table VI. Note that the "Loyalty" variable is a semantic feature, which describes the loyal degree of a customer, that is, the preference of a customer influenced by social networking. Mathematically, "Loyalty" is a sub-preference relation among customers, similar to the "Loyalty" concept in section II.

TABLE VI
VARIABLES

Features	Description	Features	Description
Age	Age of user	Average_Cost_Min	Cost of per minute
Loyalty	Duration in net	Weekend_Calls	Count of calls during weekend
Tariff	Phone charge plan	Weekend_Mins	Minutes of calls during weekend
Level	Customer levels (VIP or non-VIP)	MO_SMS	count of MO SMS
Peak_Calls	Count of calls during peak time	MO_MMS	count of MO MMS
Peak_Mins	Minutes of calls during peak time	MT_SMS	count of MT SMS
OffPeak_Calls	Count of calls during off-peak time	MT_MMS	count of MT MMS
OffPeak_Mins	Minutes of calls during off- peak time	Recharge_Balance	recharge over account balance
International_Mins	Minutes of international calls	Credit_Value	user credit value
National_Calls	Count of national calls	Voice_Calls	count of voice call
National _Mins	Minutes of national calls	Voice _Mins	duration of voice call
All_Calls_Mins	Minutes of all calls	Balance	account balance
Nat_Call _Cost	Costs of national calls	Recharge	recharge value
Avepeak	Average minutes of calls during peak time	Up_Speed	data upload speed
Aveoffpeak	Average minutes of calls during off-peak time	Down_Speed	data download speed
Aveweekend	Average minutes of calls during weekend	TCP_Status	TCP connection status
Avenational	Average minutes of nation calls	TCP_Mins	TCP return time
Actual Call Cost	Cost of actual calls	Stream_Size	streaming file size
Total Call Cost	Cost of total calls	Stream_Packets	streaming download packets
Total Cost	Cost	Alert_Time	alert time
Call_Cost_Per_Min	Call cost of per minute	Local_Code	local area code

B. Clustering Results

Based on the above preparations, we implement the parallel SDSCM and parallel K-means with a historical dataset which contains the information of over 1.8 million subscribers. The Apache Hadoop version is 1.0. And the platform has 16 nodes of the cluster, each of which is Intel (R) Core (TM) 2 Duo CPU E7400 @ 2.8 GHz, 2 GB. We connect the nodes via local area network and the operating system on each node is ubuntu 12.04.

The semantic concept is described as "high churn rate". According to step 2 in Algorithm 2, we implement AFS for determining parameters of SCM automatically. The results are that τ_1 equals to 0.523 and τ_2 0.7845.

The comparative results of the serial SDSCM and parallel SDSCM are shown in Fig.6 and Fig.7.

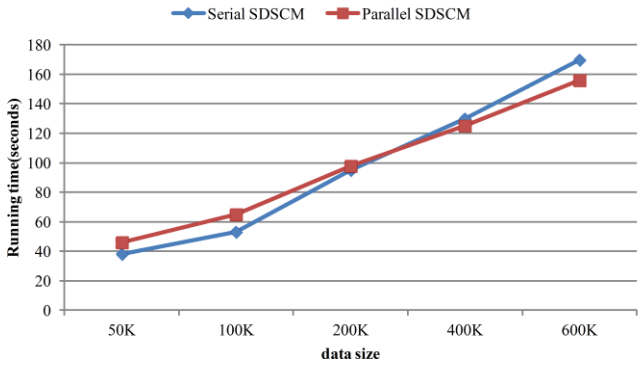

Fig.6. Running time of the serial SDSCM and parallel SDSCM deployed on an increasing scale of data set

It is noticeable from Fig. 6 that running time of the parallel SDSCM is longer than the serial SDSCM when processing small data sets, which is because the concurrent processes of parallel SDSCM spawn. These two algorithms have the same running time when the size of the data set is 250K. When the size of the data set exceeds that threshold, the parallel SDSCM gradually becomes more efficient. However, once the size of the data set reaches 600K, the running time of parallel SCM is unobservable because the machine is out of memory.

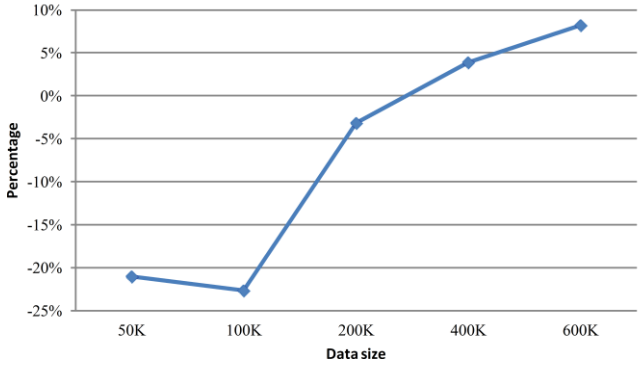

Fig.7. Percentage of running time speedup at different data sizes

It can be seen from Fig. 7 that the advantage of parallel SDSCM in computation speed is obvious when the size of the dataset is between 100K and 200K. Nevertheless, it increases slowly when the size of the data set is larger than 400K.

In summary, compared with serial SDSCM, parallel SDSCM is much more efficient to deal with large data set.

The accelerated ratio testing is also conducted. The results

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

are shown in Fig.8.

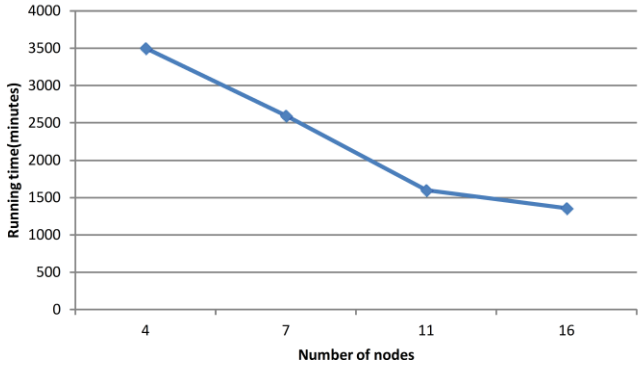

Fig.8. Accelerated ratio test of the proposed algorithm deployed on an increasing number of nodes

In stand-alone mode, the machine is not capable of dealing with data whose size is larger than 600K. Hence, to process big data set successfully, using more nodes is necessary. In the experiments, 4 nodes, 7 nodes, 11 nodes and 16 nodes are used to do the accelerated ratio test respectively. It is apparent from Fig. 8 that the running time of each task decreases with the increase of nodes. Therefore, increasing the number of nodes significantly improves the performance of handling the same data set. That indicates that MapReudce programming model is conducive to implement parallel SDSCM and K-means. But when the processing nodes are more than 11, the running time of the algorithm is not dramatically reduced after adding more nodes. The reason is that communication cost increases rapidly with the growing number of nodes. Thus, for a given data size, it is not true that the more processing nodes is, the faster the running time is. To avoid wasting valuable distributed resources, measurement among the data size, the number of processing nodes and the communication cost is essential.

C. Management suggestion

In the subsection C and D, we use a 2GB data set to do the analysis. Clustering results of subsection C is shown in this part. Note that the data set has marked the status of customers. The number 1 and 0 mean churned customers and stable customers, separately. After clustering, churn rate for every cluster is obtained. The objective is to prevent customer churn in the future. The analysis of the churn rate and characteristics of each cluster can help business people develop precise marketing strategies.

Churn rates of clusters are shown in Table VII. In Table VII, x_i is the amount of customers in each cluster and e is the churn rate. According to Table VI, the attribute $Total\_Cost$ represents a customer's total cost within 6 months, namely the value of a customer to the company. Therefore, the total value of each cluster is $V_i = \sum_1^{x_i} Total\_Cost$.

As shown in Table VII, cluster-6 and cluster-8 have relatively high churn rates, which indicates that customers in these two clusters are more prone to churn. Therefore, marketing activities should care more about these customers. After analyzing their characteristics of cluster-6 and cluster-8, we provide some useful management suggestions in Table VIII.

TABLE VII
CHURN RATES OF CLUSTERS

Number of Cluster	x_i	e	V_i
Cluster-6	24,149	0.093	2,149,261
Cluster-8	21,938	0.086	2,171,862
Cluster-1	16,425	0.082	1,182,600
Cluster-5	14,180	0.071	907,520
Cluster-2	9,321	0.065	1,006,668
Cluster-7	7,196	0.059	813,148
Cluster-3	3,940	0.058	480,680
Cluster-4	2,851	0.045	373,481

TABLE VIII
CHARACTERS OF CUSTOMER CHURN

Number of cluster	Ratio	Key characteristics	Character generalization	Management suggestions
Cluster-6	13.6%	Total cost is low; Average cost per minute is high, and charged duration is less in addition to free charge; Call minutes are less during peak and off-peak time and more on weekend.	Low-value customers; Calls are mostly on weekend; Tariff plans need to be optimized.	Provide preferential policies in peak time and weekend; Optimize tariff plans by providing more free call minutes or cashback activity.
Cluster-8	11.4%	Call less but long average minutes on weekend; Rate of call minutes on weekend is low; Minutes are more at peak time and short at off-peak time; Total cost is high.	High-value customers; Work phone type; Family phone over the weekend.	Recommend new popular business for entertaining, such as mobile phone ringtone, newsletter, etc; Provide preferential policies on workdays.

D. Marketing simulation

To deal with the fifth problem in section II, we hold a simulated marketing activity.

Section D has obtained the clustering results of SDSCM. Now the most difficult problem of the company is to select the worthy customers that need to be retained. In order to maximize profit, the company should keep customers who may create the maximum value in minimum cost. If the decision is wrong, carrying out the real market activities, e.g., such as phone bill discount, not only waste resources, but also hardly meet the best marketing effect. Thus, it is necessary to simulate a marketing strategy before implementing the real market promotion activities. In the simulation, we calculate the expected net income of the above eight clusters, and then we choose the target cluster which has the largest expected net income. Along with specific retention measures, we guarantee the company to achieve maximum profits.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2547584, IEEE Transactions on Industrial Informatics

There are three parameters in the simulation.

1. The response rate (r) stands for the proportion of the customers who attend the promotion activity organized by the company, namely phone bill discount, and continue to use its phone services in the long run. This parameter is determined by historical data of marketing activities.

2. The retaining cost (c) represents the average cost of passing the marketing information through the telephone or SMS to a potential customer that needs to be retained.

3. The discount rate (d).

There are four steps in the simulation.

1. According to Table VII, SDSCM algorithm divides the customers into eight clusters, and there are x_i customers in each cluster with e churn rate. Provided that the response rate is r and the retaining cost is c. Then, for each cluster, the responsive customers are $x_i \times e \times r$ and the total cost for each cluster is $C_i = c \times x_i \times e \times r$.

2. According to Table VII, the total value of each cluster (V_i) is obtained. Provided that the discount rate is d, then the total income of marketing activity against i cluster is $P_i = V_i \times (1-d)$.

3. The expected net income of each cluster is $N_i = P_i - C_i$.

4. Choose the corresponding cluster with maximum N_i as a prior marketing target.

To increase the accuracy of the simulation, by considering the parameter value of previous marketing activities and the business experience of the manager of Marketing Department, we assume all these three random variables to be uniformly distributed: $r \sim U(0,1)$, $c \sim U(0,1)$, $d \sim U(0,0.6)$.

According to the above simulation process, we carry out 500 random experiments. Moreover, we calculate the expected net income of each cluster and sort them in a descending order. It turns out that for 500 experiments, the descending orders are same. The same order is cluster-8, cluster-6, cluster-1, cluster-2, cluster-5, cluster-7, cluster-3, and cluster-4.

When $r = 80\%$, $c =$ ¥10 and $d = 5\%$, the experimental results are shown in Table IX.

TABLE IX
RESULTS OF MARKETING SIMULATION

Number of Cluster	x_i	e	V_i	C_i	P_i	N_i
Cluster-6	24,149	0.093	2,149,261	17,967	2,041,798	2,023,831
Cluster-8	21,938	0.086	2,171,862*	15,093	2,063,269	2,048,176*
Cluster-1	16,425	0.082	1,182,600	10,775	1,123,470	1,107,045
Cluster-5	14,180	0.071	907,520	8,054	862,144	854,090
Cluster-2	9,321	0.065	1,006,668	4,847	956,335	951,488
Cluster-7	7,196	0.059	813,148	3,397	772,491	769,094
Cluster-3	3,940	0.058	480,680	1,828	456,646	454,818
Cluster-4	2,851	0.045	373,481	1,026	354,807	353,781

The maximum N_i is ¥2,048,176 and the corresponding cluster is cluster-8. According to Table IV, cluster-6 has the highest churn rate, which means retaining the customers in this cluster can ensure companies have more customers, but the marketing activity for cluster-6 does not maximize the profits. Thus, the company should focus its attention on the cluster-8 and design specific marketing strategies for maximizing its profits.

Through making a deep analysis of the experimental results, we find that the order of the expected net income (N_i) is same as the order of the total value of each cluster (V_i). Furthermore, the above conclusion is not affected by the value of the random choose the cluster with highest value as the target marketing objects. Consequently, the company can design specific marketing strategies for obtaining higher profit.

VII. CONCLUSION

In this paper, aiming to provide companies with effective methods to prevent churning customers in big data era, we first propose a new clustering method called SDSCM based on SCM and AFS. SDSCM improves clustering accuracy of SCM and K-means. Moreover, it decreases the risk of imprecise operations management by using AFS. Experiment results indicate that SDSCM has stronger clustering semantic strength than SCM and FCM. The second contribution is that we modify the earlier serial SDSCM to big data SDSCM, and implement it with a MapReduce framework. Thirdly, we solve the customer churn problem in China telecom with big data SDSCM and big data K-means algorithms. In this case, we use the BSS and OSS data to verify the good performance of the proposed big data SDSCM. Moreover, we hold a simulated marketing campaign to find the potential customers who will be retained with lowest cost. Results show that the marketing simulation is essential to gain maximizing profits for enterprises and the enterprises should pay more attention to the valuable clusters (customers). In conclusion, the process of solving customer churn problem in China Telecom has offered novel insights for managers to raise the level of customer churn management in the big data context.

In the future, we will improve the algorithm to validate the effectiveness in terms of other risk analysis and using more abundant implementation platform.

REFERENCES

[1] J. Hadden, A. Tiwari, R. Roy, D. Ruta, "Computer assisted customer churn management: State-of-the-art and future trends," *Comput. Oper. Res.*, vol. 34, no. 10, pp. 2902-2917, Oct. 2007.

[2] N. Lu, H. Lin, J. Lu, G. Zhang, "A customer churn prediction model in telecom industry using boosting," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1659-1665, May 2014.

[3] B. Q. Huang, T. K. Mohand, B. Brian, "Customer churn prediction in telecommunications," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 1414-1425, Jan. 2012.

[4] E. G. Castro, M.S.G. Tsuzuki, , "Churn Prediction in Online Games Using Players' Login Records: A Frequency Analysis Approach," *IEEE Trans. Comput. Intell. AI Games*, vol. 7, no. 3, pp. 255-265, Sep 2015.

[5] W. H. Au, K. C. C. Chan, Y. Xin, "A novel evolutionary data mining algorithm with applications to churn prediction," *IEEE Trans. Evol. Comput.*, vol. 7, no. 6, pp. 532-545, Dec. 2003.

[6] S. Y. Hung, D. C. Yen, H. Y. Wang, "Applying data mining to telecom churn management," *Expert Sys. Appl.*, vol. 31, no. 3, pp. 515-524, Oct. 2006.

[7] T. Verbraken, V. Wouter, B. Bart, "A novel profit maximizing metric for measuring classification performance of customer churn prediction models," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 5, pp. 961-973, May 2013.

[8] Y. Huang, F. Zhu, M. Yuan, K. Deng, B. "Telco Churn Prediction with Big Data", i*n Proc. the 2015 ACM SIGMOD Int. Conf. Manage. Data*, San Francisco, 2015, pp, 607-618.

[9] CL. Chen, CY. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data", *Inform. Sci.*, vol. 275, pp. 314-347, Aug. 2014.

[10] H. Li, D. Wu, GX Li, "Enhancing Telco Service Quality with Big Data Enabled Churn Analysis: Infrastructure, Model, and Deployment", *J. Comput. Sci. Technology*, vol. 30, no. 6, pp. 1201-1214, Nov. 2015.

[11] X. D. Liu, "A new mathematical axiomatic system of fuzzy sets and systems," *Int. J. Fuzzy Math.*, vol. 3, pp. 559-560, 1995.

[12] X. D. Liu, "The fuzzy sets and systems based on AFS structure, EI algebra and EII algebra," *Fuzzy Sets and Syst.*, vol. 95, no. 2, pp. 179-188, Apr. 1998.

[13] S. L. Chiu, "Fuzzy model identification based on cluster estimation," *J. Intell. and Fuzzy Syst.*, vol. 2, no. 3, pp. 267-278, 1994.

[14] X. Wu, X. Zhu, G. Q. Wu, W. Ding, "Data mining with big data," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 197-107, Jan. 2014.

[15] X. D. Liu, W. Wang, T. Chai, "The fuzzy clustering analysis based on AFS theory," *IEEE Trans. Syst., Man, Cybern. B, Cybern*, vol. 35, no. 5, pp. 1013–1027, Oct. 2005.

[16] G. Bilgin, E. Sarp, Y. Tülay, "Segmentation of hyperspectral images via subtractive clustering and cluster validation using one-class support vector machines," *IEEE Trans. Geoscience and Remote Sensing*, vol. 49, no. 8, pp. 2936-2944, Spet. 2011.

[17] K. J. Kohlhoff, S. P. Vijay, B. A. Russ, "K-means for parallel architectures using all-prefix-sum sorting and updating steps," *IEEE Trans. Parallel Distrib. Syst*, vol. 24, no. 8, pp. 1602-1612, Aug. 2013.

[18] C. Boutsidis, M. I. Malik, "Deterministic feature selection for k-means clustering," *IEEE Trans. Inf. Theory*, vol. 59, no. 9, pp. 6099-6110, Sept. 2013.

[19] F. Afsari, M. Eftekhari, E. Eslami, P. Y. Woo, "Interpretability-based fuzzy decision tree classifier a hybrid of the subtractive clustering and the multi-objective evolutionary algorithm," Soft Computing, vol. 17, no. 9, pp. 1673-1686, Jan. 2013.

[20] A. Garcia-Piquer, A. Fornells, J. Bavardit, A. Orriols-Puig, "Large-scale experimental evaluation of cluster representations for multi-objective evolutionary clustering," *IEEE Trans. Evol. Comput*, vol. 18, no. 1, pp. 36-53, Feb. 2014.

[21] A. Soualhi, C. Guy, R. Hubert, "Detection and diagnosis of faults in induction motor using an improved artificial ant clustering technique," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 4053-4062, Sept. 2013.

[22] J. Dean, S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Commun. the ACM*, vol. 51, no. 1, pp. 107-113, 2008.

[23] Y. J. Xu, W. Y. Qu, Z. Y. Li, G. Y. Min, "Efficient-means++ Approximation with MapReduce." *IEEE Trans. Parallel and Distrib. Syst.*, vol. 25, no. 12, pp. 3135-3144, Dec. 2014.

[24] I. Klevecka, J. Lelis, "Pre-processing of input data of neural networks: the case of forecasting telecommunication network traffic." *elektronikk: Telecommunications Forecasting (Special issue in co-operation with International Institute of Forecasters)*, vol. 104, no. 3/4 , pp. 168-178, 2008.

Wenjie Bi received the Ph.D. degree from Central South University, Changsha, China, in 2008.

He is a Professor of the School of Business, Central South University, Changsha, China. His research interests lie in the area of behavioral decision theory, supply chain risk management and dynamic pricing. He has published two research books and 11 refereed journal articles in *Knowledge-based Systems, Information Economics and Policy, Journal of Industrial and Management Optimization*, etc., and has hosted two National Nature Science Foundation of China.

Meili Cai received the B. Admin. degree in information management and information system from Ludong University, Yantai, China, in 2013. She is currently working toward the Master's degree from the Central South University in Management Science and Engineering, Changsha, China.

Her research interests include big data analytics, supply chain risk management.

Mengqi Liu received the Ph.D. degree from Hunan University, Changsha, China, in 2013.

He is an assistant Professor of School of Business Administration, Hunan University, Changsha, China. And he is now working in the area of capital operation and supply chain risk management. He has published 10 articles in *European Journal of Operational Research, Transportation Research Part E: Logistics and Transportation, and International Journal of Production Research*, etc., and is the director of one National Nature Science Foundation of China and one National development and Reform Commission Project.

Guo Li is an associate Professor of School of Management and Economics at Beijing Institute of Technology, China. He received his Ph.D. in Management Science and Engineering from Huazhong University of Science and Technology. He has published papers in *Annals of Operations Research, Journal of the Operational Research Society, Transportation Research Part E* and *International Journal of Production Research*, etc.

His research Interests include supply chain risk management, assembly system and interface between operation, marketing and finance.