Semi–supervised K-Means Clustering by Optimizing Initial Cluster Centers

Xin Wang[1], Chaofei Wang[2], and Junyi Shen[1]

[1] Department of Electronic and Information Engineering
Xi'an Jiaotong University, Xi'an 710049, China
wangx@cetin.net.cn
[2] China Defense Science and Technology Information Center
Beijing 100142, China
wcf-119@163.com

Abstract. Semi-supervised clustering uses a small amount of labeled data to aid and bias the clustering of unlabeled data. This paper explores the usage of labeled data to generate and optimize initial cluster centers for k-means algorithm. It proposes a max-distance search approach in order to find some optimal initial cluster centers from unlabeled data, especially when labeled data can't provide enough initial cluster centers. Experimental results demonstrate the advantages of this method over standard random selection and partial random selection, in which some initial cluster centers come from labeled data while the other come from unlabeled data by random selection.

Keywords: semi-supervised clustering, k-means, initial cluster centers, max-distance search.

1 Introduction

In pattern recognition, machine learning and relative fields, supervised learning is a method based on a great amount of labeled data to supply training sets. However, there are a mass of unlabeled data compared with limited labeled ones in many practical problems, such as Web Page Classification, Gene Analysis, Voice Recognition and so on. It is a truth that limited training data can not provide sufficient distribution information of dataset, which leads to unsatisfied result in practical applications. Meanwhile, significant number of unlabeled data is useless in supervised learning. Unsupervised learning method tries to build classifier by means of detecting hidden structures in unlabeled data. But it is difficult to make sure the accuracy on dealing with mass data. Therefore, it begins to raise more concern and become a new hot research issue that semi-supervised learning method utilizes comprehensively a bit labeled data and massive unlabeled ones.

For different learning tasks, semi- supervised learning can be divided into semi-supervised classification and semi-supervised clustering [1]. Semi-supervised classification [2, 3] makes use of a large amount of unlabeled data to enlarge the training set, which can compensate the disadvantage due to the inadequate labeled

Z. Gong et al. (Eds.): WISM 2011, Part II, LNCS 6988, pp. 178–187, 2011.

data. Semi-supervised clustering [4-7] utilizes some labeled data to obtain a better clustering. This paper explores the use of labeled data to generate initial cluster centers for k-means algorithm, which biases clustering towards a good direction. Semi-supervised classification algorithms cannot instead of semi- supervised clustering algorithms to complete some learning tasks, in which a small amount of labeled data can not reflect the structure of dataset [8].

This paper introduces a semi-supervised k-means clustering algorithm, called Seeded K-means (Basu etc., 2002), which uses labeled data to generates initial cluster centers. But this algorithm is proceeded in the assumption that the labeled dataset covers all categories (in other words, each category has one labeled data at least [9]). It is obvious that this hypothesis is very limited. A more common problem is that we only have some labeled data which come from a part of categories, and we have not any one labeled data which come from the other categories. Based on this situation, this paper presents a max-distance search approach to find some optimal initial cluster centers from unlabeled data. We present results of experiments which demonstrate the advantages of our method over standard random selection and partial random selection, in which some initial cluster centers come from labeled data while the other come from unlabeled data by random selection.

2 Problem Description

In this section, we definite some concepts used in this paper, and then introduce the background knowledge, finally describe the problem we are facing.

2.1 Seed Set

Given a dataset X as previously mentioned, k-means clustering of the dataset generates a k-partitioning $\{X_l\}_{l=1}^{k}$ of X. Let $S \subseteq X$, called the Seed Set, be a subset consisting of labeled data (called the Seeds) which are provided as follows: for each $x_i \in S$, the user provides the cluster X_l of the partition to which it belongs. The Seed Set is used to generate initial cluster centers for k-means algorithm.

2.2 Complete Seed Set and Incomplete Seed Set

We assume that corresponding to each partition X_l of X, there is typically at least one seed $x_i \in S$. Note that we get a Seed Set with a k-partitioning $\{S_l\}_{l=1}^{k}$, called the Complete Seed Set.

However, in most cases we can't get the Seed Set with a k-partitioning $\{S_l\}_{l=1}^{k}$ but a j-partitioning $\{S_l\}_{l=1}^{j}, (j < k)$, called the Incomplete Seed Set, which means that corresponding to some partitions of X (actually j of k), there is typically at least one seed $x_i \in S$, but corresponding to the other partitions (actually $k - j$ of k), there is not one seed $x_i \in S$.

An example of Complete Seed Set and Incomplete Seed Set is presented in Fig.1.

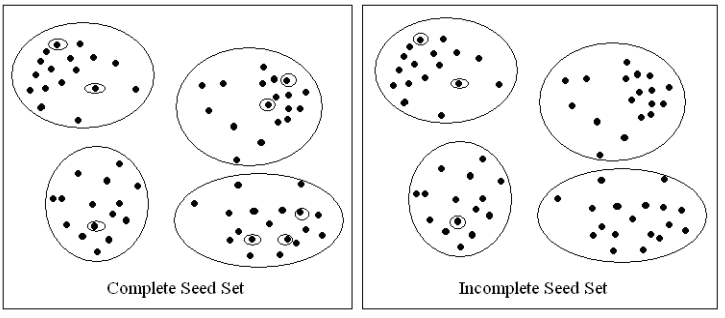

Fig. 1. Complete Seed Set and Incomplete Seed Set

2.3 Semi–supervised K-Means Clustering

K-means clustering (MacQueen, 1967) is a method commonly used to automatically partition a dataset into k clusters. The algorithm is presented in detail in Fig.2 [9, 10].

Algorithm: K-means
Input: a dataset $X = \{x_1, x_2, ..., x_N\}$, number of clusters k
Output: k-partitioning $\{X_i\}_{i=1}^{k}$ of X
Method:
1. Select k data points as the initial cluster centers $\{\mu_1, \mu_2, ..., \mu_k\}$.
2. Each data point x_i is assigned to its closest cluster center.
3. Each cluster center μ_i is updated to be the mean of its constituent data points.
4. Repeat 2 and 3 until K-means objective function is optimized.

Fig. 2. K-means Algorithm

It is well known that the most challenge of k-means algorithm is selection of the initial cluster centers. The traditional k-means algorithm randomly selects k data points as initial cluster centers from unlabeled dataset, which leads to the chances of it getting stuck in poor local optima. In the research of semi-supervised k-means algorithm, the outbreak is taking advantage of labeled data to obtain initial cluster centers.

So the problem is how can we utilize the limited labeled data to obtain good initial cluster centers?

3 Algorithms

In this section, we explain how semi-supervision can be incorporated into the k-means algorithm by optimizing initial cluster centers. According to the two kinds of Seed Set mentioned in chapter 2.2, we can divide semi-supervised k-means into the semi-supervised k-means based on Complete Seed Set, called “CSK-means” for short and

the semi-supervised k-means based on Incomplete Seed Set, called "ISK-means" for short, and then we give the mathematical motivation behind the two proposed algorithms.

3.2 CSK-Means

In CSK-means, a complete seed set is used to initialize the k-means algorithm. Thus, rather than initializing k-means from k random data points, the initial center of the l th cluster is initialized with the mean of the l th partition S_l of the seed set. The calculation of the initial cluster centers $\{\mu_l\}_{l=1,2,...,k}$ can be written as:

$$\mu_l = \frac{1}{|S_l|}\sum_{x \in S_l} x, l = 1,2,...,k \tag{1}$$

The algorithm is presented in detail in Fig. 3.

Algorithm: CSK-means
Input: a dataset $X = \{x_1, x_2, ..., x_N\}$, number of clusters k, set $S = \bigcup_{l=1}^{k} S_l$ is a complete seed set
Output: k-partitioning $\{X_l\}_{l=1}^{k}$ of X
Method:

1. initialize: $\mu_l = \frac{1}{|S_l|}\sum_{x \in S_l} x$, $l = 1,2,...,k$
2. Each data point $x_i \in S_l$ is assigned to the cluster center μ_l
3. Each data point $x_i \in X - S$ is assigned to its closest cluster center.
4. Each cluster center μ_l is updated to be the mean of its constituent data points.
5. Repeat 3 and 4 until K-means objective function is optimized.

Fig. 3. CSK-means clustering algorithm

3.3 ISK-Means

In ISK-means, an incomplete seed set is used to initialize the k-means algorithm. Thus, rather than CSK-means, the key point of ISK-means is how to calculate the $k - j$ initial cluster centers when j of k initial cluster centers can be initialized with the means of the j partitions $\{S_l\}_{l=1}^{j}, (j < k)$.

Partial random selection. A simple method to initialize the $k - j$ initial cluster centers is selecting them randomly from the unlabeled dataset, which is called partial random selection. We can call it ISK-means ^{P}, short for the ISK-means algorithm with the method of partial random selection. The algorithm is presented in detail in Fig. 4.

Easy to analysis, this algorithm has a better performance than unsupervised k-means, but the method of partial random selection still can make it get stuck in poor local optima.

Algorithm: ISK-means ^{p}
Input: a dataset $X = \{x_1, x_2, ..., x_N\}$, number of clusters k, set $S = \bigcup_{l=1}^{j} S_l, j < k$ is an incomplete seed set
Output: k-partitioning $\{X_l\}_{l=1}^{k}$ of X
Method:
1. Initialize:
1a. Caculate j of k initial cluster centers: $\mu_l = \frac{1}{|S_l|}\sum_{x \in S_l} x$, $l = 1,2,...,j$
1b. Select $k-j$ initial seeds from set $X - S$ randomly
2. Each data point $x_i \in S_l$ is assigned to the cluster center μ_l
3. Each data point $x_i \in X - S$ is assigned to its closest cluster center.
4. Each cluster center μ_l is updated to be the mean of its constituent data points.
5. Repeat 3 and 4 until K-means objective function is optimized.

Fig. 4. ISK-means ^{p} clustering algorithm

Max-distance searching. In fact, the Seed Set covering j categories has provided a priori knowledge for the rest of the $k-j$ categories. These $k-j$ initial centers should be far away from the j initial centers calculated by equation (2), because the distance between different categories must be as far as possible in a clustering problem. So it is a better idea to choose the farthest data point away from known j initial centers as the $(j+1)$ th initial center than random selection. This calculation method of initial centers is described as below.

- For the j categories covered in $S = \bigcup_{l=1}^{j} S_l, j < k$

$$\mu_l = \frac{1}{|S_l|}\sum_{x \in S_l} x \quad , \quad l = 1,2,..., j \tag{2}$$

- Choose the farthest data point away from known j initial centers as the $(j+1)$ th initial center:

$$\mu_{j+1} = x_f : when, \max \sum_{l=1}^{j} \left\| x_f - \mu_l \right\| \tag{3}$$

- Repeat step b2 until that all of k initial centers have been obtained.

An example of max-distance searching method is presented in detail in Fig. 5, and we can get a useful conclusion from it. These initial centers generated by max-distance searching have very good dispersion, which is in accord with the most important feature of clustering: distances between different clusters need to be as far as possible.

Furthermore, these initial centers can improve the performance of clustering algorithm. We can call it ISK-means ^{m}, short for the ISK-means algorithm with method of max-distance searching. The algorithm is presented in detail in Fig. 6.

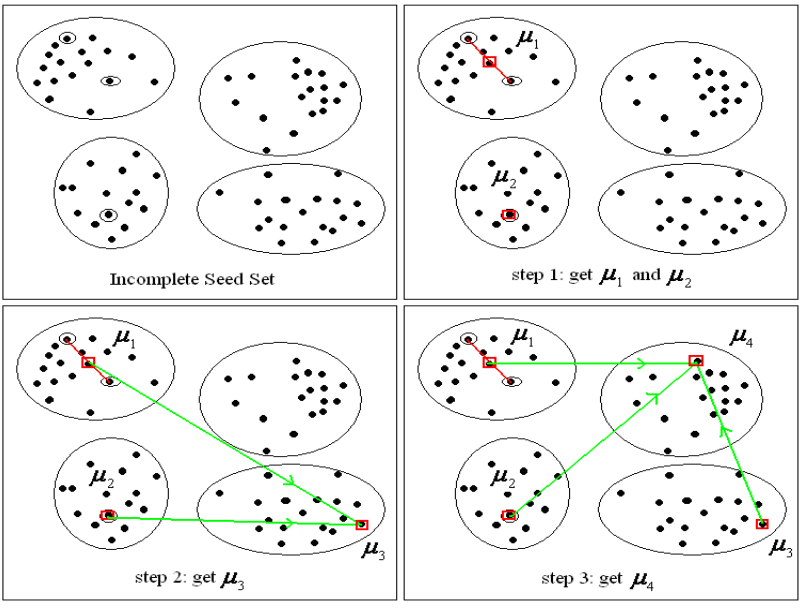

Fig. 5. An Example of Max-distance Searching Method

Algorithm: ISK-means ^{m}

Input: a dataset $X = \{x_1, x_2, ..., x_N\}$, number of clusters k, set $S = \bigcup_{i=1}^{j} S_i, j < k$ is an incomplete seed set

Output: k-partitioning $\{X_i\}_{i=1}^{k}$ of X

Method:

1. Initialize:

1a. Calculate j of k initial cluster centers: $\mu_l = \frac{1}{|S_l|}\sum_{x \in S_l} x$, $l = 1,2,...,j$

1b. Search $k-j$ initial cluster centers from set $X-S$ using max-distance searching

2. Each data point $x_i \in S_l$ is assigned to the cluster center μ_l
3. Each data point $x_i \in X-S$ is assigned to its closest cluster center.
4. Each cluster center μ_l is updated to be the mean of its constituent data points.
5. Repeat 3 and 4 until K-means objective function is optimized.

Fig. 6. ISK-means ^{m} clustering algorithm

4 Experiments

In our experiments, we made use of five UCI datasets [11], i.e. wine, iris, balance, sponge and spectf heart, and one text dataset of practical documents which had been downloaded from the sohu.com website. Its distribution has been shown as table 1. For each dataset, we ran four algorithms: Unsupervised K-means, CSK-means, ISK-means ^{p}, ISK-means ^{m}, and obtained a complete set (for CSK-means) and an incomplete set (for ISK- means ^{p} and ISK-means ^{m}) from the dataset by randomly selection. In all cases, results are averaged over 100 runs, and the accuracy was used to evaluate the performance of these algorithms.

Table 1. Distribution of text dataset

ID	Category Name	Total
1	Finance and Economics	60
2	Sports	80
3	Cars	84
4	Game	86
5	Tourism	76
6	Military affairs	65
7	House Property	45
8	Education	50
9	Healthy diet	66
10	Constellation	38

4.1 Results on UCI Data Sets

Table 2 presents the experimental results on five UCI datasets.

Table 2. Results on UCI datasets

Set	C	I	D	K-means	CSK-means	ISK-Means ^{p}	ISK-means ^{m}
wine	3	178	13	0.59	0.65	0.62	0.63
iris	3	150	4	0.88	0.89	0.88	0.88
balance	3	625	4	0.38	0.45	0.39	0.44
sponge	12	76	45	0.35	0.59	0.38	0.51
spectf	73	267	45	0.25	0.51	0.38	0.46

Note: C-the number of categories, I-the number of instances, D-the number of dimensions.

We can draw the flowing conclusions via analyzing these results:

Conclusion 1: There are few differences among the performances of these four algorithms if the number of clusters is small such as wine, iris and balance datasets. The main reason is that usually unsupervised k-means can get the global optima through numbers of experiments (over 100 times) when the dataset is a convex

dataset with small number of clusters. It is difficult to learn more knowledge from a seed set unless the situation that dataset itself has some noise points which are ticklish in unsupervised k-means but noted clearly in the seed set.

Conclusion 2: If the number of clusters is bigger, such as sponge and spectf heart dataset, the performance of CSK-means algorithm exceeds others. It illustrates that given more supervision can promote better performances of algorithms [12-13].

Conclusion 3: Especially when the seed set is incomplete, the performance of ISK-means ^{m} is quite similar with CSK-means whereas higher than ISK-means ^{p}. It shows that the initial cluster centers derived from max-distance search method are good discrete, meanwhile they are nearly as good as initial cluster centers obtained by complete seed set.

4.2 Results on Text Dataset

For the documents dataset, a vocabulary of 7,162 words except the stop words was generated. Each document is represented as a vector in a 7,162 dimensional space, with TFIDF weighting [14].

Table 3 shows the experimental results on the text datasets. We can conclude that CSK-means algorithm can get the global optima in the case of complete seed set, and ISK-means ^{m} algorithm can obtain better results than ISK-means ^{p} in the case of incomplete seed set. However, it is difficult to get a complete seed set actually. In contrast, it is easier to get incomplete one. Therefore, in practical applications it is more valuable to study the algorithms on incomplete seed set basis.

Table 3. Results on text dataset

Set	C	I	D	K-means	CSK-means	ISK-Means ^{p}	ISK-Means ^{m}
text	10	600	7162	0.52	0.81	0.62	0.74

Incomplete seed sets are different in number of partitions and data quantity in per partition. So what effects would different incomplete seed sets have on ISK-means? Based on this question, firstly we changed the number of partitions but maintained seed quantity in per partition as 10%, and we got the experimental results as shown in Fig. 7 by means of operating ISK-means ^{m} and ISK-means ^{p} algorithms again.

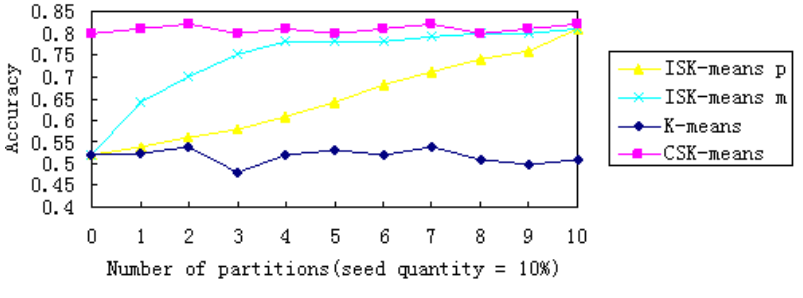

Fig. 7. Incomplete seed sets with different number of partitions

Fig. 7 shows that as the number of partitions increase in complete seed set, the performance of ISK-means ^{p} is a linear growth, whereas for ISK-means ^{m}, the improvement of performance is gradually weak when the number of partitions in incomplete seed set is close to that in complete seed set(actually when the number of partitions > 4). Thus, it can be seen that ISK-means ^{m} algorithm can exploit its advantages when the number of partitions in incomplete seed set is small.

Then we maintained the number of partitions be equal to 4 but changed the seed quantity in per partition, and we got the experimental results as shown in Fig. 8 by means of operating ISK-means ^{m} and ISK-means ^{p} algorithms again.

Fig. 8 shows that when the number of partitions is constant, increasing seed quantity in per partition can also improve performances of algorithms. ISK-means ^{p} can be improved significantly, because it has greatly narrowed down the random searching region with increase of seed quantity. However it is weak to assist ISK-means ^{m} in searching the $k-j$ initial cluster centers by increasing the quantity of seeds in the j seed sets of an incomplete seed set when 10% seeds in per partition have been obtained.

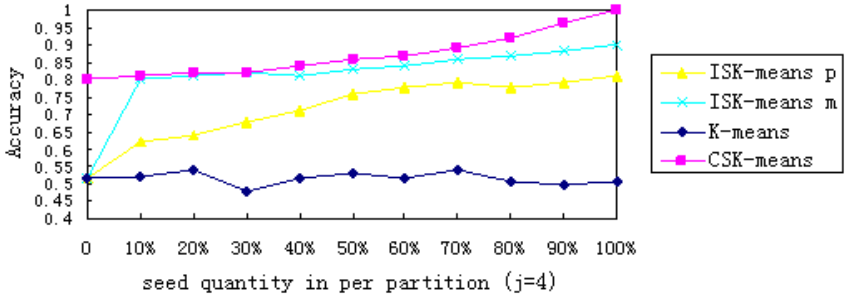

Fig. 8. Incomplete seed sets with the same number of partitions but different number of seeds

5 Conclusion

In this paper, we explained how semi-supervision can be incorporated into the k-means algorithm by means of optimizing initial cluster centers. We divided the supervised information into complete seed sets and incomplete seed sets, so that we could divide semi- supervised k-means into CSK-means and ISK-means. CSK-means can get the initial cluster centers by calculating means of a complete seed set.

However, ISK-means can only get part of initial cluster centers by calculating means of an incomplete seed set. ISK- means ^{m} gets the other centers by max-distance searching method, while ISK-means ^{p} gets them by randomly selecting. Experimental results demonstrated that CSK-means has great performance on complete seed set and ISK-means ^{m} has more advantages than k-means and ISK-means ^{p} on incomplete seed set.

References

1. Olivier, C., Bernhard, S., Alexander, Z.: Semi- Supervised learning, pp. 3–10. MIT Press, Cambridge (2006)
2. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: COLT 1998, Madison, WI, pp. 92–100 (1998)
3. Zhang, T., Ando, R.K.: Analysis of spectral kernel design based semi-supervised learning, pp. 1601–1608. MIT Press, Cambridge (2006)
4. Nizar, G., Michel, C., Nozha, B.: Unsupervised and semi-supervised clustering: a brief survey. In: Proc. of 6th Framework Programme (2005)
5. Basu, S., Bilenko, M., Mooney, R.: A probabilistic framework for semi-supervised clustering. In: Proc. of the 10th ACM SIGKDD Int'l. Conf. on Knowledge Discovery and Data Mining, pp. 59–68. ACM Press, Seattle (2004)
6. Tao, L., Hongjian, Y.: Semi-supervised learning based on k-means clustering algorithm. Application Research of Computers 27(3), 913–916 (2010)
7. Davidson, I., Basu, S.: Survey of clustering with instance level constraints. ACM Trans. on Knowledge Discovery from Data, 1–44 (2007)
8. Shi, Z.: Semi-supervised model based document clustering: a comparative study. Machine Learning 65(1), 3–29 (2006)
9. Basu, S., Banerjee, A., Mooney, R.J.: Semi- supervised clustering by seeding. In: Proc. of the 19th International Conference on Machine Learning, pp. 19–26 (2002)
10. Wagstaff, K., Cardie, C., Rogers, S.: Constrained k-means clustering with background knowledge. In: Proceedings of the 18th International Conference on Machine Learning, pp. 577–584. Morgan Kaufmann Publishers Inc., San Francisco (2001)
11. Blake, C., Keogh, E., Merz, C.J.: UCI repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine (1998), `http://archive.ics.uci.edu/ml/datasets.html`
12. Daoqiang, Z., Shiguo, C.: Experimental comparisons of semi-supervised dimensional reduction methods. Journal of Software 22(1), 28–43 (2011)
13. Xiao, Y., Jian, Y.: Semi-supervised clustering based on affinity propagation algorithm. Journal of Software 19(11), 2803–2813 (2008)
14. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing & Management 24(5), 513–523 (1988)