Semi-supervised Clustering Using Incomplete Prior Knowledge

Chao Wang, Weijun Chen, Peipei Yin, and Jianmin Wang

School of Software, Tsinghua University, Beijing 100084, P.R. China
chao-wang05@mails.thu.edu.cn

Abstract. Clustering algorithms incorporated with prior knowledge have been widely studied and many nice results were shown in recent years. However, most existing algorithms implicitly assume that the prior information is complete, typically specified in the form of labeled objects with each category. These methods decay and behave unstably when the labeled classes are incomplete. In this paper a new type of prior knowledge which bases on partially labeled data is proposed. Then we develop two novel semi-supervised clustering algorithms to face this new challenge. An empirical study performed on benchmark dataset shows that our proposed algorithms produce better results with limited labeled examples comparing with existing baselines.

Keywords: semi-supervised clustering, seeded clustering, clustering with prior knowledge.

1 Introduction

Semi-supervised clustering algorithms have recently been studied by many researchers with considerable interests. Although better results have been shown by some of the improved clustering algorithms, most of these methods depend heavily on the completeness of the prior knowledge. This means that all classes in the dataset need at least one labeled object which is also called complete seeding. However, in practical domains, usually only partial prior knowledge is provided that some unseeded categories exist. In such a clustering task, the existing algorithms such as Seeded-KMeans (Basu et al., 2002) tend to generate unsatisfied partitions to the data set because they need complete seeding knowledge.

In this paper we present two novel semi-supervised clustering algorithms (FS-KMeans and SS-KMeans) which can take use of the partial prior knowledge to estimate the rest parts of the seed set. Our experiments results show that the new algorithms can not only significantly boost the performance of semi-supervised clustering but also behave more stably on benchmark datasets.

The organization of this paper is as follows. In Section2 two novel semi-supervised clustering algorithms are proposed to solve the new challenge. Section3 presents a comparative study and discusses experimental results on two benchmark datasets, followed by concluding remarks in section4.

Y. Shi et al. (Eds.): ICCS 2007, Part I, LNCS 4487, pp. 192–195, 2007.

2 Algorithms

2.1 Seeded-KMeans

Seeded-KMeans (Basu et al., 2002) is a semi-supervised variant of KMeans, where initial background knowledge, provided in the form of labeled data points, is used in the clustering process. Thus, rather than initializing the KMeans from K random points, the mean of the lth cluster is initialized with the mean of the lth partition of the seed set . Then it repeats the point-assignment and recomputing means until convergence. Notice that the labels of the seeds could be changed in the assignment step. However, with a view to an incomplete seeding problem, we assume that S only contains data points from L classes ($L < K$). So $L < K$ classes have no labeled instances given already for the clustering task. As a result, the initial steps of the Seeded-KMeans and the Constrained-KMeans have to be adapted to this now problem.

2.2 Tow Incomplete-Seeding KMeans Algorithms

Noticing that the centroids of some clusters with seeds should reflect the distributions of the other clusters without seeds, we can utilize this information to choose better initial centers for unseeded clusters. Two more sophisticated algorithms we developed are given bellow.

Table 1. Algorithm:Farthest-Seeded-KMeans

Input Set of data points $X = \{x_1, x_2, \ldots, x_N\}, x_i \in R^d$, number of clusters , set $S_g = \cup_{l=1}^{L} S_l$ of initial seeds provided.

Output Disjoint K partitions $\{X\}_{l=1}^{K}$ of such that KMeans objective function is optimized

Method 1. Generating Step

(a) for the clusters with seeds, $\mu_h^{(0)} \leftarrow \frac{1}{S_h} \sum_{x \in S_h} x$,for $h = 1, \ldots, L$;

(b) choose $K-L$ data points farthest from any cluster center as new centroids $\mu_h^{(0)}$, for $h = L+1, L+2, \ldots, K$;

2. Iterating Step

Repeat until *convergence*

(a) *assignment*: assign each data point to the cluster h^*(i.e. set $X_{h^*}^{(t+1)}$) where $h^* = argmin_{h \in (1,\ldots,K)} \|x - \mu_h^{(t)}\|^2$

(b) *update*:recompute the means as $\mu_h^{t+1} \leftarrow \frac{1}{|X_h^{t+1}|} \sum_{x \in X_h^{t+1}} X$;

(c) $t \leftarrow t + 1$;

In the Farthest-Seeded-KMeans (FS-KMeans), the clustering process is divided into two steps which are the same as original KMeans. However, in the seeds generation, the partial seed set is used to produce initial centroids for the rest clusters. Then the point farthest from any cluster center at present is chosen

as the centroid for the cluster. We could also use some other criterions such as choosing the points with the largest standard deviation for one particular attribute. After repeating this for times, the initial centroids for the rest clusters are produced. The algorithm is presented in detail above.

Another algorithm we proposed, Splitting-Seeded-KMeans (SS-KMeans)is based on such an idea: to obtain clusters, split the set of all points into two clusters, select one of these clusters to split, and so on, until clusters have been produced. However, we have some seeds for clusters already, so we firstly generate a partitioning of the dataset, and then split these clusters to produce new clusters. In our experiments, we choose the cluster with the largest SSE to split.The details of SS-KMeans are given below.

Table 2. Algorithm:Splitting-Seeded-KMeans

Input Set of data points $X = \{x_1, x_2, \ldots, x_N\}, x_i \in R^d$, number of clusters , set $S_g = \cup_{l=1}^{L} S_l$ of initial seeds provided.

Output Disjoint K partitions $\{X\}_{l=1}^{K}$ of such that KMeans objective function is optimized

Method 1. Initialization a list Δ of clusters to contain the cluster consisting all points;

2. Pre-Clustering Step
   (a) Initialization by seeds:$\mu_h^{(0)} \leftarrow \frac{1}{S_h} \sum_{x \in S_h} x$, for $h = 1, \ldots, L$;
   (b) Generate L clusters using KMeans clustering with $\mu_h^{(0)}$;
   (c) Update the list Δ;
3. Splitting Step
   Repeat follow steps *until* K clusters are generated:
   (a) choose a cluster from list Δ;
   (b) split the cluster into two partitions by KMeans with $K = 2$;
   (c) Update the list Δ;
4. Refining Step
   (a) Initialize centroids with the means of clusters in list Δ;
   (b) Conduct KMeans on the whole dataset.

3 Experiments

The four clustering algorithms - FS-KMeans, SS-KMeans, Seeded-KMeans and random KMeans - are compared on high-dimensional text datasets (subsets of CMU 20-Newsgroups), with varying seeding, using mutual information as evaluation measure. In Seeded-KMeans, FS-KMeans and SS-KMeans, the seeds were selected from the dataset according to the corresponding seed fraction which vary from 0.1 to 1 in steps of 0.1. The four algorithms were compared with the unseeded categories increased from fully seeded to completely unseeded.

Table 3 summarizes the results of our experiments on Newsgroups data set when comparing FS-KMeans and SS-KMeans to Random KMeans and Seeded-KMeans. The number of unseeded categories varied from 0 (i.e. complete seeding) to 5 (i.e. no category had seed set). Clearly, both FS-KMeans and SS-KMeans outperforms their unsupervised and semi-supervised learning counterparts when the prior knowledge is incomplete in the form of partially seeded. However, the FS-KMeans fluctuates in a range because when the number of unseeded categorizes increases, because FS-KMeans has a tendency to choose outliers as the candidate initial centers because it picks the farthest point rather than points in a dense region. Comparatively, SS-KMeans did not only result good MI measures but was also more stable than the other methods even when all the categories were unseeded. This is mainly owed to the Refine Step in the SS-KMeans which guarantees to find a minimum to the objective function.

Table 3. Performance comparisonon 20-Newsgroups dataset with incomplete seeding

unseeded categories	0	1	2	3	4	5
Random KMeans	0.516	0.499	0.508	0.501	0.496	0.497
Seeded-KMeans	0.588	0.581	0.577	0.571	0.563	0.551
FS-KMeans	0.588	0.586	0.582	0.579	0.569	0.563
SS-KMeans	0.615	0.613	0.608	0.605	0.601	0.579

4 Conclusion

We have presented two novel algorithms FS-KMeans and SS-KMeans which effectively utilize the partial prior knowledge provided as incomplete seed set. Experimental results on benchmark datasets show that: (a) the two novel algorithms can estimate the underlying seeds more reasonably than the original seeded KMeans. (b) SS-KMeans does not only produce superior performance but also behave stably when the unseeded categorizes increase.

References

1. Basu, S., Banerjee, A., & Mooney, R. J.:Semi-supervised clustering by seeding. Proceedings of 19th International Conference on Machine Learning (ICML-2002) 19-26
2. Bilmes, J.: A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. Tech. rep. (1997) ICSI-TR-97-021, ICSI.
3. Bilenko, M., Basu, S., & Mooney, R. J.: Integrating constraints and metric learning in semi-supervised clustering. Proceedings of 21st International Conference on Machine Learning (ICML-2004)
4. Wagstaff, K., Cardie, C., Rogers, S., & Schroedl, S.: Constrained K-Means clustering with background knowledge. Proceedings of 18th International Conference on Machine Learning (ICML-2001) 577-584