# Background Knowledge Integration in Clustering Using Purity Indexes

Germain Forestier, and Cédric Wemmert and Pierre Gançarski

Image Sciences, Computer Sciences and Remote Sensing Laboratory
University of Strasbourg, France
forestier,wemmert@unistra.fr
https://lsiit-cnrs.unistra.fr/

**Abstract.** In recent years, the use of background knowledge to improve the data mining process has been intensively studied. Indeed, background knowledge along with knowledge directly or indirectly provided by the user are often available. However, it is often difficult to formalize this kind of knowledge, as it is often dependent of the domain. In this article, we studied the integration of knowledge as labeled objects in clustering algorithms. Several criteria allowing the evaluation of the purity of a clustering are presented and their behaviours are compared using artificial datasets. Advantages and drawbacks of each criterion are analyzed in order to help the user to make a choice among them.

**Key words:** Clustering, background knowledge, semi-supervised algorithm, purity indexes

## 1 Introduction

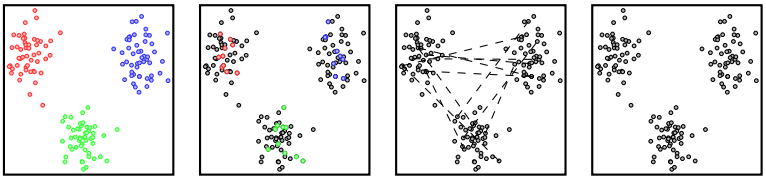
Knowledge integration to guide the clustering process is a major issue in data mining and an active research area. Indeed, fully unsupervised approaches raise some problems when dealing with more and more complex data. Moreover, background knowledge on the studied data are often available. Thus, it is important to work on proposing new approaches (semi-supervised methods) able to deal with such knowledge, to produce better results and to enhance the performance of the algorithms (speed-up, quality of the solutions, etc.).

The background knowledge can be represented in many different ways as they are strongly dependent on the studied domain. Even the number of clusters to find can be considered as knowledge on the data. Many works [1,2] addressed the problem of using background knowledge, represented as constraints between two objects of the dataset. These constraints give the information that the two objects have to be in the same cluster (*must-link*) or, on the contrary, that they should not be in the same cluster (*cannot-link*). Labeled samples can also be considered as another kind of knowledge. In a similar way than supervised classification methods that learn a classification function from a learning set composed of labeled objects, this information can be used during the clustering process to guide the algorithm towards a solution respecting this knowledge. It

is not necessary to have many labeled samples as in the supervised case, and they do not have to belong to each class of the problem.

In this context, the concept of *purity* of the clusters is very important. The purity evaluates the quality of the clusters according to the labeled samples available. A cluster is considered pure if it contains labeled objects from one and only one class. Inversely, a cluster is considered as impure if it contains labeled objects from many different classes.

The purpose of this article is to present and compare many different ways to evaluate the purity of the clusters. In the section 2, we give a state of the art about knowledge integration in data mining to introduce the context of this study. Then, in section 3, purity indexes are formalized and compared. Finally, we draw conclusions and give some directions of future work.



(a) Supervised (b) Partially labeled. (c) Partially constrained. (d) Unsupervised

**Fig. 1.** Example of different kinds of background knowledge.

## 2 Clustering with background knowledge

Many approaches have been investigated to use background knowledge to guide the clustering process.

In constrained clustering, knowledge is expressed as *must-link* and *cannot-link* constraints and is used to guide the clustering process. A *must-link* constraint gives the information that two data objects should be in the same cluster, and *cannot-link* means the opposite. This kind of knowledge is sometimes easier to obtain than a classical subset of labeled samples. Wagstaff et al. [1] presented a constrained version of the KMEANS algorithm which uses such constraints to bias the assignment of the objects to the clusters. At each step, the algorithm tries to agree with the constraints given by the user. These constraints can also be used to learn a distance function biased by the knowledge about the links between the data objects [2]. The distance between two data objects is reduced for a must-link and increased for a cannot-link. Huang et al. [3] presented an active learning framework for semi-supervised document clustering with language modeling. The approach uses a gain-directed document pair selection method to select cleverly the constraints. In order to minimize the amount of constraints

required, Griga et al. [4] defined an active mechanism for the selection of candidate constraints. The active fuzzy constrained clustering method is presented and evaluated on a ground truth image database to illustrate that the clustering can be significantly improved with few constraints. Recent works on constrained clustering are focused on evaluating the utility (i.e the potential interest) of a set of constraints [5,6].

Kumar and Kummamuru [7] introduced another kind of knowledge through a clustering algorithm that uses supervision in terms of relative comparisons, e.g. $x$ is closer to $y$ than to $z$. Experimental studies on high-dimensional textual data sets demonstrated that the proposed algorithm achieved higher accuracy and is more robust than similar algorithms using pairwise constraints (*must-link* and *cannot-link*) for supervision. Klein et al. [8] allowed instance-level constraints (i.e. must-link, cannot-link) to have space level inductive implications in order to improve the use of the constraints. This approach improved the results of the previously studied constrained KMEANS algorithms and generally requires less constraints to obtain the same accuracies. Basu et al. [9] presented a pairwise constrained clustering framework as well as a new method for actively selecting informative pairwise constraints, to get improved clustering performance. Experimental and theoretical results confirm that this active querying of pairwise constraints significantly improves the accuracy of clustering, when given a relatively small amount of supervision.

Another way to integrate background knowledge is to use a small set of labeled samples. Basu et al. [10] used a set of samples to *seed* (i.e. to initialize) the clusters of the KMEANS algorithm. Two algorithms, SEEDED-KMEANS and CONSTRAINED-KMEANS, are presented. In the first one, the labeled samples are used to initialize the clusters and the clusters are updated during the clustering process such as in the KMEANS algorithm. In the second one, the labeled samples used during the initialization stay in their assigned cluster, and only the unlabeled samples can change of cluster during the cluster affectation step of KMEANS. The choice between these two approaches must be done according to the knowledge about noise in the dataset.

To tackle the problem of incorporating partial background knowledge into clustering, when the labeled samples have moderate overlapping features with the unlabeled data, Gao et al. [11] formulated a new approach as a constrained optimization problem. The authors introduced two learning algorithms to solve the problem, based on hard and fuzzy clustering methods. An empirical study shows that the proposed algorithms improve the quality of clustering results despite a limited number of labeled samples. Basu et al. [12] also proposed a probabilistic model for semisupervised clustering, based on HIDDEN MARKOV RANDOM FIELDS (HMRF), that provides a principled framework for incorporating supervision into prototype-based clustering. Experimental results on several text data sets demonstrate the advantages of this framework.

Another approach, called supervised clustering [13], uses the class information about the objects as an additional feature, to build clusters with a high class-based purity. The goal of supervised clustering is to identify class-uniform

clusters having high probability densities. Supervised clustering is used to create summaries of datasets and for enhancing existing classification algorithms.

Different kinds of background knowledge are introduced by Pedrycz et al. [14], namely partial supervision, proximity-based guidance and uncertainty driven knowledge hints. The authors discuss about different ways of exploiting and effectively incorporating these background knowledge (known as *knowledge hints*) in the fuzzy c-means algorithm. In [15], Bouchachia and Pedrycz presented an extension of the fuzzy collaborative clustering which takes into account background knowledge through labeled objects. One of the advantages of the method is to take into account the classes split in several clusters. During the collaboration step, the method identify if a class correspond to various clusters and add or remove clusters according to this information. More recently, Pedrycz [16] presented some concepts and algorithms to collaborative and knowledge-based fuzzy clustering. The FUZZY C-MEANS algorithm (FCM) was used as an operational model to explain the approach. Interesting linkages between information granularity, privacy and security of data in collaborative clustering were also discussed. The problem of data privacy when clustering multiple datasets was also recently discussed in [17]. An application of fuzzy clustering with partial knowledge to organize and classify digital images is also proposed in [18]. The author present an operational framework of fuzzy clustering using the FUZZY C-MEANS algorithm with an augmented objective function using background knowledge. Experiments are carried out on collections of images composed of 2000 images.

## 3 Clustering evaluation

The evaluation of the purity or the quality of a clustering consists in determining if the repartition of the objects in the different clusters is coherent with the available knowledge on the data. We consider here the knowledge as a set of labeled objects. Let us define some notations to formalize the purity indexes:

- Let $N$ be the number of labeled samples
- Let $\mathbb{C} = \{c_1, c_2, \ldots, c_K\}$ be the clusters found by the clustering algorithm
- Let $\mathbb{W} = \{w_1, w_2, \ldots, w_C\}$ be the classes of the labeled objects
- Let $c_k$ be the objects composing cluster $k$ and $w_k$ the objects composing class $k$
- Let $|c_k|$ be the cardinal of cluster $k$
- Let $n_j^i = |w_i \cap c_j|$ be the number objects of cluster $i$ being in class $j$

### 3.1 Purity evaluation

The easiest way to compute the purity of a clustering is to find the majority class in each cluster and to count the number of labeled objects of this class in each cluster [19]. Then, the purity can be defined as:

$$\boldsymbol{\Pi}_{\text{simple}}(\mathbb{C}, \mathbb{W}) = \frac{1}{N} \sum_{i}^{K} \arg\max_{j}(n_j^i) \tag{1}$$

This evaluation of the purity consists in estimating the percentage of labeled objects of the majority class in each cluster for all the clustering. It takes its value in $[0; 1]$, 1 indicating that all clusters are pure, i.e. they contain only labeled objects of one class.

Another way to estimate the clusters purity is proposed by Solomonoff et al. [20]. The authors define the purity as the probability that, given a cluster $i$ and two randomly chosen labeled objects of this cluster, they both are of the same class $j$. The probability that the class of the first object is $j$ is $n_j^i/|c_i|$. The probability that the class of the second object is also $j$ is $(n_j^i/|c_i|)^2$. Finally, the purity of a cluster $i$ can be defined as:

$$\pi_{\text{prob}}(c_i) = \sum_{j}^{C} \left(\frac{n_j^i}{|c_i|}\right)^2 \tag{2}$$

which can derived to a clustering by:

$$\boldsymbol{\Pi}_{\text{prob}}(\mathbb{C}, \mathbb{W}) = \frac{1}{N} \sum_{i}^{K} |c_k| \pi_{\text{prob}}(c_i) \tag{3}$$

The advantage of this measure, compared to the simple purity evaluation (1), is to take into account the distribution of all the classes in the cluster, and not only the majority class. Thus, it promotes clusters composed of labeled samples from a limited number of classes. Its value is in $[0; 1]$, 1 indicating that all the clusters are pure.

However, these two purity indexes present a major drawback. They over evaluate the quality of a clustering having a large number of clusters. Indeed, the purity is maximal when having one cluster per objects (which is generally not considered as a *good* solution). De facto, if these measures are used in an algorithm allowing the number of clusters to change, it will tend to converge to a solution having to many clusters. Many propositions have been given to cope with this problem. In [21], Ajmera et al. have proposed to calculate the clusters purity according to their composition in terms of classes, but also the purity of the classes in terms of clusters (for each class, its distribution among all the clusters is observed). Then, the two values are merged to become the purity evaluation of the clustering. This enables to penalize solutions proposing too many clusters. The classes purity is computed in the same way as the clusters purity:

$$\pi_{\widetilde{\text{prob}}}(w_i) = \sum_{j}^{C} \left(\frac{n_j^i}{|w_i|}\right)^2 \tag{4}$$

which gives the following definition for all the clusters of a clustering:

$$\boldsymbol{\Pi}_{\widetilde{\text{prob}}}(\mathbb{C}, \mathbb{W}) = \frac{1}{N} \sum_{i}^{K} |c_k| \pi_{\widetilde{\text{prob}}}(w_i) \tag{5}$$

The clusters purity and the classes purity are then combined as follows:

$$\boldsymbol{\Pi}_{\text{overall}}(\mathbb{C}, \mathbb{W}) = \sqrt{\boldsymbol{\Pi}_{\text{prob}}(\mathbb{C}, \mathbb{W}) \times \boldsymbol{\Pi}_{\widetilde{\text{prob}}}(\mathbb{C}, \mathbb{W})} \tag{6}$$

Another approach consists in also considering a quality measure of the clustering. Demiriz et al. [22] used an optimization algorithm with a purity index called GINI which is similar to the criterion given in 2. To avoid this case, the algorithm generates solutions with too many clusters, the objective function to optimize is an arithmetic mean between the clusters purity and quality. The quality of the clusters is evaluated according to Davies-Bouldin index [23], which promotes well separated compact clusters. The combination of these two criteria enables to avoid extreme solutions (e.g. one cluster for each object).

Finally, Eick et al. [13] proposed to use a penalty criterion, to penalize solutions having too many clusters. The penalty is calculated as follows:

$$\text{penalty}(K) = \begin{cases} \sqrt{\frac{K-C}{N}} \text{ si } K \geq C \\ 0 \qquad\quad \text{sinon} \end{cases} \tag{7}$$

with $K$ the number of clusters, $C$ the number of classes and $N$ the number of objects. It can be used with any purity index, as the simple criteria (1):

$$\boldsymbol{\Pi}_{\text{penalty}}(\mathbb{C}, \mathbb{W}) = \boldsymbol{\Pi}_{\text{simple}}(\mathbb{C}, \mathbb{W}) - \beta \text{penalty}(K) \tag{8}$$

Another solution is to evaluate the Normalized Mutual Information (NMI) index between available knowledge and the clustering:

$$\boldsymbol{\Pi}_{\text{nmi}}(\mathbb{C}, \mathbb{W}) = \frac{I(\mathbb{C}, \mathbb{W})}{[H(\mathbb{C}) + H(\mathbb{W})]/2} \tag{9}$$

$I$ is the mutual information:

$$I(\mathbb{C}, \mathbb{W}) = \sum_i \sum_j n_j^i \log \frac{n_j^i}{|c_i|/N \times |w_j|/N} \tag{10}$$

$$= \sum_i \sum_j \frac{n_j^i}{N} \log \frac{n_j^i}{|c_i| \times |w_j|} \tag{11}$$

$H$ is the entropy:

$$H(\mathbb{W}) = -\sum_k \frac{|w_k|}{N} \log \frac{|w_k|}{N} \tag{12}$$

The mutual information $I$ (10) evaluates the quantity of information providing by the clustering on the classes. The denominator in (9) enables to normalize the criterion which value is in $[0; 1]$, 1 indicating pure clusters. This index is maximal when the number of clusters is equal to the number of classes. Thus, it does not have the drawback of the previous criteria presented above.

### 3.2 Partitions comparison

Another commonly used criterion to compare partitions is the *rand* index [24]. It consists in comparing pairs of objects and to check if they are classified identically in two partitions. In our case, we verify if each pair of objects identically labeled according to the background knowledge are in the same cluster. A pair of objects is a *true positive* (TP) if the two objects have the same label and are in the same cluster. It is a *true negative* (TN) if they have different labels and are in different clusters. A *false positive* (FP) corresponds to a pair of objects having different labels but in the same cluster, whereas a *false negative* (FN) corresponds to a pair of objects having the same label but being in two different clusters. The *rand* index can then be defined as:

$$\boldsymbol{\Pi}_{\text{rand}}(\mathbb{C}, \mathbb{W}) = \frac{TP + TN}{TP + FP + FN + TN} \tag{13}$$

$(TP + FP + FN + TN)$ representing all pairs of objects and $(TP + TN)$ all pairs of objects correctly classified. One drawback of this index, is that a same weight is given to false positives and false negatives.

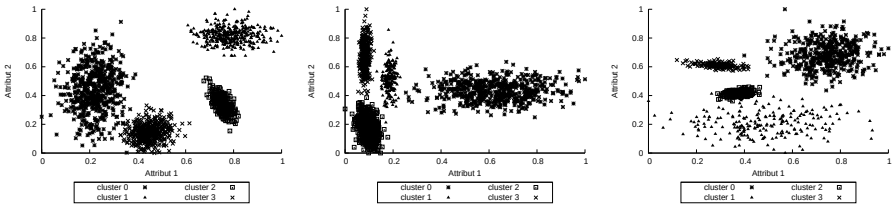
Regarding the *F-Measure* [25], it enables to affect weights to these values, according to the precision (P) and the recall (R):

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN}$$

$$\boldsymbol{\Pi}_{\text{fmeasure}}(\mathbb{C}, \mathbb{W}) = \frac{(\beta^2 + 1)P \times R}{\beta^2 P + R} \tag{14}$$

The $\beta$ parameter can be used to more penalize the false negatives as the false positives, giving it a value over one ($\beta > 1$). If $\beta = 1$, the precision and recall have the same importance.

The advantage of these two criteria ($\boldsymbol{\Pi}_{\text{rand}}$ and $\boldsymbol{\Pi}_{\text{fmeasure}}$) is that they implicitly integrate the number of clusters, putting the solutions proposing to many clusters at a disadvantage. Indeed, the more the number of clusters is increased, the more the pairs of objects differ from the available knowledge.




(a) 2d-4c-no0. (b) 2d-4c-no1. (c) 2d-4c-no2.

**Fig. 2.** The three datasets used for the evaluation.

### 3.3 Evaluation of the quality criteria

In this section, the criteria presented before are going to be evaluated on different datasets. Figure 2 shows three artificial datasets, each representing four clusters in a two dimension space. The KMEANS algorithm was used on these data, with a number of clusters varying from 2 to 8. For each clustering, the different measures presented in the previous sections were calculated. Three configurations have been evaluated, the first with 1% of labeled objects, the second with 10% of labeled objects and the last one with 25% of labeled objects in the dataset. Each experiment has been ran 100 times with random initialization, and the results were averaged. Figures 3 (a), (b) and (c ) give the results respectively for the dataset presented in figure 2(a), 2(b) and 2(c).

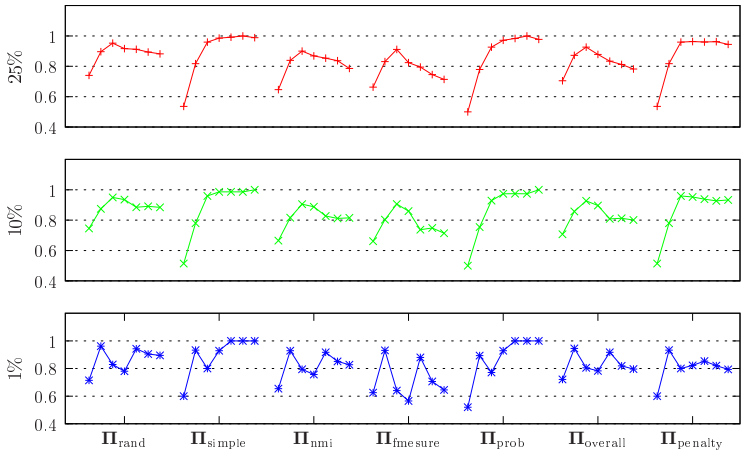
One can observe that when only few labeled objects are available (1%), quite all the criteria have an unpredictable behaviour. Indeed, it is not guarantee that the labeled set contains examples of all classes. That is why these criteria can hardly be used when only few knowledge is available. When the number of labeled objects increases (10%), the probabilty to have samples of each class in the labeled set also increases. Therefore, the evolutions of the criteria are more typical. One can observe the already mentioned problem that some purity measures overevaluate the quality of the clustering when the number of clusters increase. Indeed, the simple purity index ($\boldsymbol{\Pi}_{\text{simple}}$) and the cluster purity index ($\boldsymbol{\Pi}_{\text{prob}}$) increase as the number of clusters increase. The other criteria ($\boldsymbol{\Pi}_{\text{rand}}, \boldsymbol{\Pi}_{\text{nmi}}, \boldsymbol{\Pi}_{\text{fmeasure}}, \boldsymbol{\Pi}_{\text{overall}}, \boldsymbol{\Pi}_{\text{penalty}}$) tend to decrease as the number of clusters increase. The most characteristic are $\boldsymbol{\Pi}_{\text{fmeasure}}$, $\boldsymbol{\Pi}_{\text{overall}}$ and the $\boldsymbol{\Pi}_{\text{nmi}}$. The criteria $\boldsymbol{\Pi}_{\text{rand}}$ and $\boldsymbol{\Pi}_{\text{penalty}}$ decrease less significantly. It is interesting to notice that there is no noticeable difference between the results obtained with 10% or 25% of labeled objects.

## 4 Conclusion

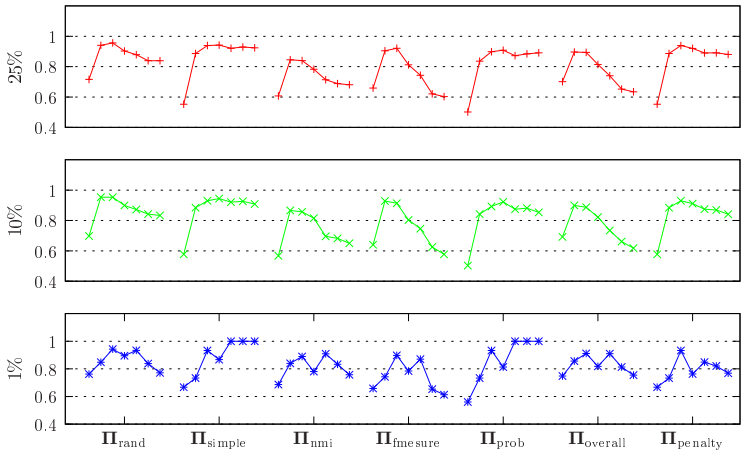
Knowledge integration in clustering algorithms is a really important issue. As more and more knowledge are available on the data manipulated, it is necessary to propose new approaches that enables to deal with this huge amount of information.

In this article, we have presented how to take advantage of labeled objects to evaluate the purity of a clustering. Many criteria were exhibited, formalized and compared. One observation is that purity evaluation without taking into account the number of clusters tends to overevaluate the quality of the results. To cope with this problem, it is possible to penalize results with a huge number of clusters. Another type of criteria only compare how pairs of objects were classified, as the *F-Measure* which has given particularly good results in our experiments.

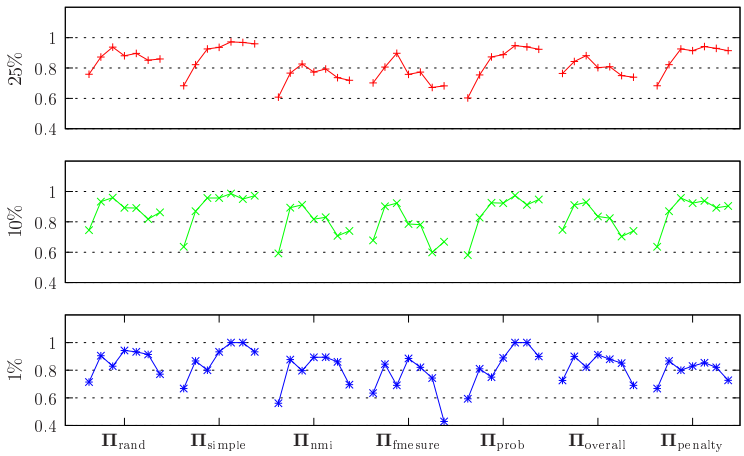
In the future, we aim to evaluate more criteria and to compare other types of domain knowledge, as for example constraints on the objects of the dataset. Moreover, it would be necessary to study the behaviour of these criteria when labeled objects of a same class belongs to different clusters.




(a) Criteria evolution according to the number of clusters for the dataset Fig. 2(a)




(b) Criteria evolution according to the number of clusters for the dataset Fig. 2(b).




(c) Criteria evolution according to the number of clusters for the dataset Fig. 2(c).

**Fig. 3.** Criteria evolution.

## References


1. Wagstaff, K., Cardie, C., Rogers, S., Schroedl, S.: Constrained k-means clustering with background knowledge. In: International Conference on Machine Learning. (2001) 557–584
2. Bilenko, M., Basu, S., Mooney, R.J.: Integrating constraints and metric learning in semi-supervised clustering. In: International Conference on Machine Learning. (2004) 81–88
3. Huang, R., Lam, W.: An active learning framework for semi-supervised document clustering with language modeling. Data & Knowledge Engineering **68**(1) (2009) 49–67
4. Grira, N., Crucianu, M., Boujemaa, N.: Active semi-supervised fuzzy clustering. Pattern Recognition **41**(5) (2008) 1851–1861
5. Davidson, I., Wagstaff, K.L., Basu, S.: Measuring constraint-set utility for partitional clustering algorithms. In: European Conference on Principles and Practice of Knowledge Discovery in Databases. (2006) 115–126
6. Wagstaff, K.L.: Value, cost, and sharing: Open issues in constrained clustering. In: International Workshop on Knowledge Discovery in Inductive Databases. (2007) 1–10
7. Kumar, N., Kummamuru, K.: Semisupervised clustering with metric learning using relative comparisons. IEEE Transactions on Knowledge and Data Engineering **20**(4) (2008) 496–503
8. Klein, D., Kamvar, S., Manning, C.: From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In: In The Nineteenth International Conference on Machine Learning. (2002) 307–314
9. Basu, S., Banerjee, A., Mooney, R.J.: Active semi-supervision for pairwise constrained clustering. In: SIAM International Conference on Data Mining. (2004) 333–344
10. Basu, S., Banerjee, A., Mooney, R.J.: Semi-supervised clustering by seeding. In: International Conference on Machine Learning. (2002) 19–26
11. Gao, J., Tan, P., Cheng, H.: Semi-supervised clustering with partial background information. In: SIAM International Conference on Data Mining. (2006) 489–493
12. Basu, S., Bilenko, M., Mooney, R.J.: A probabilistic framework for semi-supervised clustering. In: International Conference on Knowledge Discovery and Data Mining. (2004) 59–68
13. Eick, C.F., Zeidat, N., , Zhao, Z.: Supervised clustering - algorithms and benefits. In: International Conference on Tools with Artificial Intelligence. (2004) 774–776
14. Pedrycz, W.: Fuzzy clustering with a knowledge-based guidance. Pattern Recognition Letters **25**(4) (2004) 469–480
15. Bouchachia, A., Pedrycz, W.: Data clustering with partial supervision. Data Min. Knowl. Discov. **12**(1) (2006) 47–78
16. Pedrycz, W.: Collaborative and knowledge-based fuzzy clustering. International Journal of Innovative, Computing, Information and Control **1**(3) (2007) 1–12
17. Fung, B.C., Wang, K., Wang, L., Hung, P.C.: Privacy-preserving data publishing for cluster analysis. Data & Knowledge Engineering **68**(6) (2009) 552 – 575
18. Loia, V., Pedrycz, W., Senatore, S.: Semantic web content analysis: A study in proximity-based collaborative clustering. Fuzzy Systems, IEEE Transactions on **15**(6) (2007) 1294–1312
19. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)

20. Solomonoff, A., Mielke, A., Schmidt, M., Gish, H.: Clustering speakers by their voices. In: Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on. Volume 2. (May 1998) 757–760
21. Ajmera, J., Bourlard, H., Lapidot, I., McCowan, I.: Unknown-multiple speaker clustering using hmm. In: International Conference on Spoken Language Processing. (September 2002) 573–576
22. Demiriz, A., Bennett, K., Embrechts, M.: Semi-supervised clustering using genetic algorithms. In: Intelligent Engineering Systems Through Artificial Neural Networks. (1999) 809–814
23. Davies, D., Bouldin, D.: A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-1**(2) (1979) 224–227
24. Rand, W.M.: Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association **66** (1971) 622–626
25. van Rijsbergen, C.J.: Information Retrieval. Butterworths, London (1979)