URN: urn:nbn:de:bvb:473-irb-466507
DOI: https://doi.org/10.20378/irb-46650

Combining Supervised and Unsupervised Learning—An Ensemble Learning Framework Proposal

Michael Siebers
Otto-Friedrich-Universität
D-96045, Bamberg, Germany
michael.siebers@uni-bamberg.de

Abstract

Combining supervised and unsupervised learning is no new idea. Nevertheless, this was mainly done in a semi-supervised fashion. I will present a first approach to a framework using both methods on a classification task without using unlabeled data. I will evaluate the framework on an artificial data set and provide further research plans and ideas.

1 Introduction

In recent years combination possibilities of supervised and unsupervised learning algorithms have been explored. Generally this was done in the semi-supervised learning framework [Zhu, 2010]. There hypotheses learned from labeled data are fostered by exploiting characteristics of unlabeled data [Chawla and Karakoulas, 2005; Wemmert *et al.*, 2009].

However, to the best of my knowledge no such approach exists that uses labeled data only. I will develop a straight forward framework to use a clustering algorithm to improve supervised learning. The first, basic approach to this framework deals with classification learning. It will be presented in the next section. In Section 3 I will give a prove of concept on an artificial data set. I will conclude with further plans and ideas in Section 4.

2 Framework

The core of the proposed framework is to have an ensemble of classifiers and to apply them according to a clustering. At first I will show how these clustering and classifiers can be learned. Afterwards I will detail the application.

2.1 Preliminaries

Each data point or instance x belongs to one class $k \in \mathbf{K}$. All possible instances form the instance space $\mathbf{X}$.

A hypothesis h is a mapping between instance space $\mathbf{X}$ and classes $\mathbf{K}$:

$$\mathrm{h}\colon \mathbf{X} \to \mathbf{K}\,.$$

All possible hypotheses are combined in the hypothesis space $\mathbf{H}$.

A learning algorithm Λ takes a number of labeled instances $\mathbf{D}$ and learns a hypothesis h:

$$\Lambda\colon \mathcal{P}\left(\mathbf{X} \times \mathbf{K}\right) \to \mathbf{H}\,,$$

where the training set $\mathbf{D}$ is a set of instances with associated class label

$$\{(x_i, y_i)\}_{i=1}^{N}, y_i \in \mathbf{K}\,.$$

A clustering $\mathrm{f_c}$ is a function which assigns each instance x to a cluster $c \in \mathbf{C}$:

$$\mathrm{f_c}\colon \mathbf{X} \to \mathbf{C}\,.$$

No cluster may be empty ($\forall c \in \mathbf{C} \exists x \in \mathbf{X}\ \mathrm{f_c}(x) = c$). The set of possible clusterings is called clustering space $\mathbf{F_c}$.

A clustering algorithm Γ learns a clustering from given unlabeled instances: $\Gamma\colon\ \mathcal{P}(\mathbf{X}) \to \mathbf{F_c}$.

2.2 Training and Application

Learning a clustering ensemble hypothesis S given training data $\mathbf{D}$ is a three step process:

1. Apply the clustering algorithm to the training set without the associated labels,
2. divide the training set according to the clusters, and
3. apply the learning algorithm to each training set slice.

This results in an ensemble of one clustering and $|\mathbf{C}|$ hypotheses. A pseudocode algorithm of the training phase can be seen in Algorithm 1.

Input: A learning algorithm Λ
Input: A clustering algorithm Γ
Input: A training set $\mathbf{D} = \{(x_i, y_i)\}_{i=1}^{N}$
$\mathbf{D}' \leftarrow \{x \mid (x, y) \in \mathbf{D}\}$
$\mathrm{f_c} \leftarrow \Gamma\left(\mathbf{D}'\right)$
foreach $c_t \in \mathbf{C}$ **do**
  $\mathbf{D}_t \leftarrow \{(x, y) \in \mathbf{D} \mid \mathrm{f_c}(x) = c_t\}$
  $\mathrm{h}_t \leftarrow \Lambda\left(\mathbf{D}_t\right)$
end
return $\left(\mathrm{f_c}, (\mathrm{h}_t)_{t=1}^{|\mathbf{C}|}\right)$

Algorithm 1: Training phase of clustering ensemble learning.

A clustered ensemble hypothesis $S\left(\mathrm{f_c}, (\mathrm{h}_t)_{t=1}^{|\mathbf{C}|}\right)$ is applied to an unseen instance x by first computing the cluster c_t the instance *best fits in* and then applying the hypothesis h_t assigned to this cluster. For centroid-based clusterings the calculation of the appropriate cluster is done by finding the closest cluster centroid—in terms of euclidean distance.

A pseudocode algorithm is provided in Algorithm 2. A flowchart depicting the whole process (training and application) is found in Figure 1.

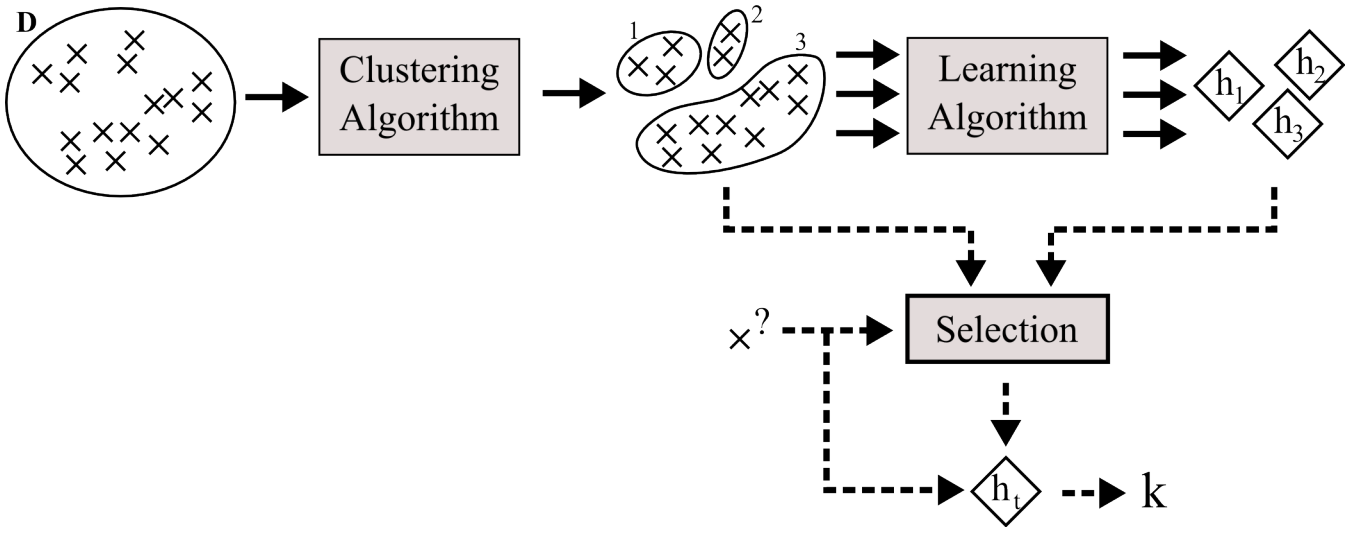

Figure 1: Flowchart of clustering ensemble learning. Training and application phase. Solid arrows correspond to the training phase, dashed arrows to the application phase.

Input: An instance x
Input: A clustering $\mathrm{f_c}$
Input: An ensemble of hypotheses $(\mathrm{h}_t)_{t=1}^{|\mathbf{C}|}$
$c \leftarrow \mathrm{f_c}(x)$
$\mathrm{h} \leftarrow \mathrm{h}_t$ such that $c = c_t$
return $\mathrm{h}(x)$

Algorithm 2: Application phase of clustering ensemble learning.

3 Artificial Example

The successful IT company *AComp* is about to launch its latest product *S-Thing*. The product placement will be accompanied by an advertising compaign. To assess which customers will be affected by the advertisement they conducted a survey in advance: They showed the intended advertisement to 200 of their customers. Afterwards the subjects rated whether they would buy the advertised product. Possible ratings were *presumably buy*, *perhaps buy*, and *presumable not buy*. Additionally the customers ages (*age*) and average annual expenditures (*expenditure*) were recorded. The acquired data can be seen in Figure 2.

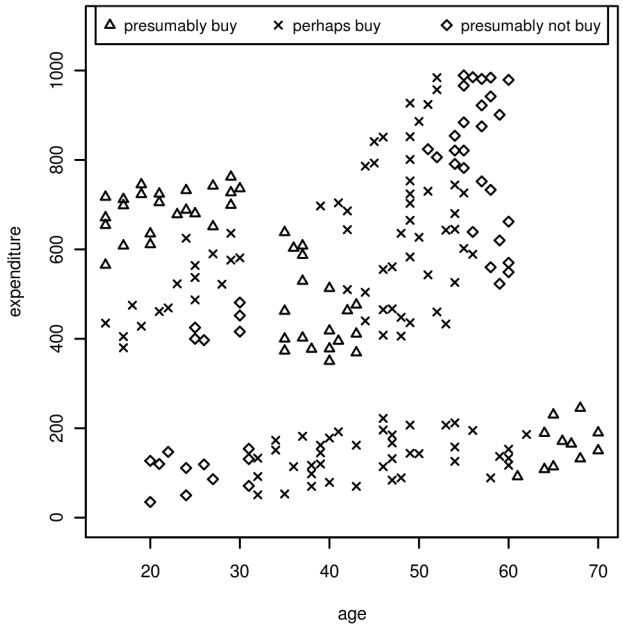

Figure 2: Data acquired in the *S-Thing* study.

3.1 Classifiers

In order to carry the results over to other customers they intended to learn a classifier and to apply it to their customer data base. To select a classifier and estimate its performance they conducted a 10-fold cross validation on the given data. Explored learning algorithms were Naïve Bayes (NB), C4.5, and 3-nearest neighbours (3NN). After inspecting the data—and noting the three separate clusters—they added clustering ensemble learning to this list. The same three algorithms were used as base learning algorithms, k-medoids was used as clustering algorithm.

For the 3NN learner the neighbours were weighted by their euclidean distance. In C4.5 the information gain was used for attribute selection; the pruning confidence was 0.25. In the clustering ensemble learning the data was split in 3 clusters—according to visual evidence. K-medoids used 100 k-means runs with $1,000$ optimization steps each. The performance estimation was repeated 20 times, results were averaged. As first step the data was normalized. Both *age* and *expenditure* were linearly transformed into the range from 0 to 1. The comparison was realized using RapidMiner $5.1.006$[1].

3.2 Results

The accuracy rates of the applied classifiers can be seen in Table 1. The best accuracy is achieved using the proposed framework using 3NN as base learning algorithm. Overall the proposed framework excelled over applying the learning algorithms alone. For both methods—using only the learning algorithm or the proposed framework—the nearest neighbours approach performs best.

[1]RapidMiner is an open source data mining tool provided by Rapid-I (`www.rapid-i.com`).

	classic	clustered
NB	.575	.788
C4.5	.896	.910
3NN	.909	.918

Table 1: Accuracies for the *S-Thing* data. The column *classic* refers to solely applying the learning algorithm, *clustered* denotes to the introduced framework.

The greates improve is gained for NB. Though the improvement is small for the other learning algorithms, even the best, localized algorithm can improve using the framework. It is evident that there are situations when applying the proposed framework increases performance. The characteristics of these situations (data, learning and clustering algorithms) are to be explored.

4 Outlook

The proposed framework is far from complete. Up to now it has two major limitations: Only clusterings partitioning the data are considered and only one hypothesis is selected for classification. Thus further furmulations of the framework will support other clustering approaches—like fuzzy clustering, hierarchical clustering, or overlapping clusterings—and will be able to include different hypothesis in the final classification.

The resulting framework will be evaluated using artificial and real world data. This will respect diverse learning algorithms, different classification algorithms, and varying hypothesis combination schemes. The results will be contrasted with major learning algorithms.

There are multiple extention or variation possibilities for the framework:

- The framework could be modified to allow for regression tasks,
- the framework could be carried over into the semi-supervised domain,
- learned clusters could have an inherent weight, modelling their reliability,
- sensibility for misclassification costs could be incorporated, or
- active learning methods could be explored.

Some of these will surely be part of my further work.

Finally one major component of my analysis will be the framework's resemblance of human concept learning. Depending on the clustering algorithm involved the intended framework might be similar to prototype or exemplar theories theirof [Medin and Schaffer, 1978].

References

[Chawla and Karakoulas, 2005] Nitesh V. Chawla and Grigoris Karakoulas. Learning from labeled and unlabeled data: An empirical study across techniques and domains. *Journal of Artificial Intelligence Research*, 23:331–366, 2005.

[Medin and Schaffer, 1978] Douglas L. Medin and Marguerite M. Schaffer. Context theory of classification learning. *Psychological Review*, 85(3):207 – 238, 1978.

[Wemmert *et al.*, 2009] Cédric Wemmert, Germain Forestier, and Sébastien Derivaux. Improving supervised learning with multiple clusterings. In Oleg Okun and Giorgio Valentini, editors, *Applications of Supervised and Unsupervised Ensemble Methods*, volume 245 of *Studies in Computational Intelligence*, pages 135–149, Berlin Heidelberg, 2009. Springer-Verlag.

[Zhu, 2010] Xiaojin Zhu. Semi-supervised learning. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning*. Springer Science+Business Media, 2010.