Semi-supervised Hidden Markov Random Fields (HMRF) Kmeans. Theory to Implementation

Dimitrios Pritsos and Efstathios Stamatatos

University of the Aegean
Karlovassi, Samos – 83200, Greece.
{dpritsos, stamatatos}@aegean.gr

1 Introduction

The objective of the Semi-supervised Clustering is to incorporate in the procedure of clusters discovery or assignment, the prior knowledge about the skeleton of the clusters schema. There are several efforts on Semi-supervised model inference in both Expectation Maximization (EM) clustering based models and in Agglomerating clustering based models. According to [3] there three main groups of EM based semi-supervised clustering methods:

1. *Constraints-based mehtods* are using the provided supervision for guiding the algorithm towards a data partitioning which is avoiding (but not prevening) the constraints violation.
2. *Distance-based approaches* in clustering method with a particular distance funciton; the distance function is parametrized and the parameters values are learned to satisfy the constraints.
3. *Semi-supervised clustering based on Hidden Markov Random Fields* where the constraint-based and distance-based approaches are combined into *a unified probabilistic model*.

In EM clustering based models there have been several efforts where the labeled data where provided in the clustering model in the form of data-labels pairs or in the initialization phase of theclustering model. In this work we present the Hidden Markov Random Fields Model(HMRF) Kmeans, where the prior knowledge about the structure or skeleton of the clusters schema has been embedded into the model in the form of constraints [1]. The HMRF Kmeans is a hard-clustering model due to the *hard* assignment of the data point to one of the a-priori fixednumber of clusters. However, the same models can be transformed into a relatively easy soft-clustering model where of each data point only Maximum a-posteriry Probability (MAP) of the point to bea member of each cluster of th final schema.

The HMRF Kmeans Semi-supervised clustering method it has been implemented, in this work, for being tested on the Web Genre Identification (WGI) Information Retrieval (IR) taxonomy problem.Therefore, here we only present the model inference procedure where the distance measure, a.k.a distortion function/measure, is the cosine similarity because in the IR literature is the distortionmeasure where in most cases maximizes performance in problems where the

feature space is particularly large, as in this case. In case one would be interested in changing the model to asoft-clustering method then the probability destiny function of the final model should have been chosen to be some properly parametrized Von Mises Fisher distributions.

Since this work is focusing on the implementation of Semi-supervised HMRF-Kmeans in the IR domain framework, it has to be noted that this Semi-supervised model advantage is the interactivelearning setting where this model can be used [3], since the constraints are provided to the model in two different sets the *Must-Link* and*Cannot-Link*. These sets are not necessarily the same in size or complimentary one to the other.

What it follows is the model inference line of thought based on the there resources [1, 3, 2].

2 Model inference

The objective of a semi-supervised model like HMRF Kmeans is to drive the procedure of clustering schema taking into account the prior knowledge we have about the clusters in the form of*must-link* and *cannot-link* constraints. The main difference in the graphical model of a topical EM algorithm is the nodes of the observed labeled data over the hidden, a.k.a latent,variables as depicted in fig.1, with red colored arrows and gray shaded nodes.

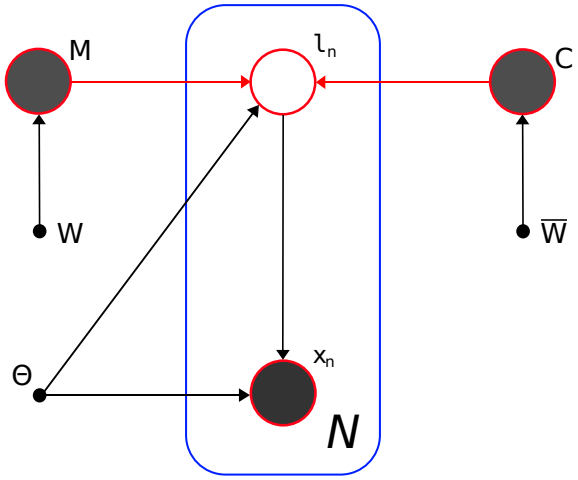

Fig. 1. Semi-supervised HMRF Expectation Maximization or Kmeans Clustering Graphical Model.

The goal of EM is to maximize the *log likelihood function* $P(X|\Theta)$ with respect of Θ as in eq.1.

$$lnP(X|\Theta) = ln\left\{\sum_{l} P(X, L|\Theta)\right\} \tag{1}$$

Where $X = \{\mathbf{x}_i\}_{i=1}^{N}$ is the set of *observable random variables* given by the conditional probability $P(X|\Theta)$ and $\mathbf{x}_i$ is a random vector (or data point) of the corpus under taxonomy. Note the boldface notation of the random vector in order not to be mixed with x_i which will a feature (or variable) of this vector. Moreover, N is the number of vector as depicted in the graphical model of fig.1.

Due to the relatively complex marginal distributions, like $P(X|\Theta)$, over observed data points where they are computationally intractable, there is a common practice to incorporate *latent or hidden variables* in order to express the conditional probability calculation more tractable over the expanded space of observed and latent variables. In eq.1 L is the set of hidden (or latent) variables over X observable data points [2].

Therefor *a hidden field* $L = \{\mathbf{l}_i\}_{i=1}^{N}$ random variables, *whose values are unobservable*. In the clustering framework, the set of hidden variables are the unobserved cluster labels on the points, indicating cluster assignments. Every hidden variable $\mathbf{l}_i$ takes values from the set $1, ..., N$, which are the labels of the clusters.

Now every random data point $\mathbf{x}$ can be generated from a conditional probability distribution $P(x_i|\mathbf{l}_i)$ determined by the corresponding hidden variable $\mathbf{l}_i$. Note that we know a-priory that the random data points X are conditionally independent given the hidden variables L. Thus $P(X|L) = \prod_{\mathbf{x}_i \in X} P(\mathbf{x}_i|\mathbf{l}_i)$. Note that $\mathbf{l}_i$ can be either a vector or a singleton depending on the algorithm setting, either for soft-clustering or for hard-clustering respectively. Thus value and vector for $\mathbf{l}_i$ will be used interchangeably in this text.

The set of $\{X, Z\}$ is *the complete data set*, while the $\{X\}$ is the incomplete data set. The likelihood function for the complete data set simply takes the form $lnp(X, Z|\Theta)$ as shown in eq.1, and theoretically that maximization of this complete data log likelihood function is straightforward. However, in practice we are not given the complete data set, but only the incomplete data points X. Thus, our only knowledge of for the values of the latent (hidden) variables in L is given by the posterior distribution $p(Z|X, \Theta)$. Because we don't have available the complete-data log likelihood, we consider instead its expected value under the posterior distribution of the latent (hidden) variables.

In eq.1 and in fig.1 there are some parameters Θ which are governing the initial and the final schema of the PDF's mixture. These are the parameters we have to find computationally and where in employee EM algorithm (alg.1) with the following general and distribution inexpedient (i.e. irrespectively where the PDF's are Gaussian, Von Mises Fisher etc). In particular with EM we are interactively finding the proper set of Θ parameters by calculating the expected posterior distribution $P(L|X, \Theta)$ of the latent (hidden) variables L.

Algorithm 1.1: The generic for of Expectation Maximization either for both Soft- and Hard-clustering

Data:
$\mathbf{X}$ observable data points.
$\mathbf{K}$ possible states clusters we expect to be existing in the data-set.
$\mathbf{L}_{i=1}^{K}$ the hidden field variables (or vectors) with values $\{1, ..., \mathbf{K}\}$ or $(0, 1]$
$\mathbf{\Theta}$ an initial state about the PDF mixture model.

Result:
$\mathbf{l} = \{\mathbf{p}_k\}_{k=1}^{K}$ where $\mathbf{p}$ can be either $\{0, 1\}$ or $(0, 1]$ depending on the soft-clustering of hard-clustering set-up.
$\mathbf{\Theta}$ the final set of parameters after the Probability Density Functions Mixture.

Choose an Initial setting for $\mathbf{\Theta}^{OLD}$;
while *I iterations reached* **do**
  1. E-step Evaluate $P(\mathbf{L}|\mathbf{X}, \mathbf{\Theta}^{OLD})$;
  2. M-step Evaluate $\mathbf{\Theta}^{\mathbf{NEW}}$ given by (a) and (b);
    a. $\mathbf{\Theta}^{NEW} = \arg_{\mathbf{\Theta}} \max \Omega(\mathbf{\Theta}, \mathbf{\Theta}^{OLD})$;
    b. $\Omega(\mathbf{\Theta}, \mathbf{\Theta}^{OLD}) = \sum_{\mathbf{L}} P(\mathbf{L}|\mathbf{X}, \mathbf{\Theta}^{OLD}) lnP(\mathbf{X}, \mathbf{L}, \mathbf{\Theta})$;
  if *log likelihood convergence reached* **then**
    Breaking the loop and Ending the Clustering;
  else
    $\mathbf{\Theta}^{OLD} \leftarrow \mathbf{\Theta}$;
  end
end

The EM algorithm can also be used to find MAP (maximum a-posterior) solutions in case we have a good knowledge about the prior distribution $P(\mathbf{\Theta})$ over the parameters $\mathbf{\Theta}$. In this case the E-step remains the same as in the maximum likelihood case, while in the *M-step (b)* the $\Omega(\mathbf{\Theta}, \mathbf{\Theta}^{OLD}) + lnP(\mathbf{\Theta})$.

In HMRF-Kmeans derivation process we are starting with the PDF mixture we would like to maximize by exploring the EM for the reasons explained above, as shown in eq.2.

$$P(\mathbf{X}, \mathbf{L}, \mathbf{\Theta}|\mathbf{M}, \mathbf{C}) = P(\mathbf{\Theta}|\mathbf{M}, \mathbf{C}) P(\mathbf{L}|\mathbf{\Theta}, \mathbf{M}, \mathbf{L}) P(\mathbf{X}|\mathbf{L}, \mathbf{\Theta}, \mathbf{M}, \mathbf{C}) \tag{2}$$

Where the set of vector are the same as in alg.1 but this time the constraints set of the fig.1 have been included, i.e. Mast-link and Cannot-link constraints set $\{\mathbf{M}, \mathbf{C}\}$. Moreover, as the graphical model is describing the constraints are independent from $\mathbf{X}$ and parameters $\mathbf{\Theta}$ are, also, independent from the constraints set. Thus:

$$P(\mathbf{L}|\mathbf{\Theta}, \mathbf{M}, \mathbf{L}) P(\mathbf{X}|\mathbf{L}, \mathbf{\Theta}, \mathbf{M}, \mathbf{C}) = P(\mathbf{X}|\mathbf{L}, \mathbf{M}, \mathbf{C}) \tag{3}$$

$$P(\mathbf{\Theta}|\mathbf{M}, \mathbf{C}) = P(\mathbf{\Theta}) \tag{4}$$

Considering $\mathbf{X}$ observable data-set is convient due to compuational issues to simplify the algorithm by assuming the vectors $\mathbf{x}$ are *mutually impediment*, thus:

$$P(\mathbf{X}|\mathbf{L}, \mathbf{\Theta}) = \prod_{i=1}^{N} P(\mathbf{x}_i|\mathbf{l}_i, \mathbf{M}, \mathbf{L}) \tag{5}$$

Consequentially, from equations 2, 3, 4 and 5 we are getting the following MAP which it should be maximize.

$$P(\mathbf{X}, \mathbf{L}, \mathbf{\Theta}|\mathbf{M}, \mathbf{C}) = P(\mathbf{\Theta})P(\mathbf{L}|\mathbf{\Theta}, \mathbf{M}, \mathbf{L}) \prod_{i=1}^{N} P(\mathbf{x}_i|\mathbf{l}_i, \mathbf{M}, \mathbf{L}) \tag{6}$$

At this step of the clustering method building process we have to decide weather the clustering would be soft or hard. This decision has two consequences. Firstly, is related to the $\mathbf{L}$ latent variables type and range of values as shown in alg.1, i.e. whether $\mathbf{l}$ will be a variable or a vector of variables and whether its values will be probability estimates or 0,1 depending whether of not a data-point is belonging to the cluster $\mathbf{k}_i$. Secondly, is related whether the algorithm will be probabilistic based or distance based, i.e. whether a MAP will be maximized or an objective function with a specific distance measure (a.k.a distortion function/measure) will be minimized.

In our case go for the *distortion function* path, as the Kmeans term of the algorithm implies. As explained above since we are focusing on IR domain problems we are going to show the complete algorithm building process for the cosine similarity as the distortion function of our choice.

Each hidden random variable $\mathbf{l}_i$ has an associated set of neighbors $\mathbf{\Gamma}_i \subset \mathbf{N}$. The must-link and cannot-link constraints define the neighborhood over the hidden labels, such that the neighbors of a point x_i are all points that must-linked and/or cannot-inked with. The *random field defined over the hidden variables* is a *Markov Random Field*, where the PDF of the hidden variables *obeys the following Markov property*:

i, Pr(li —L li) = Pr(li —l j : l j N i) (1)

The above paragraph justifies the name for the algorithm HMRF and is the .....

References

1. Basu, S., Bilenko, M., Mooney, R.J.: A probabilistic framework for semi-supervised clustering. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 59–68. ACM (2004)
2. Bishop, C.: Pattern Recognition and Machine Learning, pp. 439–447. Springer (2006)
3. Chapelle, O., Scholkopf, B., Zien, A., et al.: Semi-supervised learning. pp. 73–102. The MIT Press (2006)