KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 11–14

DECOMPOSITION METHODS FOR LABEL PROPAGATION

LEHEL CSATÓ AND ZALÁN BODÓ

Abstract. In semi-supervised learning we exploit the "information" provided by an unlabelled data-set, in addition to the usually small training data-set. A commonly used semi-supervised method is label propagation [Zhu and Ghahramani, 2002] where labels are *propagated* from labelled to unlabelled data by employing similarity measures.

The problem with the method is that it requires prohibitive time requirements, therefore when a large amount of unlabelled data is used, a feasible algorithm is needed to compute the labels. In this paper we propose an approximation to label propagation. We divide the original problem into sub-problems that are computationally less prohibitive. A decomposition into K parallel sub-problems is considered where the sub-problems randomly and sparingly communicate with each other.

1. Introduction

Semi-supervised learning [Zhu and Ghahramani, 2002; Bengio et al., 2006] can be viewed as a generalisation of the classical pattern recognition algorithm to data sets where only a fragment of the available data is labelled. The motivation is that data labelling is a time consuming *human* activity whilst – in contrast – collecting unlabelled samples is cheap leading to huge amounts of unlabelled data, a good example is the data from DNA arrays [Cristianini and Hahn, 2006] with only a tiny fraction processed, or the the huge document set from the internet, exploited by Google [Page et al., 1998]. In semi-supervised learning the training data is augmented with unlabelled data, *i.e.* $\mathcal{L} \cup \mathcal{U} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_\ell, \mathbf{y}_\ell), \mathbf{x}_{\ell+1}, \ldots, \mathbf{x}_{\ell+\mathbf{u}}\}$, where ℓ and $\mathbf{u}$ are the sizes of the labelled and unlabelled parts respectively. We assume $\ell \ll \mathbf{u}$ and we use $\mathbf{n} = \ell + \mathbf{u}$. The problem now is to assign labels to the unlabelled part, using the *information* present in the joint data-set $\mathcal{L} \cup \mathcal{U}$.

We consider the inputs, $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\mathbf{n}\}$, and infer the *density* of the inputs, and with further assumptions about the *structure* of the data, we complete the labelling process. Suppose for example that "professor" is a good predictor for the *study* category, based on the labelled data alone. Then, if the words "professor" and "university" are correlated in $\mathcal{X}$, detection accuracy is improved when using both words. We apply *label propagation* [Zhu and Ghahramani, 2002], to solve the problem. It is a similarity-based technique where the labels are propagated based on closeness between data items. In this article we propose an approximation to handle large data-sets using ideas from stochastic sampling.

2. Label propagation

Label propagation exploits the neighbourhood relation – the topology of the embedding space to construct a *graph* with nodes from $\mathcal{X}$ and edges encoding *similarities*; this graph *is used* to improve classification. Label propagation simulates a diffusion process that propagates the labels to neighbouring edges leading eventually to labels for the whole set $\mathcal{X}$. The graph is *fully connected* with edges weighted by the *degree of similarity* W_{ij} of the nodes $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$W_{ij} = \exp\left(-\frac{d(\mathbf{x}_i, \mathbf{x}_j)}{\alpha^2}\right) \quad \text{and} \quad \boldsymbol{W} \stackrel{\text{def}}{=} \{W_{ij}\}_{i,j=1}^{n} \tag{1}$$

where $d(\cdot,\cdot)$ is a distance between the points and α is the radius of similarity. Other distance measures are the cosine similarity, Jaccard coefficient, Dice coefficient, or the similarity in (1) [Luxburg, 2007; von Luxburg et al., 2007]. In the following we use bold capitals for matrices and bold lowercase for vectors, other quantities are scalars. We normalise the similarity matrix $\boldsymbol{W}$, to obtain transition probabilities:

$$P_{ij} \stackrel{\text{def}}{=} \mathbf{D}^{-1/2}\, \boldsymbol{W}\, \mathbf{D}^{-1/2} \quad \text{where} \quad D_{ii} = \sum_{j=1}^{n} W_{ij} \tag{2}$$

with $\mathbf{D}$ diagonal. The resulting graph can be fully connected (see above) or sparse, these are trimmed from full graphs by cutting edges with small weights [Zhu, 2005], here we use only full matrices. We define the following *label* matrices, assuming c classes (usually $c > 2$): $\mathbf{Y}_L$ an $(\ell \times c)$ matrix, each row corresponding to an item and each column to a category; $\mathbf{Y}_U$ a $(u \times c)$ matrix to be estimated; $\mathbf{Y} = [\mathbf{Y}_L^\top,\ \mathbf{Y}_U^\top]^\top$ an $(n \times c)$ matrix – the concatenation of the above two matrices. Label propagation propagates labels using the following steps [Zhu and Ghahramani, 2002]:

(1) compute $\mathbf{Y}(t+1) = \mathbf{P}\, \mathbf{Y}(t)$.
(2) Reset the labelled data, $\mathbf{Y}_L(t+1) = \mathbf{Y}_L(0)$, set $t = t+1$ and go to (1).

When the iterations converge, labels for the unlabelled examples are given simply by taking the class with maximal label, an illustration is shown in Fig. 1. If multiple classes are desired, one can threshold $\mathbf{Y}_U$. It is interesting to note, that Google's efficient PageRank algorithm [Page et al., 1998] works in the same way, except that label propagation is performed on a directed graph.

To analyse the algorithm we write the equilibrium solution as $\mathbf{Y}^* - \mathbf{P}\, \mathbf{Y}^* = \mathbf{0}$, where $\mathbf{Y}^*$ denotes the equilibrium solution and $\mathbf{0}$ is the vector of zeroes of length n. A subsequent step is to write the above algorithm as a constrained minimisation:

$$\begin{aligned} \mathbf{Y}_U^* &= \operatorname*{argmin}_{\mathbf{Y}_u} \mathbf{Y}^\top (\mathbf{I}_n - \mathbf{P})\, \mathbf{Y} \\ &= \operatorname*{argmin}_{\mathbf{Y}_u} \begin{bmatrix} \mathbf{Y}_L \\ \mathbf{Y}_u \end{bmatrix}^\top \begin{bmatrix} \mathbf{I}_L - \mathbf{P}_{LL} & -\mathbf{P}_{UL}^\top \\ -\mathbf{P}_{uL} & \mathbf{I}_u - \mathbf{P}_{uu} \end{bmatrix} \begin{bmatrix} \mathbf{Y}_L \\ \mathbf{Y}_u \end{bmatrix} \end{aligned} \tag{3}$$

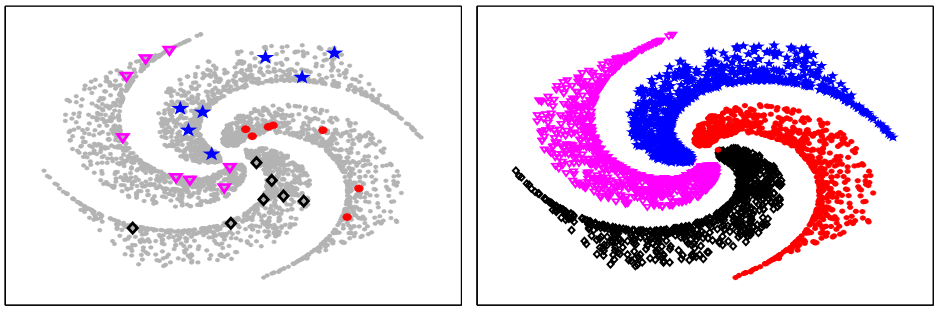

FIGURE 1. Illustration of label propagation: the left sub-figure shows the labelled data emphasised – there were 4 classes, and the right-hand side shows the resulting labels.

where the values $\mathbf{Y}_L$ do not change, with the exact solution [Zhu, 2005]:

$$\mathbf{Y}_U^* = (\mathbf{I}_U - \mathbf{P}_{UU})^{-1}\,\mathbf{P}_{UL}\,\mathbf{Y}_L \tag{4}$$

where we employed the matrix inversion lemma [Mardia et al., 1979]. As we see, to have the solution, we have to compute the inverse of a large matrix, which has cubic computation time $O(u^3)$ and can be costly for the large data-bases that could potentially be exploited.

3. An approximate solution

We propose a decomposition for the above problem. For this first we consider the optimisation problem from eq. (4) and observe that the labels can be decomposed into c components, $\mathbf{Y} = [\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(c)}]$ and we have to solve the equations independently. It is therefore enough to focus on two-class case, that is on a single vector $\mathbf{y}^{(i)} \stackrel{\text{def}}{=} \mathbf{y}$. Let $\{A_1, A_2, \ldots, A_d\}$ be a *partition* of the unlabelled set. Let us decompose the *large* $\mathbf{L} \stackrel{\text{def}}{=} \mathbf{I}_n - \mathbf{P}$ in blocks. We denote with $\mathbf{L}_{k\ell}$ the block assigned to the pair A_k and A_l, *i.e.* $\mathbf{L}_{k\ell} = \{L_{ij} | i \in A_k; j \in A_\ell\}$ leading to:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_L = \mathbf{y}_{A_0} \\ \cdots \\ \mathbf{y}_{A_d} \end{bmatrix} \qquad \mathbf{L} = \begin{bmatrix} \mathbf{L}_{00} & \mathbf{L}_{01} & \cdots & \mathbf{L}_{0d} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_{d0} & \mathbf{L}_{d1} & \cdots & \mathbf{L}_{dd} \end{bmatrix}$$

and the minimisation from eq. (3) is written as:

$$\mathbf{y}^\top (\mathbf{I}_n - \mathbf{P})\,\mathbf{y} = \sum_{a=0}^{d} \sum_{b=0}^{d} \mathbf{y}_a^\top \mathbf{L}_{ab} \mathbf{y}_b \tag{5}$$

where we stress that a and b start from 0 to include the labelled part of the data. We re-group the terms in eq. (5) to result in quadratic forms, each *within* a single partition A_k. Obviously, there is a *link* part that is responsible for the *global*

optimum, the resulting expression – equivalent to eq. (5) – is:

$$\mathbf{y}^\top (\mathbf{I}_n - \mathbf{P})\,\mathbf{y} = \sum_{a=1}^{d} \begin{bmatrix}\mathbf{y}_0\\ \mathbf{y}_a\end{bmatrix}^\top \begin{bmatrix}\mathbf{L}_{00} & \mathbf{L}_{0a}\\ \mathbf{L}_{a0} & \mathbf{L}_{aa}\end{bmatrix} \begin{bmatrix}\mathbf{y}_0\\ \mathbf{y}_a\end{bmatrix} - (d-1)\mathbf{y}_0^\top \mathbf{L}_{00}\mathbf{y}_0 + 2\sum_{a<b}^{d} \mathbf{y}_a^\top \mathbf{L}_{ab}\mathbf{y}_b \tag{6}$$

The minimisation of eq. (3) is now equivalent with the minimisation of d *independent small local quadratics*, the label propagation problem taken on the labelled set and A_ℓ each and the minimisation of the *link terms* $\mathbf{y}_a^\top \mathbf{L}_{ab}\mathbf{y}_b$, this latter making the optimisation problem global. We note that the final eq. (6) is still *exactly* the original label propagation problem. We aim for solving the small problems separately and then adjusting the original clusters according to the *fitness* of the clusters w.r.to the labelled set $\mathbf{y}_0$.

3.1. **Algorithm.** The proposed algorithm is stochastic minimisation that always finds the local optima within a single partition and updates the *partitions* such that the resulting subsets to be as uniform as possible. The algorithm is as follows:

- for $k = 1, \dots, d$ compute local optima $\mathbf{y}_k^*$;
- select pairs (k, ℓ) where we compute *pairwise* fitness of the solution, the last term in eq. (6): $-\mathbf{y}_k^\top \mathbf{L}_{k\ell}\mathbf{y}_\ell$;
- make adjustments if cluster solutions do not agree: swap data $\mathbf{x}_i$ and $\mathbf{x}_j$ that will lead to the largest increase in fitness, *i.e.* decrease in error.

References

Y. Bengio, O. Delalleau, and N. Le Roux. Label propagation and quadratic criterion. In O. Chapelle, B. Schölkopf, and A. Zien, editors, *Semi-Supervised Learning*, pages 193–216. MIT, 2006.

N. Cristianini and M. W. Hahn. *Introduction to Computational Genomics: A Case Studies Approach*. Cambridge University Press, 2006.

U. v. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis*. Academic Press, 1979.

L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1992.

U. von Luxburg, M. Belkin, and O. Bousquet. Consistency of spectral clustering. *Annals of Statistics*, 36(2):555–586, 2007.

X. Zhu. *Semi-supervised learning with graphs*. PhD thesis, School of Computer Science, Pittsburgh, PA, USA, 2005.

X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.

Department of Mathematics and Informatics, Babeş-Bolyai University Cluj–Napoca, Kogalniceanu 1, RO-400084

E-mail address: {lehel.csato, zalan.bodo}@cs.ubbcluj.ro