Classification Methods for Structured Outputs

Diego Sona, Paolo Avesani, Nicola Polettini
Fondazione Bruno Kessler (FBK-IRST)
38050, Trento, Italy
E-mail: {sona,avesani,polettini}@itc.it

Abstract - This paper is conceived as a summary of previous works that explored many alternative learning models for classification of documents on structured outputs. We provide a discussion of strong and weak points for each method.

I. Introduction

Learning general functional dependencies between arbitrary input and output spaces is one of the main goals in machine learning. In the last years, significant interest has grown around the opportunity to use flexible and powerful input and output representations. There are many different kinds of structure on the information. We may have instances encoded as a graph, links between instances, and relations among classes. Let consider the example of document organization on the Web. Documents are represented as structured XML instances that may include references to other documents in the Web space, and can be organized in a Web directory where nodes represent categories and the hierarchy encodes the relationships among categories. A learning model that deals with all these kinds of relational knowledge is still an open challenge. In this paper we are addressing those problems where we only have relationships among classes. We can talk in this case of structured output space.

As reference example in the following we will focus on the organization of Web documents. The typical case study is represented by the Web directories like Google, Yahoo!, LookSmart or the Dmoz open directory project. The usual process takes place in two stages: first the categories and the relations among them are defined, then the taxonomy is populated classifying Web documents under the proper nodes of the structure. We may distinguish two learning tasks: when the structure of categories is empty and when the structure of categories is already populated by examples. In the former case we fall in the semi-supervised problem while in the latter we address a supervised problem.

The semi-supervised problem is characterized by the lack of labeled documents while the enumeration of categories is known in advance. Let us refer to our example of Web documents organization. In the case of Web directories, classes are defined by lexical syntagms, the linguistic labels to denote the nodes in a taxonomy, and by the hierarchical relations between the categories. In this framework, the goal is to find a classification hypothesis just using the knowledge encoded in the taxonomy. In literature this scenario is referred to as "bootstrapping process" [10], [1]. The first purpose of this work is to investigate how learning models deal with the extraction of knowledge encoded in the structured output space. The main idea is to understand how to outperform the simplifying hypothesis that looks at the whole categories as a one versus all.

When a proper number of labeled documents is available we may reshape the problem as a supervised learning task. To achieve such a threshold when we are dealing with structured output space is a little tricky for two reasons. First, the number of labeled examples tends to growth exponentially accordingly with the evolution of the structure of the classes. Second, since the structure over the output space is introduced to reduce over-represented classes, it happens that when a category includes too many documents it is splitted. The side effect is that the categories never achieve a certain amount of labeled examples to trigger an effective training of a supervised classifier. The second goal of this work is to investigate how supervised learning models may be trained using few labeled examples while exploiting the structured output to collect additional related examples.

This paper is conceived as a summary of previous works that explored many alternative learning models such as regularized versions of k-means and EM, taxonomical self-organizing maps, EM with shrinkage, hierarchical Dirichlet and relational cascade-correlation. We provide a discussion of strong and weak points for each method. For a more detailed presentation of the empirical results we forward the reader to the referenced material.

II. Related Works

Classification problem with structured output are commonly approached by flattening the hierarchy of classes [7]. This solution, while neglecting the relational knowledge defined over the output space, allows to deploy

standard learning models reshaping of the classification problem.

Sebastiani [12], in his survey on machine learning methods for text classification, underlines that increasing attention is being given to hierarchical classification, exploiting the relationships between categories. In fact, many researchers have studied how a structured organization of classes, and in particular a hierarchical structure of categories, can improve the classification performance. Koller et al. [9] for example, using a Bayesian classifier at each node of the classification hierarchy, show that, in comparison to a flat approach, the hierarchical structure improves classification performance. Similar improvements are also reported in other works using many different approaches (e.g., Neural Networks [11], [2], SVMs [5], [4], Probabilistic models [3], [10]).

All these works share the common working hypothesis of looking at the hierarchy of categories as a discrimination tree. In a discrimination tree intermediate nodes subsume the sub-nodes, therefore documents are classified only on the leaves. Labeled examples of upper classes are obtained as the union of the documents classified under the lower classes. This way of proceeding allows to artificially increase the amount of labeled examples. But in the real world Web directories do not hold the implication that a document classified under a given class is necessarily classified even under the parent class.

In the following we survey both semi-supervised and supervised models extended to deal with structured output spaces. The discussion will refer empirical assessment performed on hierarchical document classification using datasets from Mesh, Google, Yahoo! and LookSmart taxonomies.

III. Learning Models

Here, we shortly present a set of learning approaches we designed, which are able to exploit the parent-child/neighborhood relationships between categories. All these approaches show that it is extremely fruitful to exploit the relationships between categories that are part of the available prior knowledge.

Starting with very few labeled data can cause severe learning problems to supervised algorithms. This is one reason why semi-supervised algorithms are better for this kind of tasks. A standard approach is to use a clustering algorithm constrained by some supervision. This supervision can be both in terms of a "proper" clusters initialization according to some prior knowledge, and in terms of labeled examples.

We did experiments on such approach adopting k-means and EM, two widely used algorithms based on a reference vector representation. The semi-supervision provided to these models was primarily based on a proper initialization of the reference vectors. In our experiments of text document classification, we did not observed any particular difference between the two approaches.

A. Regularized Clustering

Starting from the hypothesis that the relationships between classes can be used to improve model accuracy, we extended both k-means and EM introducing a regularization scheme. The underlying principle is that classes that are closer in the hierarchy tend to have similar documents. We enforce this principle during class means computation by a smoothing procedure that is carried out on the class means to obtain a "smoothed reference vector" [14]. A further explanation of the principle can be found in the Stein's paradox that says that introducing biases can sometimes improve the estimates [6]. Especially if the data used to make the estimates poorly represents the original distributions (too few data). This can be translated into the hypothesis that a weighted average between a class mean and the overall mean is a better and more robust estimate of that center, if the weights are chosen carefully. We can see the smoothing operation as a propagation scheme where information is propagated along the connections between classes.

The regularization for k-means is based on a iterative process that compute the class means using data and then smoothes the class means with the means of the nearest classes according to a neighborhood function. Similarly, in the EM algorithm, the class means are determined by using the posterior probabilities of classes given data, which were previously smoothed [14]. In this case, unlike the regularized K-means, we smooth the posterior class probabilities for each document separately instead of the class means.

In the regularized algorithms, we used various neighborhood functions for means smoothing. We observed, however, that regularization was always confirming our starting hypothesis that the exploitation of the relational information positively influences the models behavior. In particular, we observed that a good choice on the amount of propagation leads to a significant improvement in accuracy. We also observed that the Bayesian approach, independently of the regularization criteria, was always leading to better results than the regularized k-means.

With this approach, we obtain the advantage that any reference vector estimating a corresponding class distribution is more robust when taking advantage of the data in the nearest classes. However, in these models there is not any learning algorithm for finding the best smoothing parameters. Therefore, the main drawback is the

subjective choice of these parameters. We observed that, in general, regularizing the class means with the nearest classes gives better and more stable results, even with a bad choice of the regularizing parameter. This result says that it seems better to exploit the local knowledge to create robust estimators. In any case a bad choice of the regularization parameter can lead to a significantly bad model, even worse than the standard model without smoothing.

B. Taxonomic Self Organizing Maps

Trying to solve this regularization problem we devised a semi-supervised model drawing on the philosophy of Self-Organizing Maps [8] referred to as Taxonomic Self-Organizing Maps (TaxSOM) [1]. This neural model organizes text document data according to a given taxonomy using information coming both from node labels and their topological organization. The idea behind the model is very simple. A SOM can be seen as a collection of classes related each other according to a fixed topology (usually a lattice). These topological relationships have a strong impact on the model training and behavior. The idea is to exploit the learning algorithm of SOMs where the network topology is made isomorph to the organization of the classes in the taxonomy. This model has been also generalized for multi-classification tasks [13].

The advantage of adopting the SOMs learning algorithm is that we are introducing in the regularized semi-supervised classifier a standard and principle learning method for the regularization parameter. We observed that, in general, TaxSOM can obtain significantly better results than the regularized models previously described. However these results are unstable for different datasets. Actually, it is not very clear what is the right parameter assignments. In some circumstances it seems it is a good thing to start with a strong regularization (high propagation of information along the links between classes), in other cases it was better to start with nearly no regularization.

C. EM with Shrinkage

Drawing from the principle of Stein's paradox [6], there is a work [10] which aim is the determination of the smoothing parameters for semi-supervised and supervised classification in the leaves of the taxonomies. They propose a probabilistic generative model that estimates the Naive Bayes parameters improving the robustness of the estimates using the contextual knowledge on the hierarchies. This knowledge is exploited by a process referred to as shrinkage that learns the smoothing parameters from held-out data. In this way we are able overcome the problem of parameters' tuning.

Since in our task all nodes in a hierarchy can contain patterns, we extended the above model allowing to classify pattern in all nodes of a generic graph [**?**]. We provided experimental results as evidence that using shrinkage significantly improves the accuracy of models that do not use the relational knowledge.

A frequent condition that happens in this kind of task is that very few data are available with respect to the number of categories. Ideally, the proposed clustering approach was designed for a robust estimation of parameters when datasets are small. We observed, however, that for small datasets the algorithm tends to fails. The key point is that the estimation of shrinkage parameters is subject to the changes in the distributions during learning, which can be big if few examples are available. To make the model more robust we determined some rules to reduce the number of free parameters. The problem is that this reduction is still subjective and strongly dependent on the underlying dataset.

D. Hierarchical Dirichlet

We also proposed an alternative probabilistic solution [15], a generative Hierarchical Dirichlet model that derives a classification method for text documents into a given concept hierarchy. The basic idea is that each class is still described by a multinomial distribution. The novelty lies in the specification of a model of propagation with dependent Dirichlet priors where the dependence is influenced by the structure of the concept taxonomy.

Under this model we derived formulae to estimate the parameters in a supervised as well as an unsupervised setting. We have seen that under this model the parameter estimates closely resemble the shrinkage estimates used in statistics. Moreover we derived a learning method for the smoothing parameter based on the maximization of maximum likelihood on held-out data.

The advantage of this model is that it is formally derived from starting hypotheses, and there are not learning parameters that can influence the quality of classification. The main drawback of this algorithm however are its computational complexity and the restriction to hierarchies only.

E. Relational Cascade Correlation

Finally, we proposed a neural network approach to the supervised classification able to deal with classes structured as a graph (e.g., taxonomies, ontologies). The model learns from data both the class distribution and

the required amount of smoothing. The model is basically a non-stationary recursive cascade correlation. We refer to it as Relational Cascade Correlation. The idea is that for each class there is a standard cascade correlation that takes input from data and from states of nearby models. Growing the entire model layer by layer, as in standard cascade-correlation, the model increases the capacity to watch at larger neighborhood.

The main advantage is that the weighted connections transmit signals in only one direction, eliminating the problems of dynamic systems convergence in presence of cycles. Moreover, it is intrinsically non-linear and multivariate. Finally it simplify the case where multi-label multi-class classification task is required. On the other side this model can be trained only with supervised algorithms. Moreover it can be highly sensitive to the dimensionality of dataset leading to overfitting problems due to the growing number of parameters.

IV. Conclusions

We have definitively shown that there is the opportunity to boost a learning model taking advantage of the knowledge encoded in a structured output space. Relations among classes can be exploited to balance a poor amount of labeled examples. Nevertheless a more fine grained learning model to capture the weighted contribution of a single relation increases the number of parameters that have to be estimated. This way of proceed is in contrast with the starting hypothesis that only few labeled examples are available.

Although we didn't mention in advance the investigation of learning model for structured output is difficult because the setup of an empirical evaluation is not straightforward. Since the datasets provide few examples for each class the traditional cross fold validation is not working properly. Moreover the performance measures need to be revised to meet the specific nature of structured output. Up to now there is not a general agreement on how to eval a learning model for structured output.

References

[1] G. Adami, P. Avesani, and D. Sona. Clustering documents into a web directory for bootstrapping a supervised classification. *Journal of Data and Knowledge Engineering*, 54(1):301–325, 2005.

[2] R. A. Calvo and H. A. Ceccatto. Intelligent document classification. *Journal of Intelligent Data Analysis*, 4(5):411–420, 2000.

[3] M. Ceci and D. Malerba. Web-pages classification into a hierarchy of categories. In *Proc. of ECIR-03, 25th European Conf. on Information Retrieval*, pages 57–72, 2003.

[4] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni. Hierarchical classification: combining bayes with svm. In *Proc. of 23rd International Conference on Machine Learning ICML*, pages 177–184, 2006.

[5] S. Dumais and H. Chen. Hierarchical classification of web document. In *Proc. of SIGIR-00, 23rd ACM Int. Conf. on Research and Development in Information Retrieval*, pages 256–263, 2000.

[6] B. Efron and C. Morris. Stein's paradox in statistics. *Scientific American*, 236:119–127, May 1977.

[7] M. Grobelnik and D. Mladenic. Simple classification into large topic ontology of web documents. *Journal of Computing and Information Technology*, 13(4):279–285, 2005.

[8] T. Kohonen. *Self-Organizing Maps*, volume 30 of Series in Information Sciences. Springer, Berlin, 2001.

[9] D. Koller and M. Sahami. Hierarchically classifying documents using very few words. In *Proc. of ICML-97, 14th Int. Conf. on Machine Learning*, pages 170–178, 1997.

[10] A. McCallum and K. Nigam. Text classification by bootstrapping with keywords, EM and shrinkage. In *ACL-99 Workshop for Unsupervised Learning in Natural Language Processing*, pages 52–58, 1999.

[11] M.E. Ruiz and P. Srinivasan. Hierarchical text categorization using neural networks. *Journal of Information Retrieval*, 5(1):87–118, 2002.

[12] F. Sebastiani. Machine learning in automated text categorization. *Journal of ACM Computing Surveys*, 34(1):1–47, 2002.

[13] D. Sona, P. Avesani, and R. Moskovitch. Multi-classification of clinical guidelines in concept hierarchies. In *Proc. of Workshop on Intelligent Data Analysis in Medicine and Pharmacology IDAMAP*, pages 256–263, 2005.

[14] D. Sona, S. Veeramachaneni, P. Avesani, and N. Polettini. Clustering with propagation for hierarchical document classification. In *ECML-04 Workshop on Statistical Approaches to Web Mining*, pages 50–61, 2004.

[15] S. Veeramachaneni, D. Sona, and P. Avesani. Hierarchical dirichlet model for document classification. In *Proc. of Int. Conf. on Machine Learning ICML*, pages 928–935, 2005.