A Meta-Top-down Method for Large-scale Hierarchical Classification *

XIAO-LIN WANG [1,2], HAI ZHAO[1,2], BAO-LIANG LU[1,2]

[1]Center for Brain-Like Computing and Machine Intelligence
[2]MOE-Microsoft Key Laboratory for Intelligent Computing and Intelligent Systems
Shanghai Jiao Tong University, 800 Dong Chuan Rd., Shanghai 200240, China
E-MAIL: arthur.xl.wang@gmail.com, {zhaohai; blu}@cs.sjtu.edu.cn

Abstract

This paper presents the system with which we participate at the third Pascal challenge on Large-scale Hierarchical Text Classification. This system implements the meta-top-down method proposed in (Wang et al., 2011). This method employs meta-classification to enhance the classifying procedure of normal top-down methods to relieve their so-called error propagation problem. In this paper we first briefly describe this method, and then present the settings and results of our implemented system.

1 Method

The idea of Meta-Top-Down method (MetaTD) is to employ meta-classification to re-classify samples according to their results yielded by normal top-down methods. It takes the confidence scores of base-classifiers along a root-to-leaf path as a meta-level input, and takes whether the leaf node is a correct label as a meta-level output. This framework is formulated as follows,

$$\mathcal{M}(u,l) = (\mathcal{M}_x(u_x,l), \mathcal{M}_y(u_y,l))$$
$$\mathcal{M}_x(u_x,l) = \{(n_i, f_{n_i}(u_x)) | n_i \in p_l\}$$
$$\mathcal{M}_y(u_y,l) = \begin{cases} +1, l \in u_y \\ -1, l \notin u_y \end{cases}$$

where $\mathcal{M}_x$ and $\mathcal{M}_y$ are the input and output of a meta-sample respectively; u_x and u_y are the input and output of a base-sample respectively; l is a leaf node which is a validate class label for base-samples at the same time; $p_l = (n_{i_1}, n_{i_2}, \ldots, n_{i_k})$ is a path from the root to l where n_{i_1} is the root, $n_{i_k} = l$, and $(n_{i_a}, n_{i_{a+1}})$ is a parent-child relation; $f_{n_{i_a}}$ is the base-classifier of the node n_{i_a}.

The workflow of meta-top-down is presented in Fig. 1.

The training phase consists of three steps as follows,

1. Train base-classifiers f_c on the training data set T, which is the same as the normal top-down methods.

2. Construct a meta-training set with the base-classifiers and the development set D,

$$M_T = \cup_{u \in D} \{\mathcal{M}(u,l) | l \in L(u_x)\}.$$

3. Train a meta-classifier g on M_T.

The classifying phase also consists of three steps as follows,

1. Construct a group of meta-samples from a test base-sample u_x (its label u_y is unknown),

$$M_E = \{\mathcal{M}_x(u_x,l) | l \in L(u_x)\}.$$

2. Present these meta-samples to the meta-classifier g,

$$g(M_E) = \{g(\mathcal{M}_x(u_x,l)) | l \in L(u_x)\}$$
$$= \{g_{u_x,l} | l \in L(u_x)\}.$$

* This work was partially supported by the National Natural Science Foundation of China (Grant No. 60903119 and Grant No. 61170114), the National Research Foundation for the Doctoral Program of Higher Education of China under Grant No. 20110073120022, the National Basic Research Program of China (Grant No. 2009CB320901) and the European Union Seventh Framework Programme (Grant No. 247619).

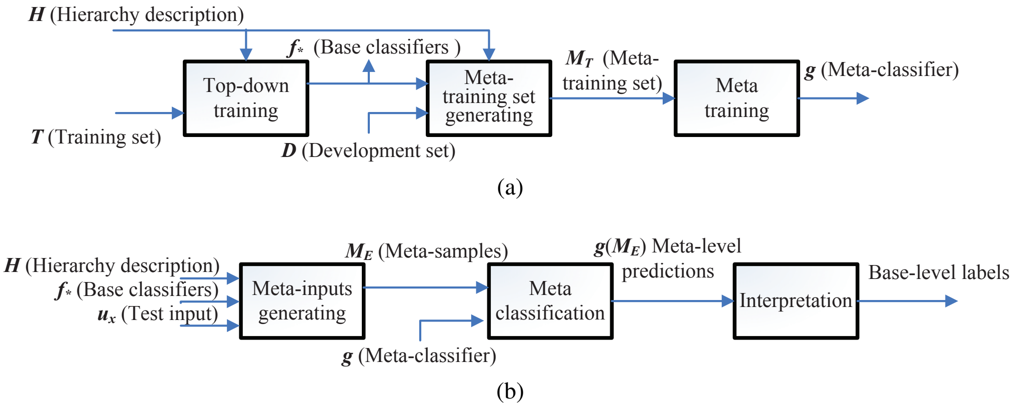

Figure 1: Workflows of MetaTD: (a) training phase; (b) classifying phase

meta-classifier	Acc	LBMiF	LBMiP	LBMiR	Training Time(s)
Liblinear	0.426	0.487	0.507	0.470	424
Max Entropy	0.422	0.482	0.512	0.455	517
SVM^{light}	0.438	0.493	0.566	0.438	1.8×10^3

Table 1: Accuracy and Time Cost on Track 1 Medium Task

3. Interpret the predictions into base-level labels.

In addition to the above description, a few technical details are required to make meta-top-down effective and efficient (please see (Wang et al., 2011)).

2 Experiment

2.1 Settings

The employed sample representation is TFIDF, and it is generated from the indexed data of LSHTC3.

The system employs SVM^{light} as the base classifier of top-down method. The default parameters of SVM^{light} are employed. Our pilot experiments show that tuning the parameters results in a minor rise of accuracy, but the procedure is quite time-consuming.

The system tries Liblinear, the Stanford Max Entropy classifier and SVM^{light} as the meta-classifier. The default parameters are employed, as our pilot experiments show that they are the optimal settings for our meta-classification task.

Meta-level predictions are interpreted to base-level predictions by a score-cut thresholding strategy. The base-level labels whose corresponding meta-score exceed a threshold are chosen as the base-level predictions. The threshold is tuned on a held-out development set.

2.2 Results

The implemented system is applied to the task of Track 1 Medium. The results show that our system is both accurate and efficient (see Tab. 1). On the aspect of accuracy, three meta-classifier achieves high accuracy where SVM^{light} is the most accurate. These results are competent among the participants of LSHTC3.

On the aspect of effeciency, however, Liblinear and Max Entropy consume little training time while SVM^{light} is painfully slow. The underlying reason is that the trainer of SVM^{light} works on the dual problem, but the number of samples is well above the number of dimensions in this application. The overall running time of the implemented system is close to that of a normal top-down method, since the meta-classification part consumes only about 10% additional time.

References

X.-L. Wang, H. Zhao, and B.-L. Lu. Enhance top-down method with meta-classification for very large-scale hierarchical classification. In *Proceedings of IJCNLP*, pages 1089–1097, Chiang Mai, Thailand, November 2011. Asian Federation of Natural Language Processing.