Hierarchical Text Classification for News Articles Based-on Named Entities

Yaocheng Gui[1,2], Zhiqiang Gao[1,2], Renyong Li[3], and Xin Yang[3]

[1] Institute of Web Science, School of Computer Science and Engineering, Southeast University, Nanjing 211189, P.R. China
[2] Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, Nanjing 211189, P.R. China
{yaochgui,zqgao}@seu.edu.cn
http://iws.seu.edu.cn
[3] Focus Technology Co., Ltd., Nanjing 210061, P.R. China
{lirenyong,yangxin}@made-in-china.com

Abstract. There exist a range of hierarchical text classification approaches that classify text documents into a pre-constructed hierarchy of categories. In these approaches, feature selections are often based on terms (words or phrases), which are unsuitable for hierarchically classifying news articles. Named entities are informative features in news articles which have not been studied seriously in previous hierarchical text classification approaches. This paper utilizes named entities as features for classifying news articles into a pre-constructed hierarchy about international relations. The feature selection is implemented based on named entities associated with local categories. Documents are then represented by the selected features using two types of models, which are Boolean model and Vector model. We train SVMs corresponding to both types of models based-on local information. The experimental results show that the use of named entities improves the performance of hierarchical text classification for news articles.

Keywords: Hierarchical Text Classification, Feature Selection, Named Entity, Support Vector Machine.

1 Introduction

There are millions of news articles about international relations available on the Web. Applications with the capability of analyzing or understanding news articles about international relations are developed for the purpose of helping people to manage the information. Hierarchical classification is at the heart of such applications, which categorized documents into a pre-constructed hierarchy – typically a tree or a direct acyclic graph (DAG). This paper aims at classifying Chinese news articles about international relations into a hierarchy with hundreds of categories, named international relation taxonomy (IRT). The IRT

S. Zhou, S. Zhang, and G. Karypis (Eds.): ADMA 2012, LNAI 7713, pp. 318–329, 2012.

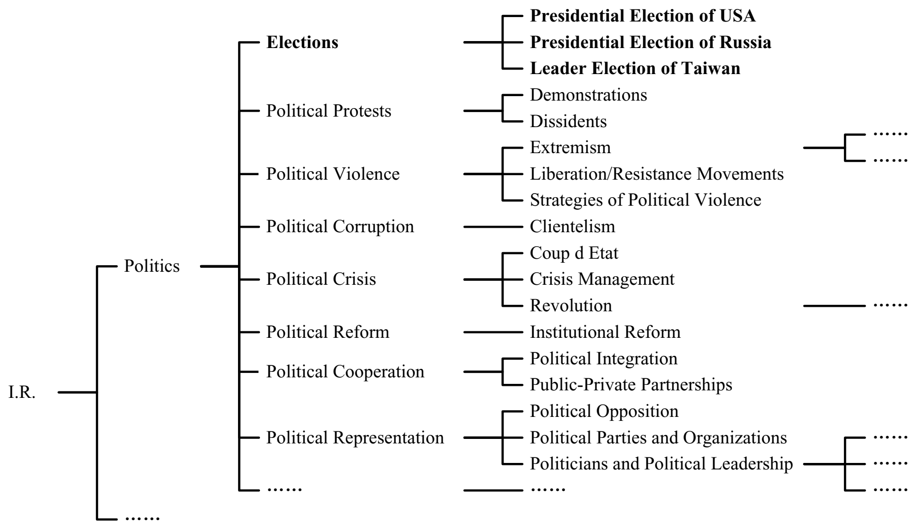

Fig. 1. Part of the International Relation Taxonomy

is constructed with a few specified categories which do not exist in other general taxonomies. Figure 1 illustrates a part of the IRT hierarchy.

The IRT is a hierarchy provided by experts of international relation domain, containing 321 categories distributed in 6 levels. A set of 5,758 Chinese news articles, which are crawled from news portals such as Sina[1] and Sohu[2], are classified into these categories manually. All the ambiguous documents are voted by three experts. Thus, 260 of these categories are related to at least one document and each category consists of 22 documents in average. Figure 2 illustrates a paragraph of a news article from the category *Presidential Election of USA* that consists of a few tagged named entities.


中新网（China News）2月29日电 据外电报道，美国（USA）密歇根州（Michigan）与亚利桑那州（Arizona）当地时间28日同时举行共和党（Republican Party）总统初选，共和党（Republican Party）总统候选人罗姆尼（Romney）已经稳赢亚利桑那州（Arizona）初选，得票率遥遥领先主要对手桑托勒姆（Santorum）；罗姆尼（Romney）还同时有望赢得密歇根州（Michigan）的初选，7成计票结果显示，罗姆尼（Romney）得票率领先桑托勒姆（Santorum）4个百分点。外电普遍称，罗姆尼（Romney）连赢两州初选，再次夺得领先优势。

Fig. 2. A paragraph of a news article from the category *Presidential Election of USA* that consists of a few tagged named entities. Three types of named entities are tagged, which are person names in blue, organization names in red, and location names in yellow. All these named entities are interpreted into English.

[1] http://www.sina.com.cn/
[2] http://www.sohu.com/

Previous approaches to solve the hierarchical classification problem can be divided in two types: global (or big-bang) approaches and local (or top-down) approaches according to [1]. The features in both types of approaches are often based-on terms, which are words and phrases. There exist two major difficulties when implementing hierarchical text classification approaches based-on terms on the IRT hierarchy:

- **Close Categories.** The first difficulty is the high similarity of categories with the same parent node in the IRT hierarchy. For example, the documents from categories *Presidential Election of USA* and *Presidential Election of Russia* talk about the same topic, i.e. presidential election, in different countries. Such categories are noted as close categories. The documents from close categories are difficult to be distinguished with features selected from terms [2].
- **Rare Categories.** The second difficulty is the sparsity of document distribution on leaf node categories. If we only consider documents assigned by human editors directly to those 260 categories (without counting in the documents assigned to their child categories), most categories have very few labeled documents. For instance, over 50% of the categories in the IRT hierarchy have fewer than 10 labeled documents. Such types of categories also exist in the Yahoo! categories, which are denoted as rare categories [1]. For hierarchical text classification with rare categories, there are not enough training examples for most learning algorithms to train an effective classifier in a high dimensionality feature space.

Named entity recognition (NER) is a subtask of information extraction that seeks to locate and classify atomic elements in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages [3]. For example, three types of named entities are illustrated in Figure 2. Through a pre-process of named entity recognition on the set of 5,758 Chinese news articles, we found that 99.4% of these articles contain named entities. Generally, each article contains a few number of named entities, 14.7 per article in average.

Intuitively, a few informative named entities, such as the candidate's name or the location name, are often sufficient to distinguish news articles about presidential election of USA from those of Russia. For example, supposing that we are trying to classify a news article into the two forementioned categories, if we have known that it is about presidential election, then the appearance of a special named entity, e.g. *Putin* or *Mitt Romney*, may directly impact the decision of classification. Given such types of named entities, other sequences of words add little differentiation power, and are therefore redundant. This type of categorization problem is characterized with redundant features [2]. It is claimed that a particular concept can be learned with a small number of features, while the rest of the features do more harm than good in close categories [2].

According to this intuition, we assume that named entities are distinguishable features for news articles about close categories. Because the named entities distribute in different categories of the IRT hierarchy, we implement feature selection method based on local collections of named entities with respect to each non-leaf categories. Furthermore, we represent text document using Boolean model as well as Vector model.

In rare categories, the dimension of the feature space consists of terms can be high for the number of training examples. The use of named entity as feature may help to reduce the dimension of the feature space. However, it can be further reduced by some aggressive feature selection strategies, e.g. select only 500 features for each category. We assume that the hierarchical text classification about rare categories will benefit from the reduced feature space brought by feature selection methods. Because it decreases the requirements of training examples .

SVM is suggested to be very robust in the presence of numerous features, according to a wide range of studies about hierarchial text classification which implement SVM as the basic classification model [1, 2, 4, 5]. We trained SVM for classification and regression according to the local training examples with respect to the category of the IRT hierarchy. Some previous work showed that the accuracy of SVM in hierarchical text classification can also be improved by local feature selection [1].

This paper aims at providing a hierarchical text classification approach for news articles by using named entities as features, implementing local feature selection with an aggressive strategies, training SVM with local information, and predicating classes with top-down approach.

The rest of the paper is organized as follows. In Section 2, we give a brief overview of related works. In Section 3, we describe the models and strategies of proposed approaches. The experimental results are shown in Section 4. Section 5 concludes with a summary and plans for future work.

2 Related Works

In hierarchical text classification, many algorithms have been proposed. According to how the hierarchical structure is explored [6], the current literature often refers to local approaches, when the system construct classifiers on each level of the hierarchy where each classifier works as a flat classifier on that level; and global approaches, when a single classifier coping with the entire class hierarchy is used. It is infeasible to directly build a classifier for a large-scale hierarchy [1]. We review the local classifier approaches and the related work on using named entity in classification in this section.

The first study of the hierarchical text categorization problem is carried out in [7]. It proposed a local approach with divide-and-conquer principle, the system first classified a document into high-level categories and then proceeded iteratively dealing only with the children of the categories selected on the previous level. This study experimentally showed that hierarchical information can be extremely benefit for text categorization by improving the classification performance over the “flat” technique, which does not use the hierarchical information.

The hierarchical SVM is first used in [4, 8] to deal with the hierarchical text classification problem. In order to assign a test example to a class, the probabilities for the predicted class were used. The first method uses a boolean condition where the posterior probability of the classes on the first and second levels must be higher than a user specified threshold. The second method uses a multiplicative threshold that takes into account the product of the posterior probability of the classes on the first and second levels.

The evaluation of SVM over the full taxonomy of the Yahoo! categories, is reported in [1]. The scalability and effectiveness of SVM and two threshold tuning methods were theoretically analyzed. A distributed classification system was developed for experiments. In implementation of SVM, the sequential minimal optimization (SMO) algorithm with a linear kernel was used for the binary version of the SVM classifier and 4,000 features were selected for each binary classification task. It is claimed that the hierarchical use of SVM is efficient enough for very large-scale classification and the effectiveness is still far from satisfactory.

Recent research has focused on two issues: increasing sparsity of training data at deeper nodes in the taxonomy, and error propagation where a mistake made high in the hierarchy cannot be recovered. The approach for classifying large scale text hierarchy at deep-nodes introduced in [5] consists of a search stage and a classification stage. In the search stage, a subset of categories from the large scale hierarchy is formed related to a given document by a category-search algorithm. In the classification stage, a specific classifier is trained for each given document. This approach prunes the large-scale hierarchy to a small subset of flat categories, which decomposes the difficult global problem into simpler local ones. Other studies suggested techniques for the error propagation issue, such as the isotonic smoothing approach provided by [9] and the Refined Experts model provided by [10].

There are a range of hierarchical text classification approaches that using terms (words or phrases) as features. A feature selection method that both consider general terms and named entities is proposed in [11]. The linear-chain CRFs is used to train named entity recognition models based on named entities extracted from the corpus. Then, a popular feature selection method – information gain (IG) – is implemented to select Chinese phrases as terms. The experimental results showed that the general feature selection method combining with named entity information performed better than traditional feature selection methods. However, it is implemented for flat text classification instead of hierarchical text classification.

The hierarchical text classification approach proposed in this paper uses named entity as feature and information gain as feature selection method according to [11]. However we consider only named entities as features instead of term features and implement information gain based on local categories of the IRT hierarchy. In addition, we compare two document representation models which are Boolean model and Vector model using named entities. In the implementation of SVM, we utilize the SMO algorithm with a linear kernel for the binary version of SVM according to [1]. Besides, we also compare the SVM for regression with the binary version of SVM. The dimension of the feature space in our approach is lower than it is reported in [1]. Finally, we implement the top-down approach for class prediction according to [7].

3 Hierarchical Text Classification Approach

In this section, we first introduce the formulation of hierarchical classification for documents. Then, we present the representation models used in this paper. Finally, we propose the hierarchical classification approach for news articles based-on named entity.

3.1 Problem Formulation

We introduce the definition of hierarchical text classification presented by [6, 12, 13].

Definition 1. ***Hierarchical Text Classification*** *is the task of assigning a boolean value to each pair* $\langle d_j, c_i \rangle \in D \times C$ *with a given structure* $\mathcal{H} = \langle C, \prec \rangle$, *where* D *is a domain of documents,* $C = \{c_1, \ldots, c_{|C|}\}$ *is a set of predefined categories and* $\prec \subseteq C \times C$ *is an asymmetric, anti-reflexive, transitive binary relation on* C.

The binary relation $\prec$ represents the “IS-A” relationship. “IS-A” relationship is defined as asymmetric, anti-reflexive and transitive in [6]:

- $\forall c_i, c_j \in C,$ if $c_i \prec c_j$ then $c_j \not\prec c_i$
- $\forall c_i \in C, c_i \not\prec c_i$
- $\forall c_i, c_j, c_k \in C, c_i \prec c_j$ and $c_j \prec c_k$ imply $c_i \prec c_k$

For any hierarchy $\mathcal{H}$, it is assumed that the existence of the root (or top) category $Root(\mathcal{H})$ which is an ancestor of all other categories in the hierarchy.The IRT hierarchy in our application is exactly a tree.

3.2 Representation Models

Vector space model is an algebraic model for representing text documents as vectors of identifiers, such as terms. For example, document d_j is represented by vector $\langle w_{1j}, w_{2j}, \ldots, w_{ij}, \ldots, w_{mj} \rangle$, where w_{ij} is a value associate with the i-th feature in document d_j. We present two types of models to represent documents, which are Boolean model where w_{ij} is a binary value which denotes the appearance of the i-th feature in document d_j, and Vector model where w_{ij} is a real value which denotes the importance of the i-th feature in document d_j. Particularly, we implement the *term frequency/inverse document frequency* (TF-IDF) score as the value of w_{ij} in Vector model.

Corresponding to the two types of models, we build SVM for classification with respect to the Boolean model and SVM for regression with respect to the Vector model at each node of the hierarchy. The performance of these two types of SVM models are compared in Section 4.

3.3 Hierarchical Text Classification Based-on Named Entity

We implement the "local classifier per parent" [6] strategy in this paper. For each parent category (non-leaf category) in the hierarchy, a multi-class classifier is trained to classify documents into its child categories. To distinguish Feature Selection, Training and Testing from the content, we propose our hierarchical text categorization approach in three phases. Details of each phase is described as follows.

Feature Selection. In this phase, terms and named entities are considered as different types of features for the task of hierarchical text classification. We extract terms by constraining the length less than 6 words and the frequency more than 5 in the corpus. As a result, each news article contains 310 terms in average. On the other hand, we employ ICTCLAS[3] to recognize three typical types of named entities, which are the names of persons, organizations and locations, from each news article. Totally, each news article contains 14.7 entities in average. However, there exist a few named entities, such as the location name, e.g. China, and the organization name, e.g. Sina, that are common in news articles. These named entities are less informative and provide little power of differentiation.

Information gain is frequency employed as a termgoodness criterion in the field of machine learning [14]. We implement information gain as feature selection method based on named entities as well as terms in our approach to select a set of features for each category. The information gain of a feature t (named entity or term) with respect to category c_k is defined as follows [12]:

$$P(t)\sum_{k=1}^{|C|} P(c_k|t)\cdot\log_2\frac{P(t,c_k)}{P(t)\cdot P(c_k)} + P(\bar{t})\sum_{k=1}^{|C|} P(c_k|\bar{t})\cdot\log_2\frac{P(\bar{t},c_k)}{P(\bar{t})\cdot P(c_k)} \quad (1)$$

where $|C|$ denotes number of categories in the category set C, $c_k \in C$, $P(t)$ denotes the probability that a random document contains feature t, $\bar{t}$ represents the absence of feature t, $P(\bar{t})$ denotes the probability that a random document does not contains feature t. We compare the performance of hierarchical text classification approaches with different size of feature sets.

Comparatively, we propose a naive method, which calculating TF-IDF score for each feature and selecting the most informative features from each document. The TF-IDF score of the i-th feature in the j-th document is defined as follows [15]:

$$w_{ij}(c) = tf_{ij}\cdot\log_2\left(\frac{|Doc(c)|}{df_i(c)}\right) \quad (2)$$

where $Doc(c)$ denotes a subset of articles that labeled by category c and $|Doc(c)|$ is the size of the subset, tf_{ij} denotes the frequency that the i-th feature is mentioned in the j-th document, and $df_i(c)$ is the number of documents that mentioned the i-th feature in set $Doc(c)$. The feature set selected by naive method

[3] http://ictclas.org/

consists of one or two most informative features, with the highest TF-IDF scores, from each document.

Training. In this phase, we train multi-class classifier for each non-leaf category of the hierarchy. The multi-class classifier decomposes into a set of binary classifiers or predictors corresponding to its descendant categories. We first represent training examples by feature vectors in the local feature space associated with the current category. Then we define the set of positive and negative examples for each descendant category of the current category. Finally, we train SVM for classification and regression for these descendant categories.

There are different ways to define the set of positive and negative examples for training the binary classifiers. It is suggested that the siblings policy is in advance of using considerable less data and obtain no bad accuracy [6]. The siblings policy is defined as follows:

$$Tr^{+}(c) = *(c) \cup \Downarrow (c) \tag{3}$$

$$Tr^{-}(c) = \leftrightarrow (c) \cup \Downarrow (\leftrightarrow (c)) \tag{4}$$

where $Tr^{+}(c)$ denotes the set of positive training examples of category c, $Tr^{-}(c)$ denotes the set of negative training examples of category c, $*(c)$ denotes the set of examples whose categories is c, $\Downarrow (c)$ denotes the set of descendant categories of c, and $\leftrightarrow (c)$ denotes the set of sibling categories of c. It means that the positive examples of category c are the documents belong to category c and its descendant categories, and the negative examples are the documents belong the siblings of category c and their descendant categories.

We build SVM for classification and regression in this training phase. The SVM for classification is trained by the SMO algorithm with polynomial kernel [16], and the SVM for regression is trained by the RegSMOImproved algorithm with polynomial kernel [17]. Both of the two algorithms are provided by Weka[4] using the default parameter setting.

Testing. In this phase, a top-down classification policy is employed to classify the unlabeled news articles. For each multi-class classifier, we make decision about which descendant category are classified or predicted. This process is repeating until we reach a leaf category or can not classify the test example to any descendant. There are two tasks corresponding to the SVM for classification and SVM for regression.

In classification task, the multi-class classifier returns the positive results of its binary classifiers. For example, if the binary classifier at *Presidential Election of USA* returns true for a news article, then the multi-class classifier at *Election* will return *Presidential Election of USA* as its result. In regression task, the multi-class classifier considers the average prediction value of its predictors. For example, if the predicator at *Presidential Election of USA* returns a value higher than the average of its siblings, then the multi-class classifier at *Election* will return *Presidential Election of USA* as its result.

[4] `http://www.cs.waikato.ac.nz/ml/weka/`

4 Experimental Results

In this section, we present the experimental results of named entity based hierarchical text classification approach. To exam the utility of named entity based representation models in the first assumption, we analyze the performance of named entity features compared with term features in both close categories and rare categories. Two types of feature selection methods, the naive method and information gain method, are compared with different number of features. Besides, the comparisons of SVM for classification and regression are provided to exam the Boolean model and Vector model.

The labeled documents that related to the IRT hierarchy are divided into two sets: the training set and the testing set. For each category, the labeled documents are randomly divided into 5 folds: 4 folds for training and 1 fold for testing. In addition, we remove the categories containing only one labeled document. This partition guarantees that the testing documents are with the same distribution of the data. Further, we select a set of categories that consists less than 10 labeled documents as the rare categories, and manually choose a set of categories that contain similar topics as the close categories. Thus, we have close categories testing set, rare categories testing set and hybrid category testing set which contains both close categories and rare categories. The micro-averaged F1 scores [18] as well as the micro-averaged precision and recall are utilized as metrics.

Comparison of Features. The effectiveness of named entity feature and term feature are compared with different feature numbers on close categories testing set. In this experiment, we implement information gain method, both Boolean model and Vector model of feature vectors are tested. All results are presented in Figure 3. It is illustrated that named entity features performs better than term features when a few number, e.g. 500, of features are selected. Because the named entities are informative features for close categories, and the use of named entity deduces the dimension of the feature space.

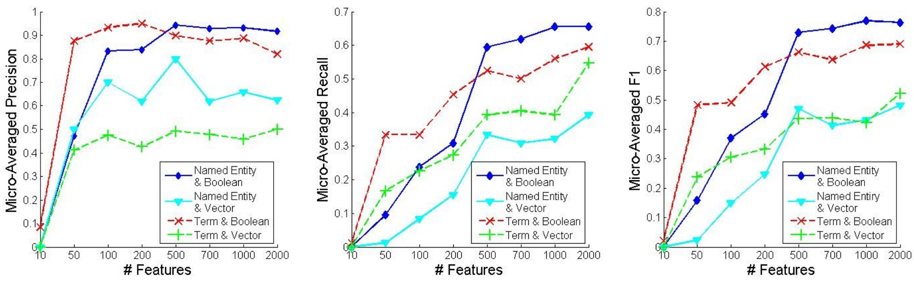

Fig. 3. Micro-averaged precision, recall and F1 score (y-axis) for hierarchical text classification using named entity and term features on close categories testing set. The x-axis denotes the number of features range from 10 to 2000.

Previous research which studied document representation using named entities in new event detection, found that the named entities does not always effective[19]. Our experiments show that the effectiveness of such document representation is also associated with the number of named entities.

Comparison of Feature Selection Methods. We compare two feature selection methods, the naive method and the information gain method. In this experiment, we implement the SVM for classification on three testing sets. The naive method selects one most informative feature from each document, and the information gain method selects 2,000 features from the category. It is illustrated in Table 1 that the information gain method performs better for both named entity features and term features.

Table 1. Micro-averaged F1 score for hierarchical text classification using information gain method and naive method

Feature Selection Methods		Testing Set		
		Close Categories	Rare Categories	Hybrid Categories
Naive Method for	Named Entity	0.705	0.309	0.407
	Term	0.549	0.323	0.352
IG Method for	Named Entity	0.763	0.266	0.405
	Term	0.689	0.292	0.393

Comparison of SVM. We compare the SVM for classification and regression in this experiment. The results are illustrated in Figure 4. It is illustrated that the SVM for regression associated with the Vector model performs better than the SVM for classification associated with Boolean model on close categories testing set, while the Boolean model performs better on rare categories and hybrid categories testing sets. Because the SVM for regression requires more training examples than the SVM for classification. Hence, the performance of SVM for regression decreases on rare categories due to the lack of training examples.

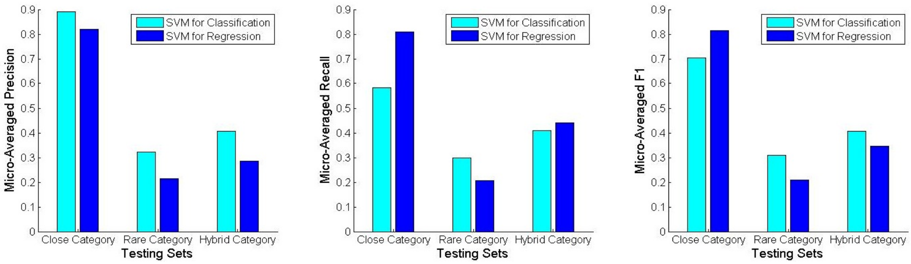

Fig. 4. Micro-averaged precision, recall and F1 score (y-axis) for hierarchical text classification with SVM for classification and regression, using named entity features on testing sets. The x-axis denotes different testing sets.

The performance of Vector model is relevant with the effectiveness of SVM for regression, it also causes that the Boolean model performs better than Vector model as illustrated in Figure 3.

Overall Performance. In this experiment, we test SVM for classification using named entity features on three testing sets, which are close categories testing set, rare categories testing set and hybrid categories testing set. The experiment results are illustrated in Figure 5. It is shown that the bad performance of hierarchical text classification on rare categories decreased the overall performance of the hierarchy. Such types of categories are suffered from the skewed distribution of data, since the use of named entity cannot improve the performance of hierarchical text classification on the rare categories.

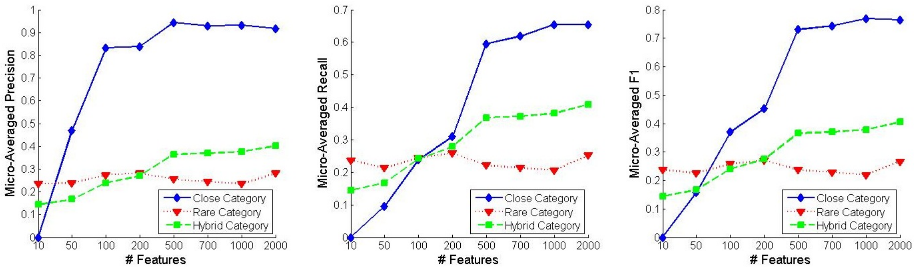

Fig. 5. Micro-averaged precision, recall and F1 score (y-axis) for hierarchical text classification using named entity features on close categories, rare categories and hybrid categories testing set. The x-axis denotes the number of features range from 10 to 2000.

5 Conclusions and Future Works

We introduce a hierarchical text classification approach for news articles about international relations, focusing on the difficulties of close categories and rare categories. We use named entities as features, implement local feature selection with an aggressive strategies, train SVM with local information, and predicate class with top-down approach. Although this approach is not effective for rare categories due to the lack of training examples, the micro-averaged F1 score for close categories is improved to 81.4%. The experiment results show that the Boolean model is more effective than Vector model for representing news articles using named entities when a few training examples are available.

For the future work, we plan to 1) analyze the performance of feature selection methods and SVM algorithms for imbalanced data sets, 2) compare to other related models and algorithms, and 3) experiment on large-scale data sets.

Acknowledgement. We gratefully acknowledge funding from the National Science Foundation of China under grants 61170165.

References

[1] Liu, T., Yang, Y., Wan, H., Zeng, H., Chen, Z., Ma, W.: Support vector machines classification with a very large-scale taxonomy. ACM SIGKDD Explorations Newsletter 7, 36–43 (2005)
[2] Gabrilovich, E., Markovitch, S.: Text categorization with many redundant features: Using aggressive feature selection to make svms competitive with c4. 5. In: Proceedings of the 21th International Conference on Machine Learning, p. 41 (2004)
[3] Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes 30, 3–26 (2007)
[4] Dumais, S., Chen, H.: Hierarchical classification of web content. In: Proceedings of the 23rd ACM SIGIR Conference, pp. 256–263. ACM (2000)
[5] Xue, G., Xing, D., Yang, Q., Yu, Y.: Deep classification in large-scale text hierarchies. In: Proceedings of the 31st ACM SIGIR Conference, pp. 619–626. ACM (2008)
[6] Silla, C., Freitas, A.: A survey of hierarchical classification across different application domains. Data Mining and Knowledge Discovery 22, 31–72 (2011)
[7] Koller, D., Sahami, M.: Hierarchically classifying documents using very few words. In: Proceedings of the Fourteenth International Conference on Machine Learning, pp. 170–178. Morgan Kaufmann Publishers Inc. (1997)
[8] Chen, H., Dumais, S.: Bringing order to the web: Automatically categorizing search results. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 145–152. ACM (2000)
[9] Punera, K., Ghosh, J.: Enhanced hierarchical classification via isotonic smoothing. In: Proceedings of the 17th International Conference on World Wide Web, pp. 151–160. ACM (2008)
[10] Bennett, P., Nguyen, N.: Refined experts: improving classification in large taxonomies. In: Proceedings of the 32nd ACM SIGIR Conference, pp. 11–18. ACM (2009)
[11] Liu, B., Li, C.: An efficient feature selection method using named entity recognition for chinese text categorization. In: Proceedings of the International Conference on Machine Learning and Cybernetics, pp. 3527–3531 (2009)
[12] Kiritchenko, S.: Hierarchical text categorization and its application to bioinformatics. PhD thesis, Citeseer (2005)
[13] Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys (CSUR) 34, 1–47 (2002)
[14] Yang, Y., Pedersen, J.: A comparative study on feature selection in text categorization. In: Machine Learning International Workshop, pp. 412–420. Morgan Kaufmann Publishers, Inc. (1997)
[15] Baeza-Yates, R., Ribeiro-Neto, B.: Modern information retrieval, vol. 82. Addison-Wesley, New York (1999)
[16] Platt, J.: Fast training of support vector machines using sequential minimal optimization. In: Advances in Kernel Methods, pp. 185–208. MIT Press (1999)
[17] Shevade, S., Keerthi, S., Bhattacharyya, C., Murthy, K.: Improvements to the smo algorithm for svm regression. IEEE Transactions on Neural Networks 11, 1188–1193 (2000)
[18] Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd ACM SIGIR Conference, pp. 42–49. ACM (1999)
[19] Kumaran, G., Allan, J.: Text classification and named entities for new event detection. In: Proceedings of the 27th ACM SIGIR Conference, pp. 297–304. ACM (2004)