Unified Classification Model for Geotagging Websites

Alexey Volkov, Pavel Serdyukov
Yandex LLC
Moscow, Russian Federation
{ark-kum,pavser}@yandex-team.ru

ABSTRACT

The paper presents a novel approach to finding regional scopes (geotagging) of websites. It relies on a single binary classification model per region type to perform the multi-class classification and uses a variety of features of different nature that have not been yet used together for machine-learning based regional classification of websites. The evaluation demonstrates the advantage of our *one model per region type* method versus the traditional *one model per region* approach.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

1. INTRODUCTION

Many web resources are relevant only to the users from some specific location. To provide results for the region-specific queries, the search engine needs to know not only the user's location, but also the regional focus of the websites.

The problem we address in this paper is the detection of the regional focus (geotagging) of a website. For each website we want to find the set of regions where the people, interested in the website's content, are located. There were approaches to this problem which involved searching the site's pages for entities e.g. toponyms, phone and ZIP codes[2]. These works did not consider any machine-learning based approach to regional classification and ignored any terms appearing on a page that were not known to be location-specific entities in advance. A number of works on regional classification of other types of web content (online photos[5], tweets[1]) used traditional approach to text classification and learned a classification function for each region present in the training set of geotagged resources, using any terms as features.Such an approach obviously suffers from the lack of training data for less popular regions, where people do not provide enough geotagged content to learn a good predictor.

In contrast to the above-mentioned approaches, we propose a classification framework, where both entity- and term-based features are used together to provide a high-quality regional classification of web-sites. Moreover, we rely on a single model per region type (e.g. city), rather than on an individual model for each region, what greatly solves the problem of data sparsity and allows for acceptable classification accuracy even for websites from the regions with just a few or even no geotagged websites in the training data.

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

2. METHOD

Our goal is to associate a correct set of regions with every website. This is a standard multi-label classification problem where each object can belong to zero or more classes.

Most of the previous works on regional classification of web resources (see Section), that employed machine learning, used a set of binary classifiers with the one-versus-all approach for the task of multi-class classification. We want to present a system that is able to use machine learning to train a single model that is able to detect multiple regions with the same or better quality than the traditional "one model per region" system. The main advantage of our approach to the regional classification problem is using a single model per region type for the multi-class classification. It is faster to train one model than thousands. More importantly, the model turns out to be better trained since it has much more positive examples than the "per region" models.

Data sources and features. The feature vectors used in our classification system combine the data external to the website's content with the information, extracted from the full text of the indexed web documents, and the geographic information, mined from the documents using entity extraction techniques. Our system does not use textual features directly, but rather uses the results of a third-party Bayesian text classifier which uses terms as features.

We use the following data sources: mentions of the region's name, zip code or phone code; website's IP address, *top level domain*; language stats of the website's pages; the result of a third-party term-based regional classifier.

The object of classification. Instead of training a set of models that classify websites as relevant or irrelevant to the corresponding regions, our system trains a model that classifies the <site,region>pairs. In other words, we train a ranking model which aims to infer the relevance of a region to a website, in a similar way that is followed, for example, by Learning to Rank algorithms[4] to infer the relevance of a website to an arbitrary query. Consequently the multi-class classification problem becomes a binary classification problem, greatly reducing the computational complexity.

Building the feature table. To train the classification model we must first find a group of *candidate regions* for each website thus forming the <site, region>pairs. A region is *candidate* for a given website if the data gathered for the website (e.g. zip codes, suggestions of the terms-based clas-

sifier, IP address) has any connection to that region. That way, the average number of website's candidate regions is much lower than the total number of all possible regions. For each <site, *candidate* region>pair we construct a feature vector using the features created using the data from the listed sources.

The feature vector. Each data source contributes some elements to the feature vector. Simple data sources just provide one or more numerical values. For sources that provide multiple values (e.g. the language stats) the value ratios are also added. Regional data sources (which for each website can provide some value for any region) each contribute 15 elements to the vector. For a given website, let $v(r)$ be the value that the data source has for some region r (e.g. the number of zip codes from r). Let $p(r)$ be the prior probability of a region r (region frequency in the training set). When applied to a set of regions R these functions are just summed: $v(R) = \sum v(r), r \in R$, $p(R) = \sum p(r), r \in R$. Used region sets: *All* is the set of all regions; *Cur* is the region of the current <site,region>pair; $Rest = All \setminus Cur$; *Rival* is the non-current region with the highest value: $Rival = argmax\ value(r), r \in All$; $RRest = Rest \setminus Rival$. Let $ratio(R) = v(R)/v(All)$, $rel(R) = v(R)/v(Cur)$, $nratio(R) = ratio(R) * p(All)/p(R)$. When 4 functions ($v$, *ratio*, *rel*, *nratio*) are applied to 4 region sets (*Cur*, *Rival*, *Rest*, *RRest*) we get 16 values. Since $rel(Cur)$ always equals 1, we exclude it.

Training the model. We employ a machine learning algorithm that uses gradient boosting and binary decision trees[3] to build the model. By applying the trained model to the feature vectors, we obtain a probabilistic score for each <site,region>pair. The resulting ranking of countries and cities w.r.t. the website is cut using selected thresholds for classification scores. The thresholds are selected to maximize the F_1-measure of the result on the training set.

A system of classifiers. Both countries and cities are geographical regions, but they differ too much, and, while we could use a single model for both countries and cities, it is better to have separate models for each of these types of regions (e.g. "country", "city"). Moreover, the problem of determining whether the website is relevant to any specific city (country) or has no regional focus is different from the problem of associating a specific city (country) with the website. Therefore our system uses four classification models for that task. Two for determining the website's relevance to a specific city (country) and another two for determining whether the website is relevant to any specific city (country) or is "global" (countrywide or worldwide). The results of the latter classifiers are used by other two classifiers as single value features.

3. EXPERIMENTAL RESULTS

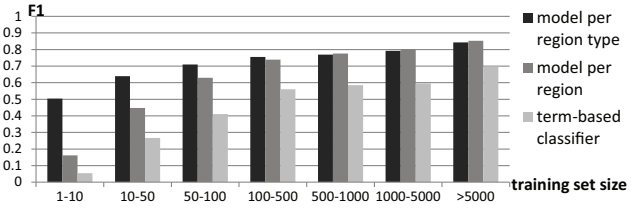
The sample dataset we used was a large website directory internally maintained by a major search engine. Directory editors manually assigned a set of about 115,000 websites to cities, countries or the worldwide categories.

For each classification system we calculated micro-averaged F_1-measure for different region types: countries, cities and the two special global classes - "Global" (worldwide) and "National" (countrywide) (See Table 1). For cities we also calculated the macro-averaged F_1-measure. To ascertain that our system works better than other systems for the small regions (small cities), we compared the systems' classification performance for cities with different amounts of available training data (See Figure 1).

Table 1: Micro- (and macro-) averaged F_1-measure

	Country	City	Global	National
our system	**0.93**	**0.83 (0.68)**	**0.65**	**0.70**
w.o. term-based f.	0.92	0.79 (0.66)	0.59	0.66
model per region	0.93	0.80 (0.51)	0.65	0.70
term-based f. only	0.91	0.63 (0.34)	0.56	0.63

Figure 1: Different ranges of training set sizes



One model per region type versus one model per region. To measure only this specific aspect, we built a classification system identical to the proposed one (same feature vectors and machine learning algorithm) except that it built an individual model for each region, not a region type. We see that our system performs much better than the "model per region" for less popular cities leading to a substantially increased macro-averaged F_1-measure.

Term-based features versus other features. Our system normally uses the output of an external term-based classifier as a feature. That classifier is built per-region, so one may argue that the system that uses its output does not adhere to the "one model per region type" idea. This is not the case since, for the system, the term-based classifier is an opaque external source. As we see, it is clear that combining the features in a single system allows us to achieve better classification quality.

Classification without training. Another interesting aspect of our system is the ability to detect regions that were not present in the training set. To evaluate this ability we conducted an experiment. We split our learning sample data into the training and test sets so that all websites belonging to a given region are either all in the training set or in the test set. This ensured that the regions used for testing were not present in the test set. We trained the city classification model using the websites from the training set and got $F_1 = 0.78$ on the training set and 0.82 on the test set.

Conclusion: The results show that our model unifying features of various types and built per region type is better than both approaches to regional classification applied in isolation and better than a model built per region. Such a model works even much better for less popular regions (less than 500 training examples). The quality of classification is rather high even when the region is almost or completely missing from the training set.

4. REFERENCES

[1] Z. Cheng, J. Caverlee, and K. Lee. You are where you tweet: a content-based approach to geo-locating twitter users. CIKM '10.
[2] J. Ding, L. Gravano, and N. Shivakumar. Computing geographical scopes of web resources. VLDB, 2000.
[3] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference and prediction.*
[4] T.-Y. Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3, 2009.
[5] P. Serdyukov, V. Murdock, and R. van Zwol. Placing flickr photos on a map. SIGIR, 2009.