ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-1101

Unsupervised Feature Generation using Knowledge Repositories for Effective Text Categorization

Rajendra Prasath [1] and **Sudeshna Sarkar** [2]

Abstract. We propose an unsupervised feature generation algorithm using the repositories of human knowledge for effective text categorization. Conventional *bag of words* (BOW) depends on the presence / absence of keywords to classify the documents. To understand the actual context behind these keywords, we use knowledge concepts / hyperlinks from external knowledge sources through content and structure mining on Wikipedia. Then, the features of knowledge concepts are clustered to generate knowledge cluster vectors with which the input text documents are mapped into a high dimensional feature space and the classification is performed. The simulation results show that the proposed approach identifies associated features in the text collection and yields an improved classification accuracy.

1 Introduction

Identifying the relevant text document in the growing text collection related to a specific topic demands *Text Categorization* (TC) - the process of assigning predefined category labels to text documents based on their content[6]. Recently TC is improved with external knowledge sources. Murata *et* al. weighted the terms based on the structural information that specifies the importance [4]. Gabrilovich proposed Explicit Semantic Analysis to generate informative and discriminative features from Wikipedia & Open Directory Project[3]. Recently, Latent Dirichlet Allocation based topic modeling has been applied to improve text classification [1]. The combination of neural network and LSI is applied to find the associative relationship among terms [7]. Expanding the BOW representation with semantic relations is described in [5]. Here, we apply an unsupervised feature generation using Wikipedia, to improve the text representation.

2 Text Categorization System

Mathematically, text categorization is the task of assigning a boolean value to each pair $< d_i, c_j > \in \mathcal{D} \times \mathcal{C}$, where $\mathcal{D}$ is the set of n documents and $\mathcal{C} = \{c_1, c_2, \cdots, c_m\}$ is a set of m predefined categories. Assign T (TRUE) to $< d_i, c_j >$ to classify d_i under c_j and assign F (FALSE) to indicate not to classify d_i under c_j. More formally the task is to approximate the unknown target function $\Phi' : \mathcal{D} \times \mathcal{C} \rightarrow \{T, F\}$ by means of a function $\Phi : \mathcal{D} \times \mathcal{C} \rightarrow \{T, F\}$ called the *classifier* such that Φ' and Φ coincide as much as possible. The classification accuracy is based on the model which coincides with the target function as closely as possible. The proposed text categorization system has two stages:

[1] Norwegian University of Science and Technology, No - 7491, Trondheim, Norway, email: rajendra@idi.ntnu.no; rajendra@cse.iitkgp.ernet.in
[2] Indian Institute of Technology, Kharagpur - 721 302, India, email: sudeshna@cse.iitkgp.ernet.in

2.1 Building Knowledge Concepts

Wikipedia is an encyclopedia with heavy linking and explicitly labeled among articles. Many types of semantic knowledge exists, but have to be extracted from Wikimedia dumps using:

(i). *Content mining* which refers to searching the article content for relevant knowledge. This implies wiki concepts that may either be definitions or illustrations or explanations, but explain the given feature in terms of its semantic relatedness.

(ii). *Structure mining* which refers to extracting knowledge from structural features such as the link graph or the inner structure of an article in Wikipedia. This implies wiki hyperlink vectors - containing words which are tagged with anchors to related wiki articles.

2.1.1 Example

Consider the wiki article featuring "abu dhabi"(a few lines):

"'Abu Dhabi"' (literally "Father of [[Gazelle]]") is the [[capital city—capital]] and second largest city of the [[United Arab Emirates]]. It is also the capital and largest city of the emirate of [[Abu Dhabi (Emirate)—Abu Dhabi]], which is the largest of the seven [[emirates of the United Arab Emirates]] by size. It was said by [[CNN]] to be the richest city in the world and is located in the center of the northern part of the [[United Arab Emirates]].

The extracted wiki concept is as follows:

abu dhabi | capital city | second largest city | united arab emirates | seven emirates | cnn | richest city | world | located | center | northern part

The extracted wiki hyperlink vector is as follows:

abu dhabi | persian gulf | saudi arabia | oman | emirate | abu dhabi emirate | khalifa bin zayed al nahayan | gazelle | capital city | abu dhabi emirate | united arab emirates | sheikh | emirates of the united arab emirates | cnn | united arab emirates | persian gulf

Here, CNN does not impact but anchored and "richest city" is useful but not anchored. Similarly "Gazelle" is anchored but hardly useful [Gazelle refers to swift animals, able to reach high speeds for long time]. The extracted content is again refined to filter the links on section, reference, external URLs, category and language tags.

2.2 Text Categorization using Unsupervised Feature Generation

Clustering identifies a pattern in the bunch of unlabeled data and organizes data into groups whose members are related in some way falling close to each others' context. While exploring the possibilities to identify the contexts of text fragments, feature space grows. This could be solved by using dimensionality reduction techniques. Using the extracted wiki concepts, we first obtain the co-occurrence

information of each pair of the features in the wiki concept vectors. Using feature co-occurrence information, we build a weighted graph $G = (V, E, A)$ where $|V| = n$, the number of features; $|E|$ points to the number of edges and A is an adjacency matrix ($|V| \times |V|$) whose nonzero entries correspond to the edge weight between a pair of features. Then we apply the kernel-based multilevel clustering algorithm [2] on the weighted graph with the desired number of m partitions(=10% of total features). Similar cluster vectors are formed using wiki hyperlink vectors and then text categorization is employed with hyperlink cluster vectors.

Algorithm 1 Text Categorization using Cluster Vectors

Input: A set of n text documents $D = \{d_1, d_2, d_3 \cdots, d_n\}$
A set of predefined category labels C

Description:

Build Wiki Cluster Vectors:

Extract wiki concept / hyperlink vectors from Wikipedia
for each feature f_i in the wiki concept / hyperlink vector **do**
  identify the existence of edges from f_i to all other features using co-occurrence information
  Store the co-occurring features with their edge weight
end for
Use kernel-based multilevel graph clustering algorithm and perform clustering to generate cluster IDs
For every cluster ID, generate feature cluster vectors
Index these feature cluster vectors

Categorization with Wiki Cluster Vectors:

Get the preprocessed text documents collection
for each processed document d_i in D **do**
  for each unique feature in d_i **do**
    Filter out its cluster ID
    Map the given feature in terms of its cluster ID
    Augment Text fragments with cluster ID mappings
  end for
end for
Build classifier on these mapped text documents (containing only cluster IDs) and record the classification accuracy

Output: The category label for each document is identified.

3 Experimental Settings

`Reuters-21578`: News wire articles with top 5 classes: TOPICS, PLACES, PEOPLE, ORGS and EXCHANGES.

`rainbow`[3] - with classifiers: Naive Bayes (NB), $k-$Nearest Neighbors[k-NN] (with k=30) and support vector machines(SVM).

`Split`: random - 60% training set and 40% the test set.

`Wikipedia`: October 2007 snapshot (English)

We perform stop word removal using SMART list and do not use any stemmer(due to the support of Wikipedia for spelling variations). We used `graclus`[4] for finding the clusters of the given graph.

3.1 Results

We have considered 60,000 wiki concepts and all five classes of Reuters-21578 with 22 splits each with 1000 documents. The proposed algorithm, with Naive Bayes method, achieves 55.05% accuracy and with $k-$NN method, achieves improvement in PLACES category with 83.93%accuracy. In another experiment, hyperlink vectors are clustered by explicitly eliminating noises during the process of building cluster vectors. Figure. 1 shows the effect of hyperlink cluster vectors on different splits of Reuters with k-NN Method. As the number of hyperlink vector increases, the classification accuracy also increases. This is due to the increase in the distinguishable terms with the increase of terms in each link cluster. Table. 1 shows the classification accuracy on Reuters-21578 by different classifiers with wiki hyperlink clusters formed from different values of wiki hyperlink vectors. The performance approaches stability as we increase the number of wiki hyperlink vectors.

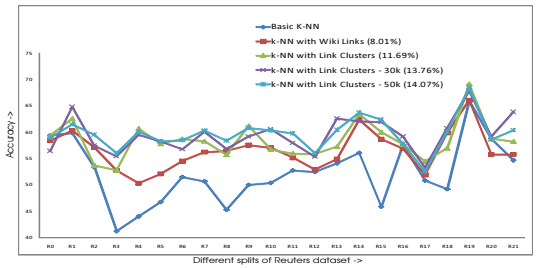

Figure 1. Effects of $k-$NN with m(=30k, =50k) Wiki hyperlink clusters

Categories	NB		$k-$NN		SVM	
	30K	50K	30K	50K	30K	50K
EXCHANGE	57.97	60.14	40.15	34.09	67.39	63.04
ORGS	71.56	71.10	50.26	62.69	75.23	72.94
PEOPLE	56.49	56.49	36.64	45.80	70.23	70.61
PLACES	63.75	64.40	48.21	49.61	72.55	70.18
TOPICS	56.98	56.21	44.04	44.65	48.41	50.10
AVERAGE	61.35	61.67	43.86	47.37	66.76	65.37

Table 1. Results of top 5 categories of Reuters with wiki cluster vectors

4 Conclusion

We investigated the effectiveness of domain specific knowledge based feature clusters for text categorization. These feature clusters are generated based on its context relationships. Content mining of any domain specific knowledge repositories could be used for generating knowledge concepts. Structural properties simply generate the concept vectors without explicit semantic analysis. The simulation results show that the proposed system improves the classification accuracy significantly.

REFERENCES

[1] S Banerjee, 'Improving text classification accuracy using topic modeling over an additional corpus', in *SIGIR '08*, pp. 867–868, NY, (2008).
[2] I.S. Dhillon, Y. Guan, and B. Kulis, 'Weighted graph cuts without eigenvectors a multilevel approach', *IEEE Trans. Pattern Anal. Mach. Intell.*, **29**(11), 1944–1957, (2007).
[3] E.Gabrilovich, *Feature Generation for Textual Information Retrieval Using World Knowledge*, Ph.D. dissertation, Technion, Israel, 2006.
[4] Murata *et* al, 'Automatic indexing: An experimental inquiry', *J. of the Association for Natural Language Processing*, **7**(2), (2000).
[5] Wang *et* al., 'Using wikipedia knowledge to improve text classification', *Knowl. Inf. Syst.*, **19**(3), 265–281, (2009).
[6] Sebastiani, 'Machine learning in automated text categorization', *ACM Computing Surveys*, **34**, 1–47, (2002).
[7] W. Wang and B. Yu, 'Text categorization based on combination of modified back propagation neural network and latent semantic analysis', *Neural Comput. Appl.*, **18**(8), 875–881, (2009).

[3] www.cs.cmu.edu/~mccallum/bow
[4] www.cs.utexas.edu/users/dml/Software/graclus.html