Machine Learning-Based Web Documents Categorization by Semantic Graphs

Francesco Camastra[1], Angelo Ciaramella[1],
Alessio Placitelli[2], and Antonino Staiano[1]

[1] Dept. of Science and Technology, University of Naples "Parthenope", Isola C4, Centro Direzionale, I-80143, Napoli (NA), Italy
{camastra, angelo.ciaramella, staiano}@ieee.org,
[2] Vitrociset s.p.a., Via Tiburtina, 1020 - 00156 Roma, Italy
alessio.placitelli@gmail.com

Abstract. This work aims to approach web pages categorization by means of semantic graphs and machine learning techniques. We propose to use a semantic graph that can provide a compact and structured representation of the concepts present in a document in order to take into account the semantic information. The semantic graph allows determining a map of the semantic areas contained in the document and their relationships w.r.t. a particular concept or term. The semantic measure between the terms is calculated by using the lexical database (i.e., WordNet). The document categorization is accomplished by a machine learning technique. We compare the performance of both supervised and unsupervised techniques (i.e., Support Vector Machine and Self Organizing Maps, respectively). The proposed methodology has been applied for classification and agglomeration of benchmark and real data. From the analysis of the results it can be shown that the model trained with semantic features obtains satisfactory results, in particular by using the unsupervised machine learning technique.

1 Introduction

With the dramatically quick and explosive growth of information available over the Internet, World Wide Web has become a powerful platform to store, disseminate and retrieve information as well as mine useful knowledge [3]. Information is mostly in the form of unstructured data. As the data on the web has been growing, it has lead to several problems such as increased difficulty of finding relevant information and extracting potentially useful knowledge. Web mining is an emerging research area focused on the application of data mining techniques to discover patterns from the Web. According to analysis targets, Web mining can be divided into three different types, which are Web usage mining, Web content mining and Web structure mining. In this work we address the problem of Web content mining. Web content mining extracts information from different Web sites for its access and knowledge discovery. In particular, we study a novel methodology for Web pages categorization considering the textual content.

S. Bassis et al. (eds.), *Recent Advances of Neural Networks Models and Applications*,
Smart Innovation, Systems and Technologies 37, DOI: 10.1007/978-3-319-18164-6_8

In the past 20 years, the number of text documents in digital form has grown exponentially [12]. As a consequence of this exponential growth, great importance has been put on the classification of documents into groups that describe the content of the documents. The function of a classifier is to merge text documents into one or more predefined categories based on their content. Each document can belong to several categories or may present its own category. In [9] the authors review the Web-specific features and algorithms that have been explored and found to be useful for Web page classification. Most approaches described in literature do not consider the semantic information in the document and therefore in some cases may not perform adequately. In [1] an approach to incorporate concepts from background knowledge into document representations for text document classification (by using boosting machine learning technique) has been proposed. To extract concepts from texts, the authors have developed a detailed process, that can be used with any ontology with lexicon. Our work aims to approach web pages categorization by means of semantic graphs and machine learning techniques [1]. The semantic graph allows determining a map of the semantic areas contained in the document and their relationships w.r.t. a particular concept or term. The similarity between the terms is calculated by using the lexical database (i.e., WordNet) and the pages are represented using a TF-IDF (*Term Frequency-Inverse Document*) mechanism.
The paper is organized as follows. In Section 2, we introduce the categorization problem of documents, and, in Section 3 the TF-IDF methodology is presented. In Section 4 we describe the semantic graph and how to use it for the the TF-IDF methodology. In Section 5, the experimental results on benchmark and real data are presented. Finally, some conclusions and future remarks are outlined.

2 Document Categorization

Categorization of documents refers to the problem of automatic classification of a set of documents in *classes* (or *categories* or *topics*). A common approach for text classification is formed by five steps. The first step (*tokenization*) eliminates the punctuation signs in the text. The second step (*stopping*), removes from the text the so-called *stopping words*, i.e., common words (e.g, articles, modal verbs, prepositions) that are widespread in every text and therefore cannot be used for discriminating a text. The third step is the *stemming*, where each term is reduced to own lexical root (or *stem*) by means of a *stemming algorithm* (e.g., *Porter's algorithm*). In the fourth step, the document is represented by means of a vector whose generic i-th coordinate is computed by TF-IDF (*Term Frequency-Inverse Document*) approach [11]. Finally, the document classification is performed by a machine learning technique. The approach described above does not consider the semantic information in the document and therefore in some cases may not perform adequately.

[1] The work was made when Alessio Placitelli was M. Sc. Student at University of Naples Parthenope.

3 Scoring

To extract information from a document we compute a score between a query term t and a document d, based on the weight of t in d. The simplest approach is to assign the weight to be equal to the number of occurrences of term t in document d ($tf_{t,d}$, *Term Frequency* of the term t in document d) [6]. Raw term frequency as above suffers from a critical problem: all terms are considered equally important when it comes to assessing relevancy on a query. In fact, certain terms have little or no discriminating power in determining relevance. A mechanism for attenuating the effect of terms that occur too often in the collection to be meaningful for relevance determination. An idea could be to reduce the $tf_{t,d}$ weight of a term by a factor that grows with its collection frequency. It is more commonplace to use for this purpose the document frequency df_t, defined to be the number of documents in the collection that contain a term t. Denoting the total number of documents in a collection by N, we define the inverse document frequency (idf) of a term t as follows:

$$idf_t = \log \frac{N}{df_t}. \tag{1}$$

Thus, the idf of a rare term is high, whereas the idf of a frequent term is likely to be low. We now combine the definitions of term frequency and inverse document frequency, to produce a composite weight for each term in each document. The TF-IDF weighting scheme assigns to term t a weight TF-IDF in document d given by

$$\text{TF-IDF}_{t,d} = tf_{t,d} \times idf_t. \tag{2}$$

We may view each document as a vector with one component corresponding to each term in the dictionary, together with a weight for each component that is given by equation (2).

4 Semantic Graph

In order to take into account the semantic information, we propose to use a semantic graph that can provide a compact and structured representation of the concepts present in a document. The semantic graph allows determining a map of the semantic areas contained in the document and their relationships w.r.t. a particular concept or term, called *target*. The semantic weight indicates how much the document is relevant w.r.t. the target. The semantic graph is a undirected, fully connected graph, consisting of the terms of the document connected by relations of similarity to a target term.

A semantic graph is computed starting from a single term. Let t be the term whose semantic graph has to be computed and N is the number of the most similar terms in the document, the construction of the semantic graph is performed by means of four well-defined phases: *similarity calculation*, *ranking*, *graph construction* and *semantic weight calculation*. The similarity (s) between the terms

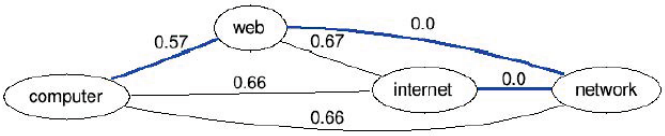

Fig. 1. Minimum spanning tree computed by the algorithm of Kruskal

is calculated by using the lexical database WordNet [7]. Next, the terms more similar to t, are ranked on the basis of a properly chosen similarity metric. Now the top N terms are used as the vertices of the undirected weighted semantic graph. For each pair of vertices an edge is created. The weight of the edge is proportional $(1 - s)$ to the semantic distance between the terms (e.g., Lin similarity in the $[0, 1]$ interval [5]). For instance, consider the construction of a semantic graph related to the target term *computer* based on a document containing the terms: *Internet*, *www*, *cat*, *network*, *software*, *computer*, *web*, and *homepage*. The information contained in the semantic graph can be represented by a single synthetic value called *semantic weight* (w_s). This value is obtained calculating the sum of the reconstructed weights $(1 - w_s)$ for the arcs belonging to the Minimum Spanning Tree (MST) of the semantic graph. The weight indicates that the semantic parsed document, represented through the semantic graph, is relevant to the target word. The higher the value of the weight, the more the document refers to the subject matter from the end target. On the contrary, the smaller this value, the less the document identifies the target. In Figure 1, the minimum spanning tree computed by the algorithm of Kruskal is presented. The final semantic weight is proportional to this estimated weight. Summarizing, the steps of the proposed categorization process are as follows. The first three steps are the same as in usual categorization process (i.e., tokenization, stop words removal, stemming), in the fourth step, to each term it is associated the semantic weight, instead of the usual TF-IDF value. In Figure 2 , the use of a semantic graph in a bag of words mechanism[6], is shown. Finally, using the TF-IDF vectors we have performed the document categorization by means of a machine learning technique. In this specific case, both Support Vector Machine (SVM) [2] and Self Organizing Maps (SOMs) [4] have been applied.

5 Experimental Results

The proposed methodology has been applied for classification (SVM) and agglomeration (SOM) of two different corpora. To evaluate the performance, the results are compared with those obtained by the standard TF-IDF mechanism considering different metrics. For the SOM, we consider the quantization error (QE), the topographic error (TE) and the combined error (CE). Regarding the SVM, the percentage of documents correctly classified is evaluated (for further results as confusion matrix, measures of precision and recall see [8]).

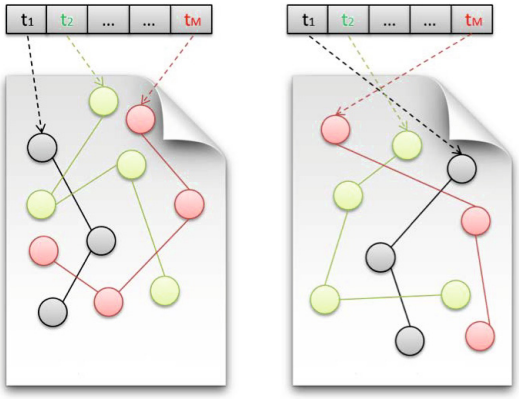

Fig. 2. Bag of words mechanism and semantic graph: extraction of features from documents

In a first phase of validation, the Reuters 21578 corpus has been considered [10]. This corpus was issued by the multinational Reuters in 2000 and made publicly available for research purposes. We used only three categories for a total of 390 documents divided as follows: Cocoa (55 documents), Money Supply (138 documents) and Ship (197 documents). In Tables 1 and 2, we report the results applying SOM and SVM techniques, respectively. Using SOM, in the case of semantic weights, a QE of 41.961, a TE of 0 and a CE of 48.414 are obtained. Instead the SVM, as reported in Table 1, has allowed obtaining a percentage of correct classification of 97.17% (379 documents). In the case of standard TF-IDF, SOM has detected a QE of 43.675, a TE of 0.005 and a CE of 49.820. The percentage of correct classification obtained through SVM is of 93.58% (365 papers).

Successively, a scraping software has been used for analyzing a web page and extract the main content excluding tags, templates and other kind of unnecessary code [8]. In order to build the corpus, the scraper was launched for five days (from March 15, 2013 to March 20, 2013) by performing the scraping of 1995 journalistic news of 5 different categories: *politics*, *sport*, *business*, *science* and *entertainment*. In Table 3, we describe the categories and information sources for the corpus [2]. The dictionary of the processed corpus is composed initially of 27305 terms. The removal of low-frequency terms leads to the elimination of 15619 words, decreasing the size of the dictionary terms to 11686 terms.

The SVM training was performed using a linear kernel and a cost coefficient of $C = 1.0$. The main objective of these experimental results is to compare TF-IDF and SWA approaches. For this reason we chose to use a simple linear kernel and to consider the same weight between the slack variability penalty and the margin in SVM optimization mechanism [2]. A 10-fold cross-validation is performed. On

[2] The corpora are available on request.

Table 1. SVM results: Reuters and scraped corpora

SVM	training set	training set	test set
TF-IDF	93.58%	81.35%	62%
SWA	97.71%	79.79%	70.36%
	Reuters	Real data	Real data

Table 2. SOM results: Reuters and scraped corpora (QE = Quantization Error; TE = Topographic Error; CE = Combined Error)

SOM	QE	TE	CE	QE	TE	CE	Test set	Test set
TF-IDF	41.96%	0.0%	48.41%	75.04%	0.01%	81.74%	83.19%	75.12%
SWA	43.67%	0.0%	49.82%	47.41%	0.04%	58.43%	86.68%	79.37%
	Reuters	Reuters	Reuters	Real data	Real data	Real data	Real data	Feature Selection

Table 3. Information sources used for the corpora (feed RSS)

Category	**Information sources**
Politics	The Guardian, The Telegraph, The Scotsman, BBC
Sports	The Guardian, The Independent, The Telegraph, Daily Mail, The Express, The Daily Star, The Scotsman, BBC
Business	The Guardian, The Telegraph, Daily Mail, The Express, BBC
Science	The Guardian, The Independent, The Telegraph, Daily Mail, The Express, The Daily Star, The Scotsman (Technology), BBC
Entertainment	The Guardian (Movie), The Telegraph, Daily Mail, The Express, The Scotsman, BBC

Table 4. Test corpus

Category	# of documents	Words (average)	Characters (average)
Politics	47	718	4224
Sport	210	523	3057
Business	165	590	3479
Science	141	584	3585
Entertainment	137	595	3448

the standard TF-IDF vectors, the classification percentage is of 81.35% (on the overall training set). In the unsupervised case, the SOM has a height of 17 and a width of 13 neurons. The estimated QE is 75.047, the TE is of 0.015 while the CE is 81.740. By using the semantic wights the SVM classification is of 80.760%. Instead, SOM has produced a QE of 47.410, a TE of 0.047 and a CE of 58.433. In Figure 3 we show the comparison between the topological mapping of the SOM obtained with TF-IDF and by using semantic weights, respectively. The results and the figures show that by using semantic weights is possible to obtain a more regular topographic map. Moreover, the generated models are evaluated through a test corpus composed by 700 terms (see Table 4 for details). We obtain a percentage of 62% of correct classifications using the TF-IDF features and of 70.36% using

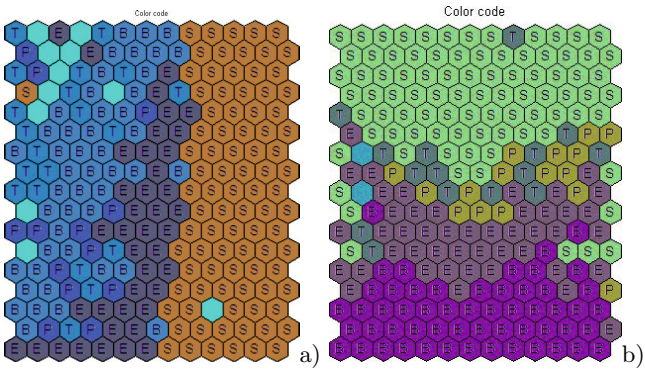

Fig. 3. Topographic result of the SOM: a) with TF-IDF features; b) with semantic features

semantic features. The performance obtained on the trained SOM are of 83.19% of correct classification for TF-IDF and of 86.68% for the semantic case.

Finally, from the previous corpus we generate a reduced corpus obtained by using a semantic feature selection. The semantic feature selection is obtained by using an aggregation approach based on the similarity measures and WordNet. In particular, the words are chosen by using an agglomerating rate and considering the most dissimilar ones for building the bag of words [8]. In this case we note that the best performance are obtained using the SOM with semantic weights (79, 37% of perfect classification).

6 Conclusions

In this paper, we presented a methodology for web pages categorization by means of semantic graphs and machine learning techniques. The semantic graph allows determining a map of the semantic areas contained in the document and their relationships (i.e., semantic metric) w.r.t. a particular concept or term. The document categorization is accomplished by means of a supervised or unsupervised machine learning technique, Support Vector Machine (SVM) and Self Organizing Maps (SOM), respectively. The model that uses semantic features and trained on the Reuters corpus obtains better results for both SVM and SOM. For the other corpora, the best results are obtained by SOM and semantic weights. We can consider that a category may present concepts strongly correlated with the other categories and this behavior can be better managed by an unsupervised mechanism. We, however, wish to highlight that by using semantic weights a Web Page can also not contain a specific term but it contains correlated concepts. In the next future the authors will focus their attention on the use of different parameters for the machine learning techniques, different semantic metrics and on the categorization of documents also using images, audio and video.

References

1. Bloehdorn, S., Hotho, A.: Boosting for text classification with semantic features. In: Mobasher, B., Nasraoui, O., Liu, B., Masand, B. (eds.) WebKDD 2004. LNCS (LNAI), vol. 3932, pp. 149–166. Springer, Heidelberg (2006)
2. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning 20(3), 273–297 (1995)
3. Divya, C.: Mining Contents in Web Pages and Ranking of Web Pages Using Cosine Similarity. International Journal of Science and Research (IJSR) 3(4) (2014)
4. Kohonen, T.: The self-organizing map. Proceedings of the IEEE 78(9), 1464–1480 (1990)
5. Lin, D.: An information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning, San Francisco, vol. 1, pp. 296–304 (1998)
6. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)
7. Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.J.: Introduction to wordnet: An on-line lexical database. International Journal of Lexicography 3(4), 235–244 (1990)
8. Placitelli, A.P.: Categorizzazione di pagine web mediante grafo semantico e tecniche di machine learning, MSc dissertion, University of Naples "Parthenope" (2013)
9. Qi, X., Davison, B.D.: Web Page classification: Features and algorithms. ACM Computing Surveys (CSUR) 41(2), 12 (2009)
10. `http://www.daviddlewis.com/resources/testcollections/reuters21578/`
11. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. In: Information Processing and Management, pp. 513–523 (1988)
12. Trstenjaka, B., Mikacb, S., Donkoc, D.: KNN with TF-IDF based Framework for Text Categorization. Procedia Engineering 69, 1356–1364 (2014)