TRASH ARTICLE DETECTION USING CATEGORIZATION TECHNIQUES

Christos Bouras
Computer Engineering and Informatics Department, University of Patras and
Research Academic Computer Technology Institute
N. Kazantzaki, Panepistimioupoli Patras, Greece
bouras@cti.gr

Vassilis Poulopoulos
Computer Engineering and Informatics Department, University of Patras and
Research Academic Computer Technology Institute
N. Kazantzaki, Panepistimioupoli Patras, Greece
poulop@cti.gr

George Tsichritzis
Computer Engineering and Informatics Department, University of Patras
N. Kazantzaki, Panepistimioupoli Patras, Greece
tsixritzis@ceid.upatras.gr

ABSTRACT

We explore techniques for detecting news articles containing invalid information, using the help of text categorization technology. The information that exists on the World Wide Web is huge enough in order to distract the users when trying to find useful information. In order to overcome the large amounts of data many methodologies of text categorization have been presented. One major problem we have to deal with is that many articles fetched by a crawler, then stored in a back-end database, and finally given as an input to a categorization subsystem, may not contain valid information for the user (trashy articles). This may lead to the user losing his trust towards the system. In this paper, we analyze the special properties of trashy news articles' categorization that allows us to detect them and we propose a specific methodology for trash detection. Finally, we evaluate the proposed algorithm on a news categorization system and we depict the overall benefit of a trash detection mechanism on the system.

KEYWORDS

trash articles, categorization, news articles, trash detection

1. INTRODUCTION

Last years, internet users have reached remarkable numbers. Additionally, the web pages along with the information that resides in them, create a chaotic condition for the World Wide Web. This condition is neither static nor stable, but a dynamic, and continuously changing one.

With the rapid growth of online information, it is necessary to deploy text categorization techniques in order to organize the data. The goal of text categorization is the classification of documents into a fixed number of predefined categories using some criteria which vary from one technique to another. Among the state of the art algorithms used for text categorization we can find variants of support vector machines algorithm (SVM), Bayes Nets and boosting approaches. Also there exist simpler approaches for text categorization that is proven to work particularly well such as simple-cosine similarity and KNN.

Support vector machines are very powerful tools, originally designed for binary classifications. The main idea of the support vector machine's algorithm is the construction of a hyperplane that acts as a decision space in such a way that the margin of separation between positive and negative examples is maximized. The large margin criterion is generalized to multiclass cases as shown in [Crammer K. and Singer Y., 2001].

Additionally, many state of the art algorithms dealing with SVM try to reduce the dimension of the feature space, using SVD/LSI as described in [H. Kim et al., 2005] or using an aggressive feature selection [Gabrilovich E., Markovitch S., 2004].

Another interesting approach to text categorization is presented by [F. Peng et al., 2004] that try to augment the naive Bayes text classifier by including observation dependencies, which form a Markov chain, and use techniques from statistical n-gram language modeling. A Naive Bayesian classifier is an algorithm for supervised learning that stores a single probabilistic summary for each class and assumes conditional independence of the attributes given the class. What [F. Peng et al., 2004] proposed, is the relaxation of some of the independence assumptions of naive Bayes, allowing a local Markov chain dependence in the observed variables.

Apart from the aforementioned methods, many techniques are used to improve the performance of the categorization algorithms, using encyclopedic knowledge or other semantic features to enhance the training set. In the effort described in [Gabrilovich E., Markovitch S., 2006] the authors use extensive encyclopedic knowledge (Wikipedia) to improve document representation for the text classification. The fact is, that many state of the art technologies try to escape the traditional bag of words model and use semantic features in order to perform better categorization results.

In a typical news categorization system, after the crawler fetches a page, text pre-processing techniques are applied in order to extract the useful content from the page [Adam et al., 2009]. Nevertheless, even after the useful text extraction has taken place, some portions of the HTML page may not contain valid information (trashy articles). For example, the text pre-processor sometimes fails to understand that user's comments under an article, a photo or a video or even a "404 not found" page, are not news articles. This is the reason that is necessary to apply another level of trash detection.

In this paper, we present an algorithm for detecting articles that contain trash, by taking advantage of the categorization procedure. Articles published in major and minor portals worldwide are usually categorized by the portal in order to offer to the website visitors the opportunity to communicate with a specific web channel and not obtain all the articles published. We take advantage of this "knowledge" about the category of an article and, moreover, we utilize the results of our classification procedure which is the relevance of each article to the predefined categories of our system. By comparing the actual category of the article (as categorized by the news portal) and the outcomes of our own categorization procedure we are able to obtain data about the validity of the text actually extracted from the original news article.

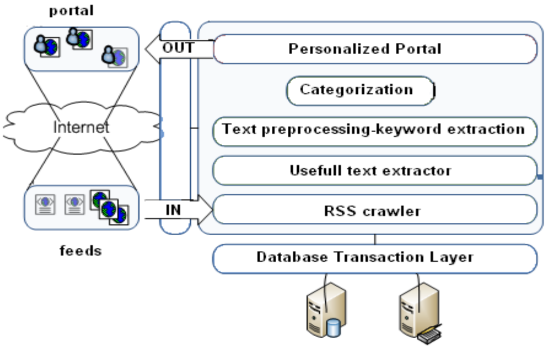
The rest of the paper is structured as follows: in section 2, a brief description of a typical news categorization system's architecture, is given for establishing the required background of the developing system. The problem description and the tools we use can be found in section 3. Section 4 gives the algorithmic outline of the trash-detection procedure. In section 5 we present the evaluation results and section 6 concludes this paper with some additional thoughts for future additions to the system.

2. ARCHITECTURE

In this section, we present the architecture of the news categorization system (Figure 1), on which we perform the trash detection algorithm. Our news categorization system consists of six layers which work independently and collaborate through a centralized back-end database. The web interface handles the incoming information flow of the mechanism which is then directed to the interior subsystems. The first levels of analysis are responsible for the crawling of the RSS feeds from major news' portals, and the discovery of the useful text portions within HTML pages. The extracted articles from each page are stored in the back-end database. Text pre-processing techniques follow and the results are sent as an input to the next levels of analysis, where core information retrieval techniques are located. These may include text categorization, text summarization and information personalization.

The categorization subsystem is based on the cosine similarity measure, dot products and term weighting calculations. More specifically, the system is initialized with a training set of articles collected from major news portals. The training set of articles is then categorized by humans. After the initialization of the system with the training set, the categorization module creates lists of keywords that are representative of a unique category, consisting of keywords with high frequency in a specific category and small or zero frequency for the other categories. As a result, a base knowledge for each category is created. Our categorization system supports 8 categories in which an article may belong, e.g. BUSINESS, ENTERTAINMENT etc. The text categorization mechanism, using the base of knowledge that exists for the categories of texts, classifies each new text, with some cross-correlation, in the existing categories. In other words, an article is classified with a degree of relevance to all of the existing categories. After the categorization procedure has taken place, the outcome is presented to the end users through the information presentation subsystem, which delivers information to the user's browser or client side desktop.

Figure 1. System Architecture



To perform our trash detection algorithm we focus upon the back-end database. The back-end database of our system contains

- the articles fetched by the crawler,
- the category of the RSS each one of the fetched articles belongs
- the relevance of every article to each of the predefined categories the system has, as a result of the categorization procedure.

It is important to be understood that every article belongs to one of the 8 categories because of the category of RSS the article comes. Additionally, each article is associated with a degree of relevance to each one of the 8 categories our system supports, as a result of the categorization procedure.

3. IMPLEMENTATION ISSUES

An article fetched by the RSS crawler, is stored in the 'articles' table (Table 1) of our back-end database. The 'RSS category' of the 'articles' table, represents the category of the RSS from which the article was taken (predefined category). The RSS feeds belong to one of the eight predefined categories, supported by our system, e.g. the RSS entitled "Sport News" from BBC[1] belongs to the category Sports. During the phase of the RSS crawling, the web article fetching mechanism, receives from the RSS the category of the articles. It is obvious, that articles that come from an RSS Feed which belongs to the Sports category will belong, with high cross-correlation, to this category, something that should also be confirmed by the categorization process in the lower levels of our mechanism. For example, Table 1 depicts an article that was retrieved from an RSS belonging to the category BUSINESS.

Table 1. A news article stored in "articles" table

Title	**Body**	**RSS category**
UN warns about higher food costs	The high food prices are already hitting many ...	Business

[1] www.bbc.co.uk – BBC News Portal

On the other hand, another table of the database (Table 2) stores for every article, the relevance to every one of the 8 pre-defined categories that our system has. The relevance of an article to the predefined categories, which is a number from 0 to 1, is depicted in the table 'articles2category' and it is a result of the the cosine similarity categorization mechanism. Table 2 depicts the relevance of an article to each of the 8 categories; thus the category BUSINESS is the category with which the article has the highest relevance, the category HEALTH is the category with the second highest relevance etc. From now on, we call MAX the highest relevance of an article, MAX-1 the second highest relevance etc.

Table 2. Article to Category Relevance

Category	Relevance
Business	26,4%
Health	10,5%
Science	9,8%
Politics	7,6%
Technology	7,3%
Education	6%
Entertainment	2,2%
Sports	1,6%

Our trashy articles detection mechanism takes into account the category of the RSS the article came from, in other words the predefined category in the ‘articles’ table, as well as, the categories, in which the article was associated by the categorization mechanism as depicted in Table 2.

4. ALGORITHM ANALYSIS

The trashy article detection algorithm takes advantage of the ‘RSS category’ field of the ‘articles’ table. From now on, we call the RSS category “predefined category”. To detect trashy articles, we compare the predefined category of every article to the categories of our system associated to this specific article during the categorization procedure. For this reason we use two heuristics. The two different heuristics are applied to each article, and if one of them succeeds then the article is marked as trash article which contains invalid information.

The first heuristic of trash detection marks an article as trash, if none of the categories that the classifier categorized an article is the same as the predefined category. As it is already mentioned, the relevance of an article to the eight predefined categories of our system is stored in a database table. From this table we can find the categories that an article is classified by our mechanism. For example, we can figure out that an article is categorized in only one category, when the MAX relevance is not close to the MAX-1 relevance. From experiments that we have conducted on the validity of the categorization mechanism it is obvious that an article is classified into a category if the second highest relevance is less than 75% of the highest relevance. Consequently, MAX-1 and MAX are not close when:

$$\text{MAX-1} / \text{MAX} < 0.75 \qquad \textbf{(1)}$$

Statistically, if we separate the categories into two groups according to the average relevance then we will observe a first group with high values of relevance which are over the average and a second group of relevance which are lower than the average. By measuring the standard deviation in each group we obtain information about how close are the values in each group. The standard output of this procedure can be summarized in a first group of two or three values that are higher than the average and a set of five or six values that are lower than the average. What we care for is the first group of values where the standard deviation is a metric that can show us the difference of the first value compared to the other one or two. By measuring the limit of the difference of the highest relevance minus the standard deviation measured on the training set that was used in order to train the categorization mechanism we extract a safe lower bound of 79% (MAX-STDEV(values higher than average)) / MAX ≈ 78,87%. By further lowering this bound in order to cover every other possible situation we conclude to the bound of 75% that is used for our analysis.

Hence, the condition (1) indicates that the categorization mechanism classified an article in only one category. Following this reasoning, an article is categorized in two categories, when the MAX relevance is close to MAX-1 relevance (2) whereas the MAX relevance is not close to the MAX-2 (3) relevance. We must denote that it is not strange for an article to belong in two categories as well (for instance an article can belong to both business and politics or entertainment and sports). The inequalities (2) and (3) indicate that the categorization mechanism classified an article in two categories.

$$\text{MAX-1} / \text{MAX} > 0.75 \qquad \textbf{(2)}$$

$$\text{MAX-2} / \text{MAX} < 0.75 \qquad \textbf{(3)}$$

To detect the trash articles we compare the predefined category to the categories that the article belongs to according to our categorization procedure. For example if an article belongs in two categories as a result of the categorization procedure and neither of these 2 categories are the same as the predefined category then this article is marked as possibly trash. An example is shown in Table 3. The MAX relevance 8,1% associates the article with the category Health. The second max relevance (MAX-1) associates the article with the category Science, and the third max relevance (MAX-2) associates the article with the category Technology. We observe that MAX-1 / MAX = 0.8 > 0.75. From (2) we consider that MAX and MAX-1 are close. On the other hand, MAX-2 / MAX < 0.75. This means that the classifier categorized the article in two categories (Health and Science). Next, we compare the predefined category which can be found in the ‘articles’ table to the two categories in which the classifier categorized the article. The predefined category is Business and it is not the same with neither the category Health nor the category Science. As a result, the article is marked as trash.

Table 3. Relevance of an article to a category and article details

Category	**Relevance**
Health	8,1%
Science	6,5%
Technology	5%
Education	1,7%
Entertainment	1,7%
Politics	1,2%
Business	1,1%
Sports	0,3%

Title	**Body**	**RSS category**
Study: Airport Screening Process Pointless	Please Note: You've requested an ABCNews.com page...	Business

It's easy to generalize the trashy detection algorithm for articles that have been categorized in N categories by our classifier.

```
for_each article{
if (MAX-1/MAX < 0.75)
{
        if( pre_category!=category_with_ MAX  mark;
}
else
{
        for (i in 2:n)
        {
                    if ( MAX-(i-1)/MAX to MAX-1/MAX > 0.75 AND MAX-(i) / MAX < 0.75)
                    {
                                if (pre_category! = category _with_MAX-(i-1) to category_with_MAX-1)
        mark;
                    }
        }
}
```

Because of the types of the categories our system supports, a legitimate article can belong to at most 3 categories. Therefore, for our system, the maximum value of N in our system is 3.

The second heuristic of trash detection marks an article as trash if it has been categorized to more than three categories. Thus an article is marked as trash when the following conditions are in effect:

$$MAX\text{-}1 / MAX > 0.75$$
$$MAX\text{-}2 / MAX > 0.75$$
$$MAX\text{-}3 / MAX > 0.75$$

This means that a legitimate cannot belong to four or more than four categories altogether. An example of an article belonging to this category is shown in Table 4. The categorization system associated the article with four categories because MAX-1/MAX, MAX-2/MAX, MAX-3/MAX are all greater than 0.75. From the assumption we have made, that only trashy articles can belong to four categories simultaneously, we mark this specific article as possibly trash.

Table 4. Relevance of an article to categories and article information

Category	Relevance
Health	6,6%
Science	5,9%
Sports	5,4%
Technology	5%
Politics	4,9%
Entertainment	4,1%
Business	3%
Education	2,9%

Title	Body	RSS category
Darwin's Creatures	Darwin’s Creatures (13 pictures). (13 pictures). 1 / 13.Beetles were Darwin’s first passion as a naturalist.	Business

5. EXPERIMANTAL EVALUATION

In this section we evaluate the trash article detection mechanism on our system PeRSSonal (Bouras C., et al., 2008). PeRSSonal is a complete system, able to gather news from major news portals, categorize and summarize them, and finally syndicate them personalized to the end users. The system is based on algorithms, which incorporate the user into the categorization and summarization procedure of news articles, while the results are presented to the user according to his/her interests and end device.

To evaluate the trash detection mechanism we calculated precision and recall for both the heuristics of trash detection. In a statistical classification task, the precision for a class is the number of true positives (i.e. the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class (i.e. the sum of true positives and false positives, which are items incorrectly labeled as belonging to the class). Recall is defined as the number of true positives divided by the total number of elements that actually belong to the positive class (i.e. the sum of true positives and false negatives, which are items which were not labeled as belonging to the positive class but should have been). About 10000 articles were used as our experimental set. From them, 1700 were marked as trash articles by experts. 1000 of these 1700 trash articles should be detected by the first heuristic. This means that the categories the expert classified each one of the articles were different than the predefined category of each article. The other 700 trashy articles should be detected by the second heuristic (the expert could not decide in which category to classify the article). We ran the first heuristic on the experimental set and it marked 992 articles as trash, from which the 90 were legitimate articles (false detections). The second heuristic returned 689 articles from which 44 were legitimate articles (false detections). The precision and the recall, for the first 100, 200, 300 etc articles returned by the first and the second heuristic are depicted in figure 2 and figure 3 respectively.

Figure 2. Precision and Recall of the first heuristic

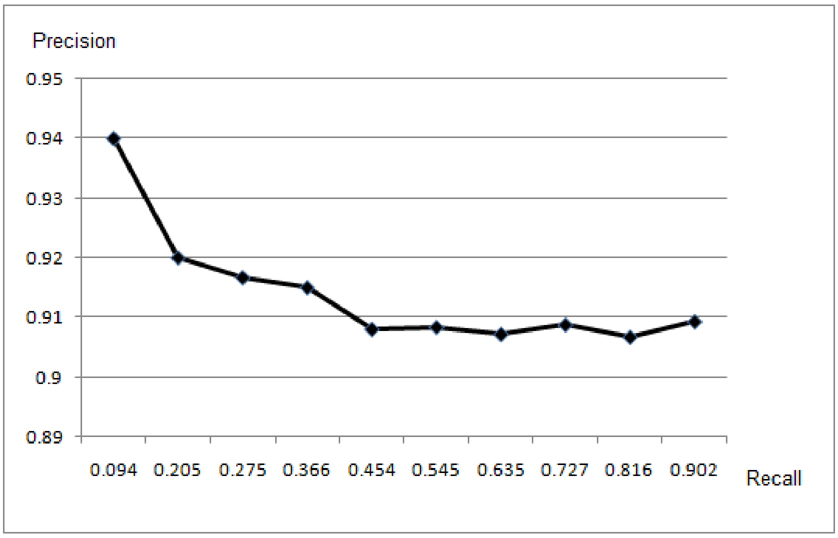
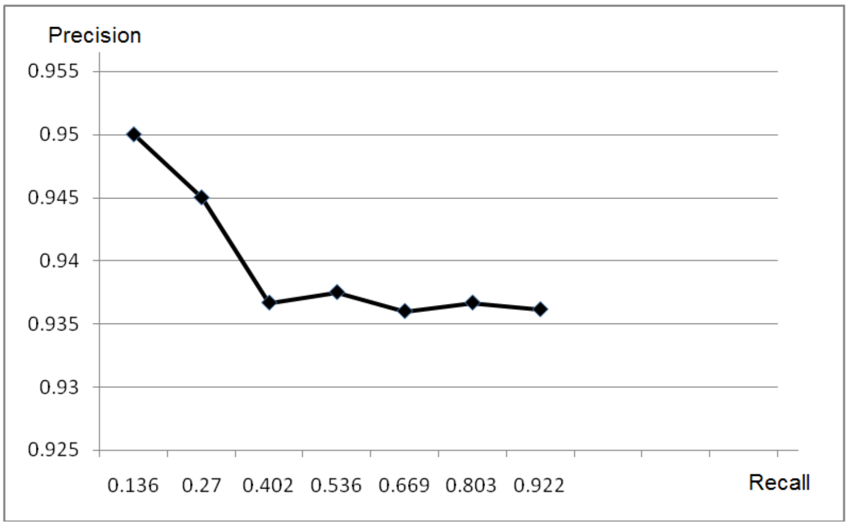


Figure 3. Precision and Recall of the second heuristic



From the figures 2 and 3 we observe that the precision and the recall of both of the proposed heuristics is greater than 90%. To compute the total accuracy of the proposed algorithm we used the F score, which is the harmonic mean of precision and recall.

$$F = 2 * (precision * recall) / (precision + recall)$$

The f score of the proposed algorithm, considering both heuristics is 91.5%.

6. CONCLUSION AND FUTURE WORK

The aforementioned algorithm, in order to detect the trashy articles, used the pre-category of the article and the relevant categories of the article in which the categorization mechanism associates the article.

From the articles that the proposed mechanism returned, the vast majority were trashy articles. The few exceptions were the articles that contained "useful information" but were marked as trashy. It was these articles that the categorization mechanism failed to classify correctly for a variety of reasons (poorly categorized articles). The main reason that these articles are difficult to be categorized is that they suppose that the reader has a previous knowledge of the topic. As a result these articles do not contain terms that could help the system to classify the article correctly into a category. The bag of words (BOW) approach used for the training of the categorization system is inherently limited, as it can only use pieces of information that are explicitly mentioned in the documents, and even that provided the same vocabulary is consistently used. Specifically, this approach has no access to the general knowledge possessed by humans, and is easily distorted by facts and terms not mentioned in the training set.

Generally for the future we would like to enhance our categorization mechanism by using encyclopedic knowledge and other semantic features to augment the classification accuracy of our categorization algorithm. This change can further enhance the accuracy of the trash detection algorithm because as we mentioned a more accurate categorization leads to less false trashy articles detections.

REFERENCES

Adam, G., et al, 2009. CUTER: An Efficient Useful Text Extraction Mechanism, *The 2009 IEEE International Symposium on Mining and Web*, Bradford.

Bai J., Nie J.-Y., Cao, G., 2005, Integrating Compound Terms in Bayesian Text Classification, *Web Intelligence*, pp. 598 – 601.

Banerjee S., et al., 2007, Clustering Short Texts using Wikipedia, *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 787 – 788.

Bloehdorn S., Hotho A., 2004, Text classification by boosting weak learners based on terms and concepts, *In Proceedings of the 4th IEEE International Conference on Data Mining (ICDM)*, pp. 331–334.

Bloehdorn S., Hotho A., 2004, Boosting for Text Classification with Semantic Features, *In Proceedings of the MSW 2004 Workshop at the 10th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 70-87.

Bouras, C., et al, 1992. PeRSSonal's core functionality evaluation: Enhancing text labeling through personalized summaries, *Data and Knowledge Engineering Journal*, Elsevier Science, Vol. 64, Issue 1, pp. 330 – 345.

Crammer, K., and Singer, Y., 2001. On the algorithmic implementation of multi-class kernel-based vector machines. *Machine Learning Research*, Vol. 2, pp. 265-292.

Forman G., 2004, A pitfall and solution in multi-class feature selection for text classification, *In Proceedings of 21st International Conference on Machine Learning*, pp. 107–114.

Gabrilovich, E., and Markovitch, S., 2006. Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopaedic Knowledge, *Proceedings of the 21st national conference on artificial intelligence*, pp. 1301-1306.

Gabrilovich E., and Markovitch S., 2005. Feature Generation for Text Categorization Using World Knowledge, *International joint conference on artificial intelligence*, pp. 1048-1053.

Gabrilovich E., Markovitch S., 2004. Text categorization with many redundant features: Using aggressive feature selection to make SVMs competitive with C4.5, *In ICML'04*, pp. 321–328.

Kim H., et al, 2005. Dimension reduction in text classification with support vector machines, *Journal of Machine Learning Research 6*, pp. 37–53.

Moschitti A., and Basili R., 2004. Complex linguistic features for text classification: A comprehensive study, *Proceedings of the 26th European Conference on Information Retrieval (ECIR)*, pp. 181-196, 2004.

Peng F., et al., 2004. Augmenting Naïve Bayes Classifiers with Statistical Languages Models, *Journal of Information Retrieval*, vol. 7, pp. 317-345, Kluwer Academic Publishers, 2004.

Schneider K., 2005. Techniques for Improving the Performance of Naive Bayes for Text Classification, *LNCS,* Vol. 3406, pp: 682-693.