Towards a Semantic Classifier Committee based on Rocchio

Shereen ALBITAR\textsuperscript{a,1}, Sébastien Fournier\textsuperscript{a} and Bernard ESPINASSE\textsuperscript{a}
\textsuperscript{a}Laboratory of systems and information science (LSIS)
Domaine Universitaire de St Jérôme Avenue de l’Escadrille Normandie-Niemen 13397 Marseille cedex 20
\{shereen.albitar,bernard.espinasse,sebastien.fournier\}@lsis.org

Abstract. This paper concerns supervised classification of text. Rocchio, the method we choose for its efficiency and extensibility, is tested on three reference corpora “20NewsGroups”, “OHSUMED” and “Reuters”, using several similarity measures. Analyzing statistical results, many limitations are identified and discussed. In order to overcome these limitations, this paper presents two main solutions: first constituting Rocchio-based classifier committees, and then using semantic resources (ontologies) in order to take meaning into consideration during text classification. These two approaches can be combined in a Rocchio-based semantic classifier committee.

Keywords. Text Classification, Semantic classification, Information retrieval, Rocchio, Similarity measures, conceptualization

Introduction

Nowadays and due to the explosive increase in published information on the Web, existing search engines seem to be unable to respond efficiently to user requests. This is often related to the traditional keyword-based indexing techniques neglecting search context \cite{1}. Aiming at more efficient and less time expensive search on the Web, it seems adequate to involve classification techniques in order to consider the contents of search engines answers applying thorough filtering and ranking. Web page classification is currently a challenging research topic, particularly in areas such as information retrieval, recommendation, personalization, user profiles etc.

Comparing their heterogeneous structure with plain text documents, Web page classification can be considered a particular case of text classification as many features can be extracted from different parts of a Web page's HTML code (title, metadata, header, URL, …) in addition to its contents \cite{2, 3}. Despite these differences, principles of plain text classification can also apply to Web page classification as well. Adapting text representation models to Web pages by integrating additional features is a promising option. Moreover, traditional text classification techniques might be applied to textual contexts extracted from Web pages \cite{4}.

\textsuperscript{1} Corresponding Author.
Most popular text classification methods are: Naïve Bayes Classifier (NB), Support Vector Machines (SVMs), Rocchio, and K Nearest Neighbor (KNN).

NB classifier [5], also called "The Binary Independence Model", is based on the independence hypothesis considering each feature independently in calculating class prototype during training phase. This unrealistic hypothesis, despite its simplicity, has critical weaknesses [6]. SVMs [7-9] are efficient methods for classification, nevertheless their learning complexity is high. This complexity is related to the number of features used in characterizing documents, so feature selection is indispensible for eliminating noisy and irrelevant features [10]. KNN [11] is also sensitive to noisy examples in training set. In addition, its classification is very slow when using large corpus [10].

Concerning Rocchio, or centroïd-based classifier [12], learned centroïd vectors of classes during training represent a classification model that summarizes the characteristics that occur in training documents. This summarization is relatively absent in other classification methods except for NB that summarizes terms occurrences in different classes in the learned term-probability distribution functions. Moreover, Rocchio takes class summarization into account in classification as each test document is compared to classes' centroïds using similarity measures. NB uses also learned probability distribution during classification to estimate the probability of the occurrence of each term independently neglecting all term co-occurrences.

Vector-based (binary or TF/IDF) representation used by preceding methods permits Semantic integration or "Conceptualization" that enriches document representation model using a certain background knowledge base [13]. In addition, both KNN and Rocchio enable involving knowledge bases in decision making through semantic similarity functions [14].

In this work, we consider Rocchio an adequate baseline text classifier for its efficiency and simplicity in addition to its extendibility with semantic resources at both levels: text representation and similarity calculation. Most of other traditional classification methods, such as SVM and NB, allow the integration of semantics essentially in text representation. Nevertheless, deploying semantic similarity functions allows a full exploitation of semantic resources (concept properties, relations between concepts, etc.) during decision making.

Moreover, varying Rocchio's characteristics results in varied performances, so constituting classifier committees based on different Rocchio-based variations might help in overcoming their limitations and in combining their advantages resulting in effective classification. Other classifier methods can also be integrated into the committee as well [15, 16].

Next section, presents experimentations realized using Rocchio method with several similarity measures on three corpora: "20NewsGroups", "OHSUMED" and "Reuters". Analyzing statistical results, many limitations are identified and discussed in order to propose appropriate solutions in third and forth section through the use of classifier committees and integrating semantics in classification respectively. Finally, we conclude with an assessment of our work, followed by different research perspectives.
1. Evaluating Rocchio Classifier Using Different Similarity Measures

In previous section, Rocchio classifier has been chosen for its efficiency and semantic extendibility. This section presents an experimental study of Rocchio-based classification of text documents referring to some implementation details. Using five frequently used similarity measures [17] (Cosine, Jaccard, Pearson, Averaged Kullback-Leibler Divergence, and Levenshtein distance) separately in experimentations enables us to evaluate Rocchio's performance independently to similarity calculation in decision making. Afterwards, results of different system configurations applied to three different corpora: 20NewsGroups, OHSUMED, Reuters corpus are compared and analyzed. Finally, we discuss certain limitations in these results intending to overcome them using Rocchio-based classifier committee and/or integrating semantic aspects in classification process.

1.1. Rocchio Implementation details

Rocchio or centroid based classification [12] for text documents is widely used in Information Retrieval tasks, in particular for relevance feedback [18]. In Rocchio, Vector Space Model (VSM) is adopted for document representation through applying four preprocessing steps (Tokenization, Stemming, Stopword Removal and Weighting) without applying any feature selection technique. Multiple weighting schemes might be used to represent the corresponding importance of each term in a document [19] like idf, idf-prob, Odds Ratio, $\chi^2$ etc. According to TF/IDF, the most popular scheme, the score of a term $t_j$ in document $d_i$ is estimated as follows:

$$w_{ij} = \frac{tf_{ij} \cdot \log (N/df_j)}{df_j}$$

1.2. Frequency of term $t_j$ in document $d_i$.

$N$: Number of documents.

$df_j$: Number of documents that contain term $t_j$.

The result of applying vector space model to a text document is a weighted vector of features:

$$d_i = (w_{i1}, w_{i2}, w_{i3}, ..., w_{in})$$

For centroid-based classification, each class is represented by a vector positioned at the center of the sphere delimited by training documents related to this class. This vector is so called the class's centroid as it summarized all features of the class as collected during learning phase. These features result from applying VSM on training documents as detailed earlier. Having $n$ classes in the training corpus, $n$ centroid vectors $\{C1,C2,...,Cn\}$ are calculated throughout the training phase. Considering the class $c_i$, its centroid $c_i = (w_{i1}, ..., w_{in})$ can be calculated using the following equation:

$$w_{ki} = \beta \cdot \frac{w_{k1}}{\|POS_i\|} - \gamma \cdot \frac{w_{kj}}{\|NEG_i\|}$$

$w_{ki}$: The weight of term $t_k$ in document $d_j$

POS$_i$, NEG$_i$: Positive and negative examples of the class $c_i$
In this work we use the following parameters ($\beta = 1$, $\gamma = 0$) focusing particularly on positive examples [12].

In order to classify a new document $x$, first we use the TF/IDF weighting scheme to calculate the vector representing this document in the space. Then, resulting vector is compared to all centroids of $n$ candidate classes using a similarity measure. So the class of the document $x$ is the one represented by the most similar centroid.

$$\arg\max_{i=1,2,...,n} (SimFun(x, C_i))$$

(3)

1.2. Experimentations and results

In these experimentations, three corpora are used: (i) 20NewsGroups [20] a collection of 20,000 newsgroups documents almost evenly divided in twenty news classes according to their content topic assigned by their authors, (ii) OHSUMED a subset of the MEDLINE database restricted to the five most frequent classes [21] (iii) Reuters [22] using the ten most frequent classes. Each corpus is divided in Training and Test sets according to their corresponding references, so experimentations are realized in two phases: Training and Test. Training is realized on the training set of each of these corpora and so class centroids are calculated. As for test, on each corpus, five experimentations are executed applying five similarity measures on the test sets (Holdout validation). For most classification tasks, classifier's accuracy [23] exceeded 90%. In order to evaluate system performance we use F1-Measure [23] that gives more information about the errors the classifiers make.

1.2.1. Experimentations on the 20NewsGroups corpus

As illustrated in Figure 1.A, system's performance varies according to the similarity measure used and the treated class. For instance, the class "talk.religion.misc" is large compared to other religious classes. As observed in results, when a classifier makes error classifying a document related to religion, the resulting class is generally "talk.religion.misc" (False negative). This explains the relatively low value of F-measure ranging between [0.5, 0.57] for "talk.religion.misc". This problem is related to large classes. Classes related to computers seem to use similar vocabulary so the classifier cannot distinguish them properly having similar centroids (similar class issue) and so resulting in values ranging from 0.5 to 0.8 in best cases. Nevertheless, all classifiers perform well treating distinct classes like "rec.sport.hockey", "rec.sport.baseball" resulting in values that exceed (0.9).

After analyzing results in details, it is observed that at least (50%) of incorrectly classified documents (False Negative) are classified in a similar class. Indeed, similar classes, using similar vocabularies, usually have their centroids close to each other in the feature space. This implies some classification difficulties in order to distinguish classes' boundaries affecting overall performance. In addition, document contents might be related to multiple classes making classifier's task tricky.

1.2.2. Experimentations on the OHSUMED corpus

Experimentation results are illustrated in Figure 1.B. In this case, we can also observe performance variations among different similarity measures especially for "C23" where
pathology documents seem to be difficult to distinguish from other classes. In fact, this class is very large compared to others treated in the same case, and in other words, its documents can be related to other classes as pathologies can affect the digestive and the cardiovascular systems ("C06", "C14" respectively). As a result, low recall and F-measure values were observed for this class (≈ 0.5).

![Figure 1. Evaluating five similarity measures on three corpora (F1-Measure)](image)

1.2.3. Experimentations on the Reuters corpus

As already seen in previous experimentations, our classifier shows some difficulties in classifying the general class "grain" as it contains information about both "corn" and "wheat" resulting in low F-measure (<0.3) as illustrated in Figure 1. C. One can also observe similarities among classes like "trade" and "ship" that limit the F-measure to the maximum value of (0.8) whereas for more distinct classes the system attained (0.9) (example "earn" and "acq").

1.2.4. Conclusion

Throughout previous experimentations, some limitations seem to affect Rocchio's performance particularly in dealing with similarities among classes, general classes and heterogeneous classes. These limitations are mainly related to class representation and similarity calculations. Promising ameliorations can be obtained combining the advantages of different classifiers and minimizing their deficiencies through classifier committees. Limitations observed with similar classes may be overcome by means of
semantic resources. Indeed, redefining centroids using concepts instead of terms might limit intersections between spheres of similar classes in concept space. Next sections present in details both solutions that can be combined eventually.

2. Towards A Rocchio-Based Committees for Text Classification

In this work we witnessed variable performance of different system configurations using each of the five similarity measures. As illustrated in Figure 1. A, Kullback-Leibler has the best performance on most classes when tested on 20NewsGroups corpus. Nevertheless, this isn't the case for both tests on OHSUMED and Reuters. Indeed, Cosine and Pearson outperform Kullback-Leibler in most cases when tested on OHSUMED corpus (Figure 1. B). As for tests on Reuters corpus, system performance varies when testing the five similarity measures on different corpus classes as shown in Figure 1. C. This comparison is supported by Macro Averaged statistics obtained from the same previous test as illustrated in Figure 2. For example, system using Kullback-Leibler similarity measure seem to outperform all others in precision, recall, and F-measure considering 20NewsGroups while it seems to be less precise but more efficient in retrieving relevant documents of OHSUMED and Reuters corpora.

![Figure 2.](image)

As the performance of the five system configurations varies, they might be appropriate candidates for constituting classifier committee. "Classifier committees are based on the idea that, given a task that requires expert knowledge to perform, k experts may be better than one if their individual judgments are appropriately combined" [22]. In the case of text classification, k different classifiers used together as an ensemble might outperform the best individual classifier.

A classifier committee is usually composed of, first, k highly independent classifiers who might differ in document representation model (and/or) classification
method, and second, a combination strategy permitting the correct aggregation of their results. In this work, our five classifiers differ in the similarity measure used as the decision criteria for document classification; this makes them differ in results. Aggregating the results of the preceding experimentations, TABLE I. illustrates the percentage of documents correctly classified by N classifiers where $N \in \{0, 1, 2, 3, 4, 5\}$ considering three corpora according to the detailed results of previously presented experimentations. For $N=5$, classifiers using 5 different similarity measures seem to agree on classifying 63%, 41% and 63% of total documents of 20NewsGroups, OHSUMED and Reuters respectively.

As for the combination strategy, many options have been proposed in the literature [22]. The simplest strategy is Majority Voting (MV) where the decision of the majority of k classifier is chosen. This solution covers cases where $N \in \{3, 4, 5\}$, resulting in ameliorating precedent percentages as illustrated in TABLE I., where the precedent percentages become (75.60%, 53.87%, 82.24%) respectively. Other combination strategies can also apply to classifier committees permitting a maximized gain so the total percentage of correctly classified documents is estimated to (84.89%, 72.58%, 90.71%) respectively. Compared to the same percentages for the best performing classifier in each case, classifier committee seems promising. Besides, classifier committee is adaptable to new contexts while members’ performance might change radically.

Table 1. Percentage of correctly classified documents by N classifier and of expected ameliorations

<table>
<thead>
<tr>
<th>Percentage of documents/ Corpus</th>
<th>20NewsGroups</th>
<th>OHSUMED</th>
<th>Reuters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classified by 0 Classifier</td>
<td>15.11%</td>
<td>27.42%</td>
<td>9.29%</td>
</tr>
<tr>
<td>Classified by 1 Classifiers</td>
<td>5.26%</td>
<td>3.27%</td>
<td>4.20%</td>
</tr>
<tr>
<td>Classified by 2 Classifiers</td>
<td>4.04%</td>
<td>15.44%</td>
<td>4.27%</td>
</tr>
<tr>
<td>Classified by 3 Classifiers</td>
<td>6.28%</td>
<td>9.11%</td>
<td>11.55%</td>
</tr>
<tr>
<td>Classified by 4 Classifiers</td>
<td>6.89%</td>
<td>3.54%</td>
<td>7.71%</td>
</tr>
<tr>
<td>Classified by 5 Classifiers</td>
<td>62.43%</td>
<td>41.23%</td>
<td>62.97%</td>
</tr>
</tbody>
</table>

Using Majority Voting $N=\{3,4,5\}$ Gain  13%  13%  19.23%
Estimated Percentage  75.60%  53.87%  82.24%

Using Committees Strategies $N=\{1,5\}$ Gain  22%  31%  27.74%
Estimated Percentage  84.89%  72.58%  90.71%

Sebastiani surveys in [22] three rules for combinations: (i) Weighted Linear Combination (WLC) where the sum of classifiers’ results is calculated using importance weights related to their efficiency (ii) Dynamic Classifier Selection (DCS) where the best classifier is chosen according to the treated document using a validation set of similar documents (iii) Adaptive Classifier Combination (ACC) that combines precedent strategies.

However, a considerable number of documents is classified by none of the five system configurations requiring the involvement of other classifiers (like NB or SVM) in our experimentations in order to cover them. Indeed, this is the major advantage of developing classifier committees; combining the potentialities of different classifiers in the same system.
3. Towards a Rocchio-Based Semantic Classification of Text

In section 2, we encountered in results some limitations related to similar classes in the corpus. These limitations can be surpassed by improving document and class representation model. Indeed, VSM suffers from certain limitations [24, 25] especially for processing composed words, synonyms, polysemy, etc. Deploying semantic resources (like thesaurus & ontologies) transforms term-based document representation into concept-based one producing conceptualized vectors in order to realize "Semantic Classification". In this section we present first different strategies for vector conceptualization. Then, we discuss classification process using conceptualized vectors. These steps constitute a semantic extension to the original Rocchio, thanks to semantic resources.

Using semantic resources, literally occurring terms detected in text can be mapped to semantically related concepts taking their senses into consideration for better classification. WordNet, Wikipedia and other domain specific resources usually called domain ontologies can be deployed in order to realize this "Conceptualization".

Next step is the integration of mapped concepts into document representation to constitute a "Conceptualized Vector". Three possible strategies for concept integration [13] : (i) Adding Concepts: where the original vector is extended and corresponding concepts are added. (ii) Partial Conceptualization: where terms are substituted by corresponding concepts. Terms having no related concepts are held in the vector (iii) Complete Conceptualization: where terms are substituted by concepts whereas remaining terms are eliminated from the final vector.

Integrated concepts are assigned new scores derived from the frequencies of their related terms. The resulting vectors of the enriched VSM can help in improving text classification [26].

The second strategy seems to be the most appropriate as it removes no term without replacing it with a related concept, so no original feature is removed from the vector (compared to the third one), and no extra feature is added (compared to the first one) resulting in minimized efficiency effects. Yet, the classifier has to be adapted to hybrid (concepts + terms) representation.

Decision-making criteria are then applied to Conceptualized Vectors in order to classify corresponding documents. The previous pure mathematic similarity measures can be used for classifying, neglecting semantic relations between relevant concepts. Moreover, only shared concepts among compared vectors can influence decision-making through their importance scores. Similar or related concepts are rarely taken into consideration though they might be useful in classification. For example, integrating super concepts of matched concepts into the conceptualized vector demonstrate some ameliorations in classification results as multiple levels of ontology are considered [13].

For more advances towards a semantic classification, semantic relations among ontology concepts can be taken into consideration through semantic similarity measures. These measures permit comparing similar concepts so they can contribute to vector comparison beside common ones, [14]. For an overview on semantic similarity measures see [27].
Applying similarity measures to semantic classification depends on adopted conceptualization strategy. "Complete Conceptualization" permits the use of semantic similarity measures. Considering other strategies that produce hybrid (concepts+terms) document representation, a combination of mathematic and semantic similarity measures can be applied to terms and concepts respectively.

4. Conclusion and Perspectives

Due to the explosive growth of the Web, many search engines demonstrate limited performance to meet the needs of internet users. This leads to a challenging need for efficient filtering and ranking techniques. This paper presents some traditional methods for text classification that apply to Web page classification as well.

We choose Rocchio that demonstrates a good performance compared to its relatively minimal complexity. In addition, it can provide relevance feedback on classification results permitting better result understanding and so potential improvements in classification. We used five different similarity measures for testing Rocchio method on three corpora (20newsGroups, OHSUMED, Reuters). Yet, several limitations were identified in classification results. Limitations could be overcome through constituting classifier committees, and also by means of semantic resources taking meaning into consideration in text classification. These extensions seem to be promising.

Thus, we propose in this work a new method for semantic classification of text based on combining different Rocchio-based classifiers. Indeed, semantics can be integrated into a Rocchio classifier through conceptualization and also during decision making through different semantic similarity measures. Document vectorization followed by the conceptualization using a knowledge base, helps to complete VSM approach with semantics. Moreover, semantic similarity measures can also be used implying similar concepts in calculating vector similarities. The next step of our work consists in developing a variation of Rocchio-based semantic classifier committee and then in evaluating it on classification tasks using other Web pages corpus like WebKB, and dmoz ODP.

References