Olex
Effective Rule Learning for Text Categorization

VeronicaLucia Policicchio

Dipartimento di Matematica,
Università della Calabria
87036 *Rende, Italy*
email : policicchio@mat.unical.it

Abstract

Text Categorization is the problem of the automatic categorization (or classification) of texts into pre-specified categories. It dates back to the early 60s, but only in the last ten years it has witnessed a booming interest, both in research area and in applicative contexts. In fact, as the modern information technologies and the web-based services successfully make a great volume of information available, the problem of accessing, selecting and managing this information, usually expressed as textual data, arises. In the research community the dominant approach to this problem is based on the application of machine learning techniques: a general inductive process automatically builds a classifier by learning, from a set of previously classified documents, the characteristics of one or more categories. The advantages of this approach over the knowledge engineering approach (consisting in the manual definition of a classifier by domain experts) are a very good effectiveness, considerable savings in terms of expert manpower, and straightforward portability to different domains.

In the last years, a great number of statistical classification and machine learning methods to automatically construct classifiers using labelled training data have been proposed. The common target to all these systems is the definition of *category profiles*, describing the characteristics that a document must have in order to be associated to one o more categories. The differences among them rely on the techniques used to this task, which vary from decision tree to genetic algorithms, from probabilistic techniques to mathematic and geometrical methods. Among them, rule learning algorithms has become a successful strategy for classifier induction. While weighted solutions such as the linear probabilistic methods or nearest-neighbor methods may also prove reasonable, the models they employ are not explicitly interpretable; rule-based classifiers, instead, provide the desirable property of being readable, easy for people to understand. Several approaches (either rule-based or not) exploiting negative information for text classification can

be found in the literature.

A general formulation of the induction problem (for text categorization) is as follows. Given

- a *background knowledge* B as a set of ground logical facts of the form $t \in d$, meaning that term t appears in document d (other ground predicates may occur in B as well)
- a set of *positive examples* expressed as ground logical facts of the form $d \in c$, meaning that document d belongs t o category c (*ideal classification*); *negative examples* are implicitly stated according to the Closed World Assumption (i.e., if $d \in c$ is not a positive example, then it is a negative one)

constructs a hypothesis $\mathcal{H}_c$ (the classifier of c) that, combined with the background knowledge B, is (possibly) consistent with all positive and negative examples, i.e., $B \wedge \mathcal{H}_c \models P$ and $B \wedge \mathcal{H}_c \not\models N$. The induced rules will allow prediction about the belonging of a document to a category on the basis of the presence or absence of some terms in that document.
The above induction problem is essentially an instance of Inductive Logic Programming (ILP), which deals with the general problem of inducing logic programs from examples in the presence of background knowledge. It is well known that ILP problems are computationally intractable, so that a main topic is that of identifying classes of programs that are efficiently learnable. The theory of *PAC-learnability* provides a model of *approximated* polynomial learning where the polynomially bounded amount of resources (both number of examples and computational time) is traded-off against the accuracy of the induced hypothesis. . We note that, while in ILP it is assumed that the input sample is consistent with some hypothesis in the hypothesis space, in TC this is not necessarily true; indeed, it is not possible, in general, to correctly categorize a document under a category only on the basis of the terms occurring in it (for instance, due to the presence of homonyms). For this reason, the expected induced hypothesis is in general one which *maximally* satisfies (both positive and negative) examples.

In this thesis we propose Olex, a novel method for the automatic induction of rule-based text classifiers. In Olex, the learning problem is stated as an optimization problem relying on the F-measure as the objective function. In particular, the optimization task is that of determining a best set $X_c = \{T_1^+, \cdots, T_n^+, T_{n+1}^-, \cdots T_{n+m}^-\}$ of *discriminating* terms (d-terms) for c. A d-term is of the form

T^s, where T is a conjunction $t_1 \wedge \cdots \wedge t_k$ of (simple) terms and $s \in \{+, -\}$ is the sign. We call T *conjunctive* term (co-term). A d-term T^s *occurs* in a document d if the co-term T *occurs* in d (denoted $T \in d$), i.e., if the term t_i occurs in d, for each $i = 1, k$. A positive d-term occurring in a document d is indicative of membership of d in c, while a negative one is indicative of non-membership. Thus, a document d containing *any* positive d-term in X_c and *none* of the negative d-terms in X_c, is *eligible* for classification under c according to X_c. Hence, the aim of the optimization task is that of finding a set X_c of d-terms such that, by classifying under c the set $E(X_c)$ of documents of the training set TS eligible for classification according to X_c, the resulting F-measure is maximum (intuitively, this corresponds to finding X_c such that $E(X_c)$ best "fits" the positive examples of c in TS; at most, $E(X_c)$ perfectly matches those examples, in which case the F-measure is 1).

Now, given a (best) set $X_c = \{T_1^+, \cdots, T_n^+, T_{n+1}^- \cdots T_{n+m}^-\}$ of d-terms, the corresponding hypothesis $\mathcal{H}_c$ is of the form

$$c \leftarrow T_1 \in d, \neg\,(T_{n+1} \in d), \cdots, \neg\,(T_{n+m} \in d)$$

$$\ldots$$

$$c \leftarrow T_n \in d, \neg\,(T_{n+1} \in d), \cdots, \neg\,(T_{n+m} \in d)$$

and states the condition "if any of the co-terms $T_1, \cdots, T_n$ occurs in d and *none* of the co-terms $T_{n+1}, \cdots, T_{n+m}$ occurs in d then classify d under category c" (notice that a co-term $T = t_1 \wedge \cdots \wedge t_k$ *does not* occur in d if any t_i, $1 \leq i \leq k$, does not occur in d). That is, the occurrence of a co-term T_i, $1 \leq i \leq n$, in a document d requires the contextual *absence* of the (possibly empty) set of co-terms $T_{n+1}, \cdots,\ T_{n+m}$ in order for d be classified under c[1]. Notice that there is one rule for each positive d-term in X_c and, for each rule, one negative literal for each negative d-term in X_c (thus, all rules share the same negative part $\neg(T_{n+1} \in d), \cdots, \neg(T_{n+m} \in d)$).

Quite obviously, if there exists a set X_c of discriminating terms such that the corresponding set $E(X_c)$ of eligible documents exactly coincides with the positive examples of c, then the induced classifier $\mathcal{H}_c$ is consistent with the training data (i.e., it allows to infer all and only the positive examples of c).

We point out that the above optimization task, and thus the task of inducing a classifier, is computationally intractable. As we will see , it requires exponential time

[1] In general, d may "satisfy" more classifiers, so that it may be assigned to multiple categories

and, even restricting to rules with just simple terms, the problem remains NP-hard. This should not be surprising, given that the Olex's hypothesis language, essentially Horn clauses extended by negative conjunctions of terms, is not PAC-learnable . To cope with the complexity of our problem, we devised a greedy heuristic approach.
Since the set X_c which maximizes the objective function depends on the choice of the vocabulary (i.e., the set of terms selected for rule induction), to pick the "best" classifier Olex proceeds by repeatedly running the optimization algorithm with different input vocabularies, and eventually selecting the classifier with the best performance.

Olex's hypothesis language is original and, as shown by the experimental results, very effective in producing accurate and compact classifiers. Experiments, carried out on two standard benchmark data collections, namely, REUTERS-21578 and OHSUMED, confirm the expectations on our model. In fact,Olex achieves very good performance on both data collections, among the best reported in the literature. In particular, Olex showed to outperform traditional classifiers such as k-NN, Naive Bayesian, C4.5, Ripper, etc., and to be competitive with SVM. Further, unlike SVM, that lacks interpretability, Olex yields classification models that can be easily read, understood and modified by humans. The induced classifiers are indeed very compact: on the top ten categories of the REUTERS-21578, the number of rules in a classifier ranges between 2 and 34.
High performance and compactness of classifiers are consequence of highly effective rules; intuitively, the paradigm "one positive literal, more negative literals" allows rules to catch most of the right documents (through the positive literal), while not making "too many" mistakes (thanks to the negative ones).
Unlike other rule learning systems, Olex is based on very simple and straightforward ideas and, thus, provides a clear intuition of what learning is about. Further, it is formally well-defined and understood.
Further, Olex enjoys a number of further desirable properties:

- it is accurate even for relatively small categories (i.e., it is not biased towards majority classes);
- it can learn from small vocabularies;
- it is robust, i.e., shows a similar behavior on both data sets we have experimented.

Further, thanks to its rule-based approach, the implemented prototype allows an immediate and sound integration of background knowledge. The usefulness of domain-specific knowledge has been evaluated on two data sets belonging to an American insurance agency, by performing a *Semantic Analysis* task, whereby documents have been represented in terms of the extracted concepts. This first empirical evaluation showed that knowledge-based feature generation does not substantially contribute to improve learning of text classification rules. This is a partial result; further investigation have to be carried out, in order to state whether this result can be generalized for our learning approach or some contribution is obtained when using more appropriate thesauri.

Lastly, the system supports the integration of a manual approach into the automatic categorization. Thanks to the interpretability of the produced classifiers, indeed, the Knowledge Engineer can participate in the construction of a classifier, by manually specifying a set of rules to be used in conjunction with those automatically learned. Experimental results showed that this cooperation may bring Text Categorization to an higher performance level.

In this thesis, after having described Text Categorization problem and discussed some interesting related works, we introduce our learning approach. More specifically, this thesis is organized as follows:

- In Part I, we formally define Text Categorization and its various subcases and review the most important tasks to which Text Categorization has been applied; eventually, we discuss the performance measures classically used to evaluate the efficacy of a classifier and describe the benchmark corpora used to test our system.

- In Part II, we give a survey of the state-of-the-art in Text Categorization, describing some of the algorithms that have been proposed and evaluated in the past.

- In Part III, after providing an overview of Olex and giving some preliminary definitions and notation, we state the optimization problem of selecting a best set of discriminating terms (which is the heart of our method) and prove that this task is computationally difficult. Thus, we propose a heuristic approach to solve it and give a description of the whole learning process. Then, we present the experimental results and provide a performance comparison with other learning approaches. Finally, in the light of the obtained results, we provide a discussion.