Book Reviews

Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms

Thorsten Joachims
(Cornell University)

Dordrecht: Kluwer Academic
Publishers, 2002, xvii+205 pp;
hardbound, ISBN 0-7923-7679-X,
$110.00, €121.00, £77.00

Reviewed by
Roberto Basili
University of Rome Tor Vergata

1. Introduction

Those trying to make sense of the notion of textual content and semantics within the wild, wild world of information retrieval, categorization, and filtering have to deal often with an overwhelming sea of problems. The really strange story is that most of them (myself included) still believe that developing a linguistically principled approach to text categorization is an interesting research problem.

This will also emerge in the discussion of the book that is the focus of this review. *Learning to Classify Texts Using Support Vector Machines* by Thorsten Joachims proposes a theory for automatic learning of text categorization models that has been repeatedly shown to be very successful. At the same time, the approach proposed is based on a rather rough linguistic generalization of (what apparently is) a language-dependent task: topic text classification (TC). The result is twofold: on the one hand, a learning theory, based on statistical learnability principles and results, that avoids the limitations of the strong empiricism typical of most text classification research; and on the other hand, the application of a naive linguistic model, the **bag-of-words** representation, to linguistic objects (i.e., the documents) that still achieves impressive accuracy.

2. Several Good Reasons for Reading the Book

When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind: it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the stage of science.
—Lord Kelvin

2.1 Statistical Learning Theory Applied to Text Classification

Joachims's book presents the application of a statistical learning model, **support vector machines** (SVMs) (Vapnik 1995), to the problem of text categorization. As the author emphasizes from the beginning of the book, it affects not only theoretical aspects, and not only empirical findings or implementation ideas, but all these aspects. The book

is the results of the author's Ph.D. thesis, and this is strongly reflected in its overall organization.

The original contribution of the Ph.D. thesis is the specific application of SVMs and some theoretical and algorithmic novelty developed for (but not limited to) text categorization. The author divides the book into four parts: "Notation", "Theory", "Methods," and "Algorithms."

"Notation" introduces the problem and surveys existing approaches to automatic text categorization. First, in chapter 2, feature representation methods are discussed, from bag of words to multiwords or "semantically" justified features. Then, four traditional methods for learning the categorization function—Naive Bayes (Tzeras and Hartmann 1993), Rocchio (Rocchio 1971), *k*-nearest neighbors (Yang 1994), and decision trees (Quinlan 1986)—are presented in the same chapter. In this chapter (that is, very early in the book) a specific notion of performance is defined. It is expressed as a function of the categorization error (i.e., the number of mismatches between the system outcome and the gold standard). Notice how this is a bias to the entire matter, as discussed below. Finally, the basic definitions for SVMs are given (chapter 3).

"Theory" proposes a general model for TC that is based on the distributional properties of (bags of) words. This results in an abstract notion of **target categorization (TCat) concept** the learnability of which via SVMs derives from their formal properties (chapter 4). Moreover, the TCat concept has an inductive nature, as it is based on the availability of a large set of training examples: Under certain assumptions on the training material, the TCats' results are linearly separable by an SVM with a controlled amount of error (or loss in predictive accuracy). Accordingly, chapter 5 analyzes methods for estimating the predictive accuracy of the target SVM. The task knowledge embedded in the training examples provides complete information about the training error, and formal results from statistical learnability theory (e.g., Vapnik 1998) (or newly introduced by using Vapnik's results as inspiration) are directly employed here as an upper bound on the testing error (i.e., the error of the generalization achieved as measured over test data).

"Methods" discusses the core technique for inducing the TC functions by means of SVMs. This is first done in the tradition of inductive approaches to TC: Training examples are used to induce the maximum margin hyperplane that separates positive from negative examples in a binary setting. The empirical evidence—good performance over three well-known benchmarking data sets—confirms the viability of the SVM induction (chapter 6). In chapter 7 the notion of transductive SVMs (Vapnik 1998) is introduced. It is the inductive task that exploits a consistent set of testing examples as a bias for building the maximum margin hyperplane. Such an approach has several analogies with forms of **active learning** (e.g., **co-training** [Blum and Mitchell 1998]), wherein evidence during learning is derived from pieces of more or less weak test evidence (e.g., independent feature spaces are used as selective information on how to sample training examples in co-training).

Algorithms and concrete methods for the application of the previous results are then reported in the fourth part, "Algorithms." In chapters 8 and 9, efficient algorithms for training inductive and transductive SVMs are finally presented, with reference to the software platform *SVMLight*, another (nonsecondary) side effect of the author's Ph.D. thesis.

2.2 Empirically Grounding a Powerful Theory

The book has great merit, as much space is given not only to theoretical and experimental aspects, but to an attempt to empirically assess support vector machines as a general learning theory for TC. Grounding formal results on large-scale data is always

attempted where possible. It is carried out over the target benchmarking corpora: a collection of Reuters news (Reuters-21578), a collection of medical texts (OHSUMED), and a set of manually classified Web pages (the WebKB collection).[1] This evidence is also used to define an abstract model of what a target text categorization concept is and how it is learnable by a SVM. The notion of TCat concept tries to capture exactly this.

These two aspects are very important, as the reader can better understand the large set of (often mathematically complex) theoretical notions against empirical data and also validate progressively the results of the theory against the evidence derived from real collections. For example, interesting sections (especially for a computational linguistics researcher) are those in which mathematical notions, such as linear separability, training error, and the TCat concept itself, are discussed via estimation against the benchmarking data (e.g., section 3 of chapter 4). Although test data cannot be considered exhaustive samples, TC benchmarks are a rather precious source of information about the distributional and linguistic properties of words. Notice that such combined theoretical and empirical analyses are rare in the literature. The result is an attempt to reconcile IR (often too much focused on empirical performance measures) and AI (more often targeted to theories of learning with weaker possibilities for large-scale empirical assessment).

Finally, as large-scale data analysis is required in the study of several NLP tasks (and not only TC), the book is a good example for researchers and practitioners in empirical language processing.

2.3 Theory, Application, and Implementation of Support Vector Machines

Covering aspects related not just to methods, but also to theory and efficient implementation of SVMs is a positive aspect of the book. The book helps in building a rather rich picture of the field. Introductory matter is presented in a comprehensive and theoretically well-founded way. Then the learning theory based on earlier results from Vapnik (1995) is described, and this has a valuable effect on the methodological contribution. All the aspects of the application of SVMs to TC are thus framed in a larger picture.

Examples are the discussion of benchmarking data in section 4 of chapter 5, in which theoretical estimators used to upper-bound the categorization error are compared to their measures as carried out over the benchmarking collections. Almost every specific parameter of the estimators that raises questions about the quality and applicability of the theory (e.g., dimension of the training set, the error embedded in training material) is analyzed comparatively against the benchmarking data. The analytical estimates are thus compared to the effective measures. This is very relevant for anyone interested in empirical analysis of linguistic data.

2.4 Expressiveness versus Efficiency

One of the contributions of Joachims's thesis is the effort spent in making SVM learning for text classification feasible. In SVMs, the induction is based on the solution of a quadratic optimization problem in which the size of the matrix is quadratic in the number of available training examples and the different values depend on the choice of the **kernel** functions.[2] The problems here are related to the iterative evaluation of

1 See details at these sites, respectively: `http://www.research.att.com/lewise/reuters21578.html`, `ftp://medir/ohsu.edu/pub/ohsumed`, `http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data`

2 Kernel functions are used to map the problem from its source feature space to a new feature space where linear separability is ensured. Typical kernel functions K are polynomial, e.g., $K(\vec{x}_1, \vec{x}_2) = (\vec{x}_1 \cdot \vec{x}_2 + 1)^d$.

the matrix, which cannot be completely computed efficiently or stored in memory (for high values of n). The author capitalizes on the idea of working on decomposed matrixes of smaller size, previously suggested by Osuna, Freund, and Girosi (1997).

First, during an iteration of the algorithm, a subset of the variables is fixed so that the current re-evaluation is carried out on the remaining subset of the variables, called the **working set**. The different values in the working set determine a specific bias for the algorithm at each step. The selection of a good working set limits the complexity of each step and increases the convergence speed.

Then, a second optimization criterion is discussed that is related to the use of support vectors (SVs). SVs are the examples closer to the separating hyperplane, such that they are at distance d (the margin) from it. As the hyperplane coefficients depend only on SVs, an algorithm able to work from the beginning on just the SVs would be the most efficient. The selection of SVs is clearly heuristic, as their position within the training data set is unknown at the beginning. The author suggests a selection algorithm that predicts (and then neglects) non-SVs among the training data according to their behavior in the previous h steps. Data left out of the analysis are checked a posteriori. The optimality conditions are checked after a solution for the subproblem has been built. In case an invalid solution has been found, optimization is rerun, although it is reinitialized through the last partial solution found. Although more details would have been beneficial,[3] empirical analysis is then discussed at length. The impacts of the different factors on the learning times are reported as they have been measured against benchmarking data collections (section 7 of chapter 8).

3. Stay Far Away?

Not everything that counts can be counted, and not everything that can be counted counts
—Albert Einstein

3.1 Where Is the Language?

After a few words spent discussing possible text representation formalisms that have been adopted in the TC literature, the author completely neglects choices other than bags of words. This representational issue is secondary throughout the book. The author instead concentrates just on learnability and categorization accuracy achieved with bags of words. According to the unquestionable fact that a text is a linguistic object, a theory related to its classification should at least include an analysis along a linguistic dimension—that is, the study of representations different in terms of linguistic properties. Unfortunately, in the book no empirical analysis of different feature representations is even attempted.

As a result, whenever empirical findings do not fit well with theoretical estimates, weak explanations are given. For example, in Table 4.3 (chapter 4, section 2.3), linear separability (i.e., separability of hyperplane with a linear kernel and with no contribution from slack variables compensating for the noise in training data) is discussed over the Reuters news and the OHSUMED abstracts. The fact that in some cases linear separability is not achieved is quickly explained in terms of "dubious" (invalid?) documents or inconsistent classifications reflecting human errors. Note that the Reuters collection is full of linguistically "dubious" documents (e.g., tables with long lists of numbers and almost no text at all; see, for example, the "earn" category). Moreover,

3 The accuracy of the empirical data made available by the author in this section is lower than in other parts of the book.

these phenomena seem to favor coarse approaches like Rocchio's or SVMs: Measures obtained without any feature selection (thus including most nonlinguistic tokens) suggest nontrivial increases in performances (see, for example, experiments discussed by Basili and Moschitti [2002]). Evidently, the reason for mismatches between empirical evidence and the theory lies in the fact that something is missing from the latter, which in fact is based on a rather rough approximation. Separability seems harder when the space based on bags of words is not expressive enough. This strictly depends on the target class (i.e., the underlying knowledge domain) and on the quality of the available training material. Bags of words provide strong evidence in a large set of TC tasks but do not suitably express the required information in *all* cases. The variability of categorization performances of SVMs across different corpora (e.g., about 0.87 break-even point on Reuters but 0.67 break-even point for OHSUMED) is further evidence that although optimization implied by SVMs is very powerful (i.e., is able to build a very good prediction function whatever the underlying feature space is), there is still something missing. Some other dimensions exist where further improvement should be looked for. Language processing could be employed in all cases in which the gap between the best bag-of-words performance and the expected optimum is not trivial.

3.2 What Is a TCat?

The concept of TCat, discussed in chapter 4, seems to be a complex artifact with no clear relationship to the target task. A TCat is fully defined in terms of the distributional properties of (a set of) words that represent it. Note that these properties are exactly the ones requested by the statistical learnability criteria for SVMs. A TCat is defined as a set of word classes determined by their common average frequency figures in the training data. Such a representation seems determined by the research goal (i.e., statistical learnability), and no other strong evidence is brought forward except for frequency figures in the benchmarking data. This is in fact an empirical validation of the learnability theory for which bags of words and their distributions in texts are sufficient to induce those SVMs that minimize categorization errors. This only implies, however, that successful learning via SVMs is a validation of the TCat notion. The reverse is not true, as this latter does not represent any theoretical explanation of what learning for text categorization is.

3.3 Which Performances?

Most of the research in information retrieval is targeted to the maximization of utility functions that are operational (e.g., accuracy in categorization or relevance of documents to queries). And this is also the perspective adopted in performance evaluation (Losee 1998) and in building benchmarks. The questions that make sense are still: Is the minimization of categorization error the entire target of text categorization? Is it the only research focus? Is separating hyperplanes the only interesting aspect of the problem, or should more powerful explanations be learned from training data?

This book suggests that classification functions can be successfully built from extensive training data (i.e., manually classified documents). The outcome of the learning framework, however, is just a Boolean function working as a black box. Unfortunately, the author never tries to give an interpretation of the kind of information induced. Are the weights induced by the SVM meaningful in some sense? Are they telling something even outside of the black box? If we are to compare two SVMs, can we rely just on differences in performance? Or should we perhaps look to other aspects to measure their usefulness? My feeling is that very little has been done for evaluating how well SVMs represent the critical aspects of the problems. For example, are the induced coefficients w_i of the hyperplane (i.e., dimensions related to the ith word)

linguistically meaningful elements, directly mappable toward lexical or semantic phenomena? In this perspective, performance measures that are counts or probabilities of classification errors are limited, or even misleading. Research should avoid the temptation of reducing to merely optimization of the precision/recall trade-off. Linguistically justified features have the inherent benefit of supporting natural explanations of the system induction, although evaluation usually does not account for them. In my view, the (linguistic) interpretation of different aspects of SVM learning in TC is an important research direction. The study of the relationship between training materials (as represented under a linguistic perspective) and the choice of kernel functions optimal for the task is a promising research line.

4. Late Reflections

Support vector machines are widely adopted today for several tasks (e.g., Kudo and Matsumoto 2001). They seem to reproduce effectively the induction of separation functions from training data. The SVM inductive setting (of which the transductive one is just a derivation) is a straightforward approach to TC induction, and this book is proof. Although it leaves a large set of open problems, the book must be seen as a relevant contribution in the area of machine learning for natural language. It is a powerful means of getting acquainted with theoretical and methodological knowledge for text classification. Unfortunately, obvious gaps highly affect its completeness as a handbook in courses on machine learning for text classification and NLP, but this was outside of the author's aims.

In current research and practice, the theory proposed in this book and its empirical grounding are important contributions to the area of empirical natural language processing. First, it embodies new ideas about learning that look for applications to new NLP tasks. Classification of NL questions in question answering and pattern acquisition for adaptive information extraction are just two examples in which a transductive SVM approach is currently being applied. Second, the book embodies an approach to empirical research that is very elegant. Solid theoretical inspiration is here systematically mirrored with real data, where motivations for the first are tentatively found in the latter. The impressive methodological and applicative achievements say much about the effectiveness reachable by elegant ways of doing research.

References

Basili, Roberto and Alessandro Moschitti. 2002. Intelligent NLP-driven text classification. *International Journal on Artificial Intelligence Tools*, 11(3): 389–423.

Blum, Avrin and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98)*, pages 92–100. Morgan Kaufmann.

Kudo, Taku and Yuji Matsumoto. 2001. Chunking with support vector machines. In *Proceedings of the Second Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2001)*, Pittsburgh, pages 192–199.

Losee, Robert M. 1998. *Text Retrieval and Filtering: Analytic Models of Performance*. Kluwer Academic, Norwell, MA.

Osuna, Edgar, Robert Freund, and Federico Girosi. 1997. An improved training algorithm for support vector machines. *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, pages 276–285.

Quinlan, J. Ross. 1986. Induction of decision trees. *Machine Learning*, 1:81–106.

Rocchio, J. J. 1971. Relevance feedback in information retrieval. In Gerard Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice-Hall, Englewood Cliffs, NJ, pages 313–323.

Tzeras, Kostas and Stephan Hartmann. 1993. Automatic indexing based on Bayesian inference networks. *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in*

Information Retrieval (SIGIR '93), Pittsburgh, pages 22–34.

Vapnik, Vladimir N. 1995. *The Nature of Statistical Learning Theory*. Springer, New York.

Vapnik, Vladimir N. 1998. *Statistical Learning Theory*. Wiley, Chichester, England.

Yang, Yiming. 1994. Expert network: Effective and efficient learning from human decisions in text categorization and retrieval. *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94)*, Dublin, pages 13–22.

Roberto Basili is an associate professor in the Department of Computer Science, Systems and Production of the University of Rome Tor Vergata, where he has worked since 1990 in the areas of lexical acquisition and machine learning for NLP. His current research interests include methods to develop large-scale and Web-based NLP systems with adequate lexical and ontological resources, as well as to make effective use of NLP techniques within application tasks such as question-answering and information extraction. Basili's address is University of Rome, Tor Vergata, Department of Computer Science, Systems and Production, 00133 Roma, Italy; e-mail: basili@info.uniroma2.it.