Application of Support Vector Machine Algorithm in E-Mail Spam Filtering

Julia Bluszcz, Daria Fitisova, Alexander Hamann, Alexey Trifonov, Advisor: Patrick Jähnichen

Abstract

The problem of spam classification has received broad and current interest in the recent years. One of the most common supervised learning methods applied for binary text classification is Naïve Bayes (NB) algorithm. We propose Support Vector Machine (SVM) spam filtering algorithm, treating NB as a baseline. Taking into account SVMs ability to separate features in high-dimensional space, we aim to prove the hypothesis stating that SVM is more effective than NB regarding the spam filtering problem.

1. Introduction.

The problem of email classification, although broadly recognized since many years, still attracts interest of researchers nowadays. Each webmail service provides its users with spam filter. Our goal is to develop our own classifier which categorizes English-language emails as spam or ham. As far as data is stored in emails in document text format, the problem of introduced binary classification should be treated as a Natural Language Processing problem. One of the most common method applied within the described field is NB [1]. It is a probabilistic classifier, which determines that an instance belongs to a class based on each of feature value probabilities. We have a hypothesis, that SVM, working effectively in a multidimensional space, appears to be a better classifier. Both methods are introduced in detail further.

For the purpose of training and testing of algorithms SpamAssassin training corpus is used. It provides emails in text format with their corresponding categories. With the use of Python programming language we trained SVM algorithm to learn to predict unseen test instances. The evaluation of the results along with the step-wise description of development of the filter are presented in the sections below.

2. Algorithms

As it was mentioned above, the core of the project is application of SVM algorithm. As a matter of evaluation, NB classifier is also implemented. In this section we introduce both in a theoretical manner.

2.1. Support Vector Machine

SVM, as a supervised learning technique, is a conjunction of linear learning machine and kernel function. It creates a hyperplane which divides two classes of variables by maximizing the margin between the hyperplane and the closest data points by assigning a set of weights w to

the feature vector. In the SVM only points that lie close to the decision boundary influence the optimal solution. Thus, the decision function is specified only by a subset of training samples, called the support vectors.

In soft-margin SVM approach (which is applied in this project) some observations are allowed to overlap to wrong classes [2]. Formally, it can be presented in the form of finding an optimal solution to the quadratic problem [3]:

$$max_{\alpha\in\mathbb{R},w\in\mathbb{R}^{\varkappa},\xi_i\geq 0}\gamma - C\sum_{i=1}^{n}\xi_i \quad \text{s.t.} \quad \forall i : y_i(w^T x_i + b) \geq \|w\|\gamma - \xi_i \tag{1}$$

The primal problem can be next transformed into easier in computation dual form:

$$max_{\alpha\in\mathbb{R}^{\aleph}:0\leq\alpha\leq C}\sum_{i=1}^{n}\alpha_i - \frac{1}{2}(\alpha \circ y)^T X^T X(\alpha \circ y) \tag{2}$$

where expression $X^T X$ denotes kernel function.

Using slack variables ξ_i one is able to control if observations can (and if yes - to what extent) be predicted on the wrong side of the margin. It allows us to take under consideration unobserved features which may still influence observations and thus cause noise in data. Large value of parameter C will discourage overlapping of classifications and in extreme cases may lead to overfitting. Small value of C will lead to loosening of our requirements and to the smoother decision boundary.

What makes SVM attractive in e-mail classification problems is its robustness and ability to handle large feature spaces. Since the algorithm is not trying to minimize the error rate, but rather separate the patterns in high dimensional space, the result is that SVMs are quite insensitive to the relative size of each class.

The advantages of SVM methods have been also confirmed by multiple scientific papers, e.g. Joachims (1998), Hidalgo (2002) or Lewis (2002), where they stated that SVM provides better results than Naïve Bayes or k-Nearest Neighbors algorithms in classification problems.

To mention some disadvantages, SVM classifiers could be calculation intensive while training the model. In this regard there are researches, which have stated, that in some cases NB algorithm appears to be more effective, i.e. much faster than SVM, showing the same classification efficiency, e.g Hassan (2012), Matwin (2012).

2.2. Naïve Bayes

Naïve Bayes is a simple, but powerful classifier based on a probabilistic model derived from the Bayes theorem. Basically, it determines the probability that an instance belongs to a class based on each of the feature value probabilities. The naïve term comes from the fact that it assumes that each feature is independent of the rest, that is, the value of a feature has no relation to the value of another feature [1].

Abstractly, NB is a conditional probability model: given a problem instance to be classified to any class $\{C_j\}_{1<j<n}$, represented by a vector of features (independent variables) $x \in \{x_1, \ldots, x_n\}$, it can be formally shown as:

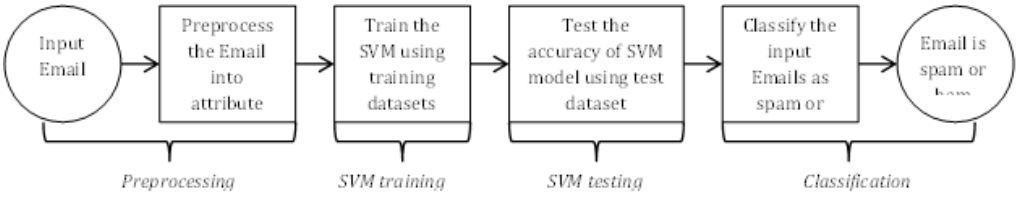
$$C_{MAP} = argmax_{c\in C} P(x_1, \ldots, x_n|c)P(c)$$

$$C_{NB} = argmax_{c\in C} P(c_j)\prod_{x\in X} P(x|c)$$

The crucial benefit of NB algorithm in technical concern is a reduction of the number of parameters, which are needed for modeling. Namely, due to the independence assumption, we need $2n$ parameters to model $P(X|Y)$ instead of original $2(2^n - 1)$ [4]. This quality guarantees simplicity and quickness of the method.

3. Spam filter development and evaluation

We divided spam classification problem into 4 subproblems, treating the first, preprocessing, as crucial. The detailed scheme of working steps from inputs till the classification results is presented below.



As far as initial data is given in the text format, several operations had to be done in the preprocessing step in order to convert emails into feature vectors.

It should be mentioned that the accuracy of classifier depends on the number and relevance of extracted features. In our case the spam filter is based on the following ones: the length of an email, amount of capital letters, non alphanumeric characters, digits, quotations and line breaks in the e-mail body, and finally occurrence of sale- and diet-oriented vocabulary.

As it follows from the presented scheme, we used holdout cross-validation to train and test SVM algorithm: we divided data corpus consisting of 6046 e-mails into training and testing sets, holding 80/20 ratio correspondingly.

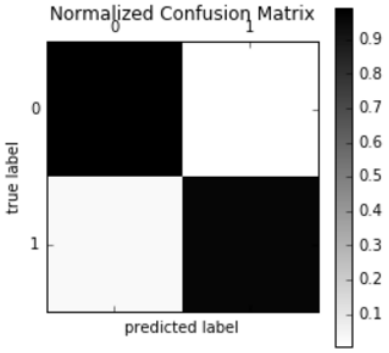
Application of Naive Bayes and SVM algorithms on the SpamAssasin corpus has confirmed our speculations. The SVM method has reached significantly higher precision rates than NB. Moreover, we added ZeroR results as an additional validation of NB and SVM scores. The detailed results of accuracy rates for all the three methods can be seen in the following classification table.

Method	Accuracy rate
ZeroR	68.71%
Naïve Bayes	79.50%
SVM	97.91%

There is no doubt that the results obtained by SVM algorithm yield the highest scores and its classification is almost flawless in contrast to more than 30% incorrectly classified mails using ZeroR and around 10% in case of NB. In the case of SVM application linear kernel turned out to be the powerful enough tool to reach satisfactory results, it proved to be much faster and as effective as gaussian and polynomial kernels. The parameter C reached optimal value of 1, permitting for small overlapping of the data points in hyperplanes. Another useful metric is an overall classification report, which shows in (i, j) cell precision – percent of instances classified as i being truly i, recall – percent of instances i being correctly classified, f-1 score which is counted with the use of formula $\frac{2*precision*recall}{precision+recall}$ and support – the number of instances of each class we have in testing set.The classification report for SVM is presented in the table below.

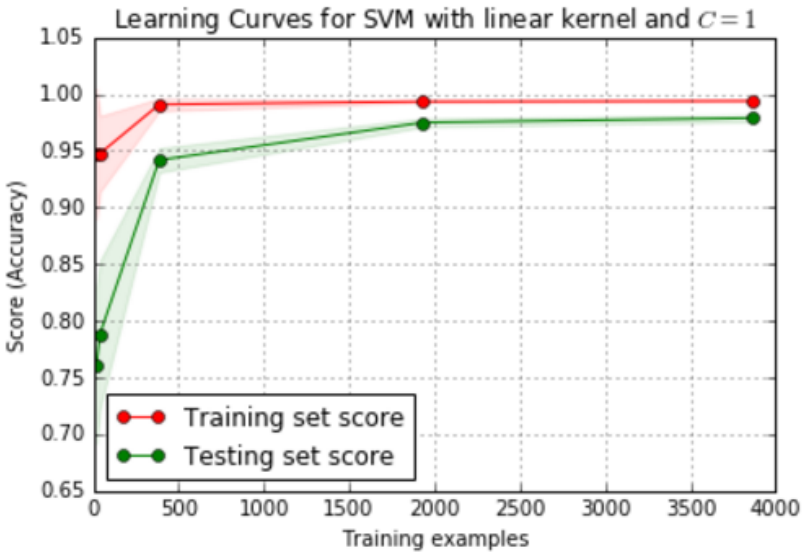
	Precision	Recall	F-1 score	Support
Spam	0,99	0,99	0,99	817
Ham	0,99	0,97	0,98	393
Average/Total	0,99	0,99	0,99	1210

In order to evaluate the precision of created filter one should consider confusion matrix (for absolute values) or normalized confusion matrix (ratio representation), which is an informative source of classification accuracy: in its (i, j) cell it shows the number of instances class i predicted to be in class j. Thus, a good classifier accumulates the numbers on the diagonal. In our case:



From the graph it can be concluded that the vast majority of mails was classified correctly, with diagonal squares being almost completely black. The slightly darker square representing falsely classified spam emails (lower left) is the result of higher penalization of classifying ham as spam than vice versa, which is much more harmful for webmail users.

Furthermore, from the graph of the learning curve it can be concluded that applied SVM algorithm reaches the better classification scores, the more samples are provided in the training set. However, our method quite quickly converges to the satisfactory precision rate and due to the method's stability remains on high level throughout the whole time. Moreover, the learning curve tells us that our model is correctly built and that there is neither a problem of overfitting nor underfitting in the model.



4. Conclusion

In conlusion, the purpose of the project was choosing the best e-mail filtering method which correctly distinguished between spam and ham. From the two methods applied, SVM proved to be far more effective than Naive Bayes algorithm, obtaining almost flawless precision rate, namely $99,99\%$ of correctly classified messages. The empirical experiment clearly proves the unquestioned advantages of this method, which are its speed, robustness and efficiency and in effect leads us to the conclusion that SVM may still be used as a state of the art, powerful e-mail filtering technique.

5. References

[1] R. Garetta, G. Moncecchi, Learning scikit-learn: Machine Learning in Python, 2013.
[2] T. R. F. J. Hastie, T., The Elements of Statistical Learning: Data Minig, Inference and Prediction, ????
[3] G. Chechik, G. Heitz, Max-margin Classification of Data with Absent Futures, Journal of Machine Learning Research 9 (2008).
[4] T. Mitchell, Generative and Discriminative Classifiers: Naive Bayes and Logistic Regression. In Mitchell, M. Hill, Machine Learning, 2015.