# A COMPARISON OF DECISION TREE & NAIVE BAYES CLASSIFIERS AS SPAM FILTERING ALGORITHMS

*Federico Cardoni & Emanuele Cappella*

## Description and Motivation of the problem

- Compare and contrast the differences in performance of Naive Bayes and Decision Tree models in binary classification predicting whether an SMS is filtered as spam or not.
- Our main aim is to evaluate if a Decision Tree model with an accurate feature definition can match the performance of a Naive Bayes Classifier in terms of Spam Filtering accuracy.

## Data Preprocessing

*Dataset*

Our dataset comes from UCI "SMS Spam Collection DATASET". The structure of the database is simple: 5572 sms, whose text is recorded, are labelled as spam and non-spam.

*Preprocessing*

— Each SMS went throught a cleaning process:

... Erasing punctuation

... All capital cases have been lowered

... Excessively long words and extremely short words have been removed

… All stopwords have been removed

… Bag of words have been produced for Spam and Non-Spam SMS

… Initial word-clouds have been plotted to identify the most used words for both cases (Figures A and B).

— A table of most-used words in spam texts for features extraction is produced (figure C)

— Considering the most common words, we defined 5 main spam categories: advertisement (e.g. "service"), adult (e.g. ""xxx"), attention (e.g. "urgent"), mobile (e.g. "mins") and winner (e.g. "prize").

This 'spam dictionary' was the basis for the subsequent predictive models. In fact:

— the spam categories represented the Decision Tree nodes;

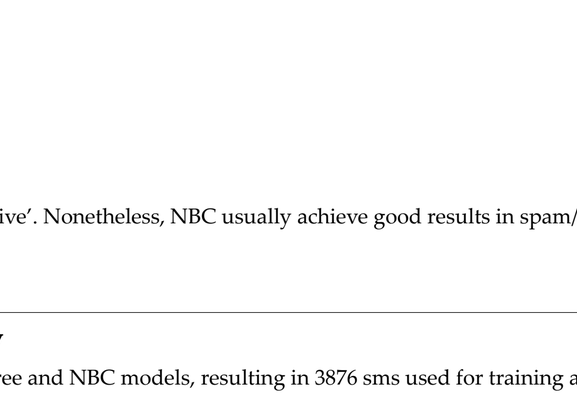
— each message was classified by NBC on the basis of the presence/absence (**Bernoulli document model**) or frequency (**Multinomial model)** of words belonging to the spam categories.


**Figure (A)**: Cloud of most common spam


**Figure (B):** Cloud of non spam words

| Word | N. Occur | % Show | % Cum |
|---|---|---|---|
| txt | 150 | 5.6582 | 5.6582 |
| mobile | 123 | 4.6398 | 10.2980 |
| claim | 113 | 4.2625 | 14.5605 |
| stop | 113 | 4.2625 | 18.8231 |
| reply | 101 | 3.8099 | 22.6330 |
| prize | 92 | 3.4704 | 26.1034 |
| won | 73 | 2.7537 | 28.8570 |

**Figure (C):** Table of Most used spam words




## ADVANTAGES AND DISADVANTAGES OF THE MACHINE LEARNING MODELS

### DECISION TREE CLASSIFIER (DTC)

__ A model that predicts the value of a target variable based on several input variables.

__ DTC is a tree in which each internal (non-leaf) node is labeled with an input feature.

__ Decision tree is a common way to solve spam/ham problems (Tretyakov, 2004).

**PROS (+)**

**(+)** Ability of selecting the most discriminatory features.

**(+)** Comprehensibility so that can be used in Rule Generation problem.

**(+)** Dealing with noisy or incomplete data.

**(+)** Handling both continuous and discrete data.

**CONS (-)**

**(-)** The high classification error rate while training set is small in comparison with the number of classes

**(-)** Exponential calculation growth while problem is getting bigger.

**(-)** Need to discrete data for some particular construction algorithm.

### NAIVE BAYES CLASSIFIERS (NBC)

__Two NBC variations were applied to the present problem: Bernoulli and Multinomial models.

(1) Bernoulli: in this case messages are represented by a binary vector, containing 1 when a certain word is present in it and 0 if not.

(2) Multinomial: the values of the vector represents the words frequency in the message.

The Bernoulli model does not allow to capture multiple occurrences of words in the same spam category, and is usually better employed for shorter texts (Shimodaira, 2015) .

**PROS(+)**

**(+)** Simplicity: easy to understand and implement.

**(+)** No complicated optimisation required.

**CONS(-)**

**(-)** The assumption of class conditional independence is 'naive'. Nonetheless, NBC usually achieve good results in spam/ham classification (Metsis et al., 2006).

## Hypothesis

Our expectations are twofold:

(1) Naive Bayes and Decision Trees will have a similar performance in accuracy (Huang et al., 2006), though we expect a slight advantage for NBC, as they are one of the most used techniques in spam filtering (Metsis et al., 2006). In both cases we hope to achieve an accuracy of by and large 85% cases.

(2) The Multinomial NBC will outperform the Bernoulli one in terms of accuracy (Shimodaira, 2015).

## Train/Test split and Evaluation Methodology

— An 80/20 train/test split was applied for both Decision Tree and NBC models, resulting in 3876 sms used for training and 1696 messages for testing.

— Each model has been trained with hyperparameter optimization.

— The evaluation criteria chosen was to minimize the Classification Error.

— Decision tree is trained minimizing the leaf size.

— Laplace smoothing was applied.

### DECISION TREE CLASSIFIER (DTC)

**Parameters**

Cross-Validation loss of the classifier has been optimized building multiple decision trees and search for the best tree with the smallest possibile leaf size.

**Experimental Results**

— 30 function evaluations has been run, minimun leaf size for the best observed feasible point is 11, while the best estimated feasible point is 14.

— "Adult" predictor has little or none importance.

— "Mobile" has the highest importance and it has been used as root.

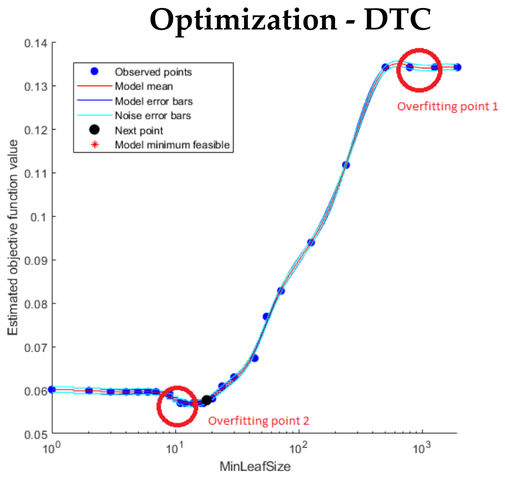
### ACCURACY RATE

| DTC | | BEST NBC |
|---|---|---|
| 94.4% | **Train** | 94.6% |
| 93.1% | **Test** | 93.4% |

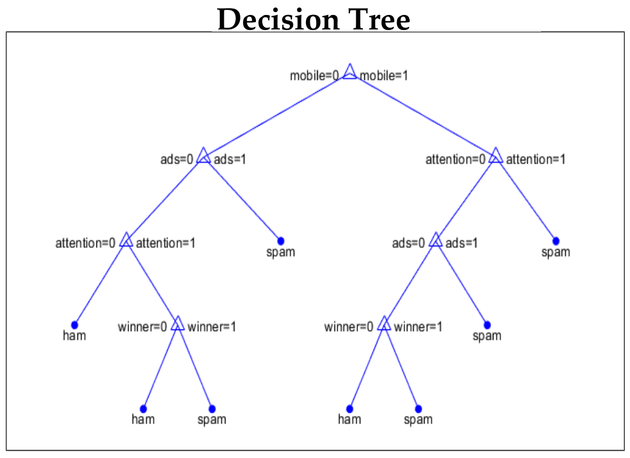
### NAIVE BAYES CLASSIFIERS (NBC)

- Both classifiers relied on the 'spam dictionary' we developed.
- The Multinomial NBC showed higher accuracy than the Bernoulli model in the test data, as expected (93.4% vs 87.9%) .
- The Bernoulli model showed a markedly higher risk of overfitting. compared to the Multinomial one, as visualized thorough the Objective function graph.
- We tested even a second Multinomial model, without our 'spam dictionary'. In this case, the vector features were simply all the words appearing in the sms. Its accuracy was similar to the Bernoulli Model (87% accuracy for test data.

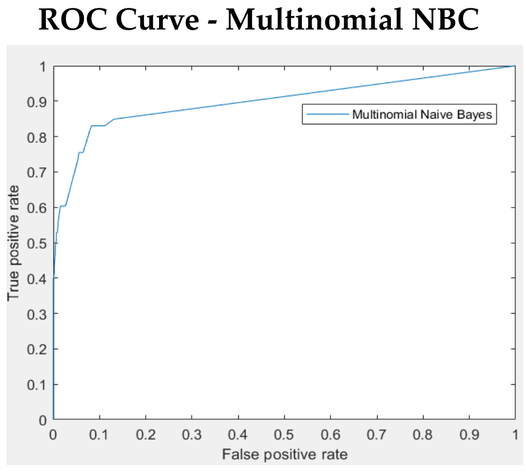
### Optimization - DTC



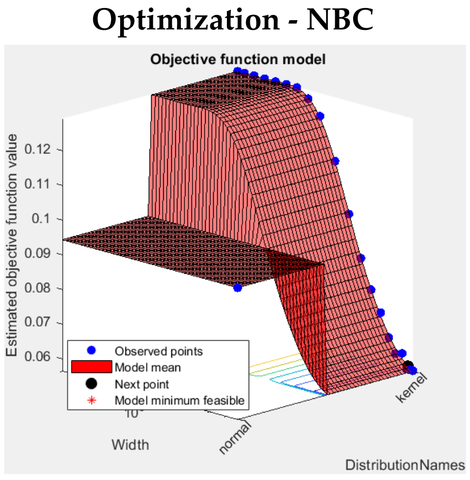

### Decision Tree




### ROC Curve - Multinomial NBC




### Optimization - NBC




## Results and Discussion

The first major result concerns the high accuracy rate on train data, namely 94.4% for Decision Tree and 94.6% for NBC, whose AUC in the ROC curve was 0.922. Our models do not seem too hindered by overfitting issues, as the Optimization graphs may testify. As expected from literature (Huang et al. 2004), the performances of Multinomial NB and Decision Tree were very similar. From our point of view, the most interesting characteristic of NBC was the chance to specify how to represent the text messages. On the other hand, we truly appreciated the Decision Tree because it offers a truly clear and intuitive way of visualising the message classification flow. As for the improvement of our models, future work may include testing a different multinomial NBC with Boolean attributes, which showed great accuracy (Metsis et al. 2006), and implementing a fuzzy decision tree algorithm (Cintra et al. 2013).

A second important insight concerns the importance of the preprocessing phase. Indeed, we are convinced that our models had good accuracy levels thanks to this preliminary work, which in our case mainly consisted in an intensive cleaning process and the creation of Word Clouds. We think that the Word Clouds are a simple yet powerful way to identify 'red flags' for spam filtering and consequenty for defining meaningful word predictors.

This insight is consistent with the results obtained from the comparison of the three different Naive Bayes Models tested. In fact, the Multinomial model predicted spam messages better than the Bernoulli one (+5.5% of accuracy). Both models relied on the 'spam dictionary' used for the Decision Tree Model. Without this classification of words, the Multinomial model performance dropped from 94% to 87%, a percentage similar to the Bernoulli one (87.9%). Thus, the Multinomial model allows better predictions, but only when based on an accurate feature definition.

In conclusion, a good preprocessing phase and the choice of good predictors for the data considered seems to matter as much as the classification algorithm employed.

## References

Cintra, M. E., Monard, M. C., & Camargo, H. A. (2013). A fuzzy decision tree algorithm based on c4. 5. *Mathware & Soft Computing*, *20*, 56-62.

Huang, J., Lu, J., & Ling, C. X. (2003, November). Comparing naive Bayes, decision trees, and SVM with AUC and accuracy. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on* (pp. 553-556). IEEE.

Metsis, V., Androutsopoulos, I., & Paliouras, G. (2006, July). Spam filtering with Naive Bayes - which Naive Bayes? In *CEAS* (Vol. 17, pp. 28-69).

Shimodaira, H. (2014). Text classification using naive bayes. *Learning and Data Note*, *7*, 1-9.

Tretyakov, K. (2004, May). Machine learning techniques in spam filtering. In *Data Mining Problem-oriented Seminar, MTAT* (Vol. 3, No. 177, pp. 60-79).