Machine Learning Project

Classification of news articles

Jakob Sahlström jasa5691@student.uu.se
Anders Lindström anli6945@student.uu.se
Jesper Dürebrandt jedu6357@student.uu.se

Friday 25th April, 2014

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

1 Introduction

Describe the problem Related work here maybe? Introduction and stuff

2 Theory

2.1 Naive Bayes classifier

Let c be the class and $A = a_1, \ldots, a_n$ be the attributes of a document. Then with Bayes Theorem

$$p(c|A) = \frac{p(A|c)p(c)}{p(A)}, \qquad (1)$$

the attributes A is classified as class C if and only if

$$f_b(A) = \frac{p(C|A)}{p(\neg C|A)} \geq 1, \qquad (2)$$

where $f_b(A)$ is called a *Bayesian* classifier. Assuming all attributes are independent given the class,

$$p(A|c) = p(a_1, \ldots, a_n|c) = \prod_{i=1}^{n} p(x_i|c)$$

the final classifier can be written as

$$f_{nb}(A) = \frac{p(C)}{p(\neg C)} \prod_{i=1}^{n} \frac{p(a_i|C)}{p(a_i|\neg C)} \qquad (3)$$

where f_{nb} is called the *Naive Bayesian* classifier.

Two models that uses the Naive Bayes assumption are the *multi-variate Bernoulli* model and the *multinomial* model. The main difference is that

in the Bernoulli model the attributes are binary, indicating if a word from a vocabulary has occurred at least once or not. In the multinomial model the frequency of words are taken into account.

2.2 Random Forest

Random Forest (RF) is based on building several considerably small classification trees. Consider having a feature vector that is of length N, then randomly select $n << N$ of those features. Build the trees with some kind of algorithm (e.g. C4.5), where information gain is taken into consideration, and no pruning is done (i.e. expand the tree fully). Then repeat selecting n new variables from N until the wanted number of trees are built.

After all the trees are built, they can be used to let a new vector of data pass through all the trees and then letting each tree *vote* on what class the vector most probably should be a part of.

Algorithm 1 Random Forest

Let $x = (x_1, x_2, ..., x_N)$ be a set of features;
while *not enough trees* **do**
Randomly pick with replacement a subset containing $n << N$ features;
Use training set to build a decision tree using a classification algorithm, e.g. C4.5, except no pruning is done.
end while

3 Method

Method and stuff

4 Results

Result and stuff

5 Related work

Naive Bayes models are widely used because of it's simplicity and efficiency. A. McCallum and K. Nigam compares the two most common models, the multivariate Bernoulli and the multinomial model. They are explained in detailed both theoretically and empirically and in general the multinomial model outperforms the Bernoulli model [4].

Random Forest is a method used widely when talking about classification. It has been used to classify web documents by keywords, where it for five and seven topics performed better than e.g. Naive Bayes and MLP [1]. In I. Kopriska, J. Poon, J. Clark, J. Chan's paper, they use Random Forest for classifying e-mails. It was able to out perform other methods, such as DT, SVM and NB[2].

6 Conclusions & Future work

Conclusion and stuff

References

[1] Myungsook Klassen and Nikhila Paturi. Web document classification by keywords using random forests, 2010.

[2] Irena Koprinska, Josiah Poon, James Clark, and Jason Chan. Learning to classify e-mail, 2006.

[3] Andrew McCallum and Kamal Nigam. A comparison of event models for Naive Bayes text classification, 1998.

[4] Harry Zhang. The Optimality of Naive Bayes, 2004.