A comparison of models for the age identification problem

Anonymous

1. Introduction

Stories and news have traditionally been expressed through long stories and newspapers. Today, through social media platforms such as Twitter and Facebook, stories and opinions are now told within the space of 280 characters. With smaller text sizes, the already challenging machine learning problem of identifying authors has become even more difficult. The goal of this report is to build and critically analyse some supervised Machine Learning methods to automatically identify the age of an author, based on a short text. Through identifying the age of an author, we can make steps towards author identification.

This report looks at three of the more popular classifiers for text classification, Multinomial Naïve Bayes, Support Vector Machines and K Nearest Neighbour, comparing and contrasting the effectiveness and choices for these learners.

2. Datasets utilized

For the purpose of this study, there was an analysis of a curated dataset of tens of thousands of blogs, utilising over 270,000 distinct blog entries[2.1]. Two pre-processed datasets were used. For dataset 1, all raw words in the blog posts were considered, allowing more accurate comparison through transforming the data by stripping punctuation and capitalization. Dataset 2, considered the frequency of 30 token words that are likely indicative of different age groups, and reduces the age to buckets between 14-16, 24-26, 34-36, 44-46 and outside of these groups.

2.1. Resources

Original data provided by report: Schler, Jonathan, Moshe Koppel, Shlomo Argamon, and James Pennebaker (2006) Effects of Age and Gender on Blogging. In Proceedings of 2006 AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs. Stanford, USA.

3. Naïve Bayes

One of the most well known tools for assessing text classification and natural language processing is Naïve Bayes. A very fast and simple to implement classifier, this classifier is a strong baseline for the age classification problem. For the training set, all distinct words in the blog posts were considered, creating a dictionary of all words tokens, their corresponding frequencies and classes. For predictions, the conditional probability of the token and class as the relative frequency of the documents belonging to a specific class is used for prediction.

While cheap, requiring no parameter tuning, and requiring only a small amount pre-processing, this classifier makes a questionable assumption that features are independent. This independence is central to Naïve Bayes, being used to get the corresponding conditional probabilities used to make predictions, and thus must always be considered. This means that predictions may not always be indicative. Furthermore, despite having a very high number of instances (which suits Naïve Bayes), outlier words can appear with a low frequency, leading to heavier weightings to unrelated classes.

When this classifier was tested on a test set of 43,000 instances, the classifier scored a 0.19 for predicting a specific age, by evaluating the number of correct predictions over the total number of predictions. It achieved 0.46 for predicting whether a post was within an age bucket or. However, upon closer analysis of the confusion matrix, the classifier predicted most classes to be within the 24-26 age bucket and only small numbers of predictions in other classes. Thus, reducing the number of features in the training data within the 24-26 range increases the diversity in class guesses. Although this leads to a lower evaluation score for the test instances, it is likely better applicable and more informative, in particular to writers outside of this age group.

4. Support Vector Machines

One of the highest performing classifiers, Support Vector Machines are respected as a strong classifier, having been applied with success to information retrieval problems, particularly text classification2. By dividing into linearly separable classes, SVMs have proved to create strong classifiers. However, due to SVMs inherently being used for problems with two discrete classes, thus utilising Dataset1 requires it does not lend itself well to a wide range of ages. Thus Dataset2 was utilised, dividing into 4 discrete age buckets and taking note of token frequencies. Through these reductions, differences between class values became much more apparent for the classifier, becoming more linearly separable, leading to a better accuracy of 0.41.

5. K-Nearest Neighbours

The final classifier that was utilised was K nearest Neighbour, a supervised learning method that translates instances and their corresponding classes into vectors, creating a geometric model from the training data. The system then predicts class by choosing a majority class from the nearest k neighbours. K-NN was chosen as a suitable classifier for this problem due to the high number of training instances used, having increased reliability drawing on the learned data. K-NN is also appropriate in a problem where there are several different age buckets, presenting itself as a multiclass problem, being able to handle high dimensional data very well.

For K-NN, the frequencies of 30 token words were utilised rather than the raw data, utilising all words in the training data as tokens. This leads to a significantly lower number of features, reducing the cost for calculating distance, decreasing the time taken for an already expensive classifier. This also reduces the number of noisy/irrelevant features, which can degrade the performance of the classifier. Furthermore, the reduction of classes to 4 buckets, each treated as a distinct class, in comparison to a large number of discrete values, leads to more distinct class boundaries, and thus a better classification, while remaining useful as a tool for age analysis.

According to accepted practice, k is often chosen to be the square root of the number of features, in this case 212. However, when applied to the dataset, this evaluated very poorly, scoring 0.2. Instead, this classifier was cross evaluated against a large range of k values, evaluating against the test instances, and k=101 scored much higher at 0.39. Also, a k value was chosen such that it was not a multiple of the number of classes, theoretically reducing the number of ties. Although choosing k is extremely time consuming, this is only required in the preprocessing step, and thus this classifier can still be applied in reasonable time.

6. Conclusions

Surprisingly, from evaluation metrics, Naïve Bayes performed the best out of the three classifiers. This difference in performance is likely due to K-NN and SVM classifiers being trained only on data within 4 age buckets, not predicting classes outside of these ranges. However, Naïve Bayes' classifier appears to predict numbers similarly almost akin to 0R, heavily favouring a single class bucket. In conclusion, for a better prediction that is more useful to users, I conclude that this implementation of K-NN is the strongest of these three classifiers.

References

M. Kay. 1986. *Parsing in Functional Unification Grammar.* In "Readings in Natural Language Processing", B. J. Grosz, K. Spark Jones & B. L. Webber, ed., pages 125-138, Morgan Kaufmann Publishers, Los Altos, California.

Schler, Jonathan, Moshe Koppel, Shlomo Argamon, and James Pennebaker (2006) Effects of Age and Gender on Blogging. In Proceedings of 2006 AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs. Stanford, USA.

https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-and-machine-learning-on-documents-1.html

http://www.cs.cmu.edu/~knigam/papers/multinomial-aaaiws98.pdf

http://blog.datumbox.com/machine-learning-tutorial-the-naive-bayes-text-classifier/

https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-and-machine-learning-on-documents-1.html

http://web.cs.iastate.edu/~honavar/text-classification-SVM.pdf