Machine Learning Algorithms and Their Characteristics in Supervised Multi-class Text Classification Problem

Anonymous

1. Introduction

In this Digital Era, it is not surprising that text classification is widely used and studied in many domains as the volume of digitized text documents are increasing exponentially every day. Machine learning with text is often challenging as text data are very distinct from other data types (e.g. numeric, categorical) as they cannot be the input of a machine learning algorithm directly. Thus, feature extraction and selection are crucial in building a good text classification model that runs efficiently. The machine learning algorithms should also be selected carefully as they have to be capable of handling the special characteristics of the text data after feature extraction. As such, a variety of machine learning algorithms have been studied thoroughly for text classification problems by a lot of researchers, including decision trees, rule-based classifiers, Support Vector Machines, Bayesian classifiers, and Neural Network classifiers. In this report, three well-known text classification algorithms, Multinomial Naive Bayes (MNB), Logistic Regression (LR), Support Vector Machine (SVM) will be studied and discussed in details through a supervised multi-class text classification problem to understand the algorithms' properties and behaviours. These three algorithms are selected because of their popularity and consistent satisfactory predictive accuracy in many literatures. The supervised multi-class text classification problem that we are going to look at is about predicting the author's age group from their blog post(s)..

2. Related Literature

In [4], McCallum and Nigam (1998) pointed out that, in many real world classification problems, documents in a specific category consists of a wide range of subjects with overlapping vocabularies, rather than just about a narrow subject with limited vocabularies. Consequently, a large vocabulary list is essential for satisfactory accuracy rate in a text classification task. And as their experiment results showed that the performance of multivariate Bernoulli NB on large vocabularies and documents with large variance in length was not very ideal, MNB is more suitable for challenging text classification tasks.

In [5], according to Joachims (1998), SVMs are very well suited for text categorisation because of the particular properties of text: high dimensional feature spaces and sparse instance vectors. He pointed out even in such high dimensional feature spaces, SVMs do not have the need for feature selection or parameter tuning to achieve a satisfactory result and thus they are highly robust.

In [6], according to Rennie, Jason D., et al., although NB is fast and easy to implement, it is flaw not only because its strong independence assumption, but also because it has skewed data bias - more training examples for one class than another can cause the decision boundary weights to be biased.

3. Methods

3.1. Data Representation & Preprocessing

As text data cannot be used directly in machine learning algorithm, the features have to be extracted from text and converted into vectors of numbers. There are many ways of doing feature extraction from text data, including word2vec and Latent Semantic Indexing. In this task, text is represented as "bag-of-words" (BOW) as this representation best suit the purpose of the task. In BOW, text is represented as a set of words and their associated frequency in the text. The advantages of BOW are it is easy to understand and implement. BOW also offers a great range of flexibility for customisation. However, the grammar, semantics, and order of words in the text are lost under the BOW representation. Moreover, the high dimensionality and sparsity of the data after vectorisation also make BOW less user-friendly. BOW was selected as the text data representation for this task despite of its shortcomings is because its advantages outweigh its disadvantages. Grammar, meaning, and emotion are not something we focus a lot on in this task and thus this degree of information loss is acceptable. More importantly, BOW takes into account of the frequency of a word appeared in

the text. This is very important to this task as the more often you see a word in certain age group, the more likely it is a valuable feature for categorisation. In conclusion, as long as the data matrix size and sparsity are carefully managed, BOW model is well-suited to represent text data in this task.

3.2. Feature Extraction & Selection

Feature selection is a crucial step in text classification as text data are generally high dimensional and contain irrelevant, noisy features. The most common feature selection for text data is stop word removal. This ensure features that are very likely to be picked for the learning process due to their high frequency in the text are actually valuable and meaningful. Other popular feature selection methods include Gini Index, Information Gain, Mutual Information, and Chi-squared statistic. In this task, stop words and words that show in less than 5 documents and 30% of the total documents will be discarded so that only useful words will be extracted and used in the classifiers. This reduces noises and feature size so that the classifiers can be trained more efficiently and effectively.

Term frequency-Inverse document frequency (TF-IDF) is also used in the task to calculate the word frequency more effectively by highlighting words that are more valuable/unique (e.g. words appear frequently on a few documents but not those appear on more than 90% of total documents).

3.3. Classification Algorithms

These three models are chosen because of their characteristics to handle sparse, high dimensional text data and good performance in many literatures. As text classification is a high dimensional problem and it is likely to be linear separable because of the high dimensionality, all three of the classifiers are linear classifiers.

3.3.1. Multinomial Naïve Bayes (MNB)

MNB is used in this task because it is easy to implement, computationally fast, and handle large amount of data. Alpha = 1.

3.3.2. Logistic Regression (LR)

LR is used in this task because it is a simple model, computationally fast, and highly interpretable. C = 1.

3.3.3. Support Vector Machine (SVM)

LinearSVC is used in this task because it is linear, and be able to handle sparse, high dimensional data without rigorous tuning. Calculated class weight from class labels is applied in the class_weight parameter in LinearSVC. “Hinge” is used in the loss function as it yields the best accuracy after a few attempts. C =1.

4. Result

4.1. Accuracy

Accuracy means the ability of the model to predict the class label correctly. These are the best results for the following feature selection methods:

	Accuracy (MNB)	Accuracy (LR)	Accuracy (LinearSVC)
CountVectorizer (max_df=1)	46.99%	50.88%	50.89%
TfidfVectorizer (max_df=1)	51.61%	51.61%	53.40%
CountVectorizer (max_df=.3)	46.01%	51.60%	52.46%
TfidfVectorizer (max_df=.3)	51.58%	53.46%	53.83%
TfidfVectorizer (max_df=.3, n_grams=(1,2))	51.96%	51.58%	**54.77%**

Table 1: Accuracy Comparison

In general, LinearSVC has the highest accuracy among the three learners, LR comes second, and MNB comes last. Lowering the max_df from 1 to 0.3 does increase accuracy for all three algorithms, showing that eliminating unhelpful data/noise does improve accuracy marginally. The last model is a combination of unigram and bigram and it does increase the accuracy because of the additional information gained from those bigrams.

5. Error Analysis & Discussion

5.1. Error Analysis

Below is the confusion matrix for the best-performing model (LSVC-bigram-unigram):

```
[[ 7925  5165     8     2     0]
 [ 1800 15451    42     4     0]
 [   56  2520     7     1     0]
 [   17   531     3     0     0]
 [ 2765  8985    46     3     0]]
```

The majority of misclassification take place

between the ’14-16’ and ’24-26’ age groups. This is not something that is unexpected as people who are in “14-16” and “24-26” do share similar values and interests. There are over 15000 instances are misclassified as the “24-26” age group. One of the possible reasons for this is that the label data in the training set is highly skewed and almost half of the training data are under the “24-26” age group. Even though class_weight had been assigned for the LSVC-bigram-unigram model trying to offset the effect of the imbalanced data, the class_weight might not be optimal to drastically decrease the effect.

In addition, since the model does not predict classes outside of the 4 designated one in the training dataset, the last row of confusion matrix does make sense. If we can modify the model and make it able to predict the unseen class, the accuracy rate would be significantly increased as there are over 10,000 instances of misclassification for the unseen class.

5.2. Discussion

Below we are going to discuss the characteristics of the machine learning algorithms and what we have learnt from this task:

5.2.1. Multinomial Naïve Bayes

MNB performs fairly well in this task despite the strong, untrue conditional independence assumption. It might be because the assumption works consistently with the ‘bag-of-words’ data representation.

MNB is a high-bias generative algorithm, so it may fail to capture important patterns and underfit the data. On the other hand, high-bias algorithms are usually simpler and thus easier to interpret. It is, indeed, the easiest model for interpretation compared to the other two algorithms. Tuning is almost effortless and its computational cost is lower than the other two algorithms.

5.3.2. Support Vector Machine

Despite of the sparse, high dimensional characteristics of text data, SVMs tend to perform well because the features tend to correlated with each other and thus organised into linearly separable categories [3]. The linear separating hyperplanes makes SVM very robust to high dimensionality. It is also because SVM is capable of handling large number of features very gracefully and does not tend to overfit.

5.3.3. Logistic Regression

Logistic Regression works well in this task because it is a linear separable task and LR, as a linear classifier, is very likely to do well in this task. It also has mechanism (C parameter) to control the complexity of the classifier and thus prevent from overfitting.

6. Conclusions

Through working on a multi-class text classification problem, we explored and studied different characteristics of MNB, SVM, and LR. They all work well in this problem because they are all linear classifiers. SVM performed the best among these three classifiers because of its robustness to high dimensional feature spaces and the ability to handle a large dataset without overfitting.

References

[1] 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.
[2] Ng, A. Y., & Jordan, M. I. (2003) On Discriminative vs. Generative.
[3] Aggarwal C.C., Zhai C. (2012) A Survey of Text Classification Algorithms. In: Aggarwal C., Zhai C. (eds) Mining Text Data. Springer, Boston, MA.
[4] McCallum, A., & Nigam, K. (1998, July). A comparison of event models for naive bayes text classification. In AAAI-98 workshop on learning for text categorization (Vol. 752, No. 1, pp. 41-48).
[5] Joachims, Thorsten. "Text categorization with support vector machines: Learning with many relevant features." European conference on machine learning. Springer, Berlin, Heidelberg, 1998.
[6] Rennie, Jason D., et al. "Tackling the poor assumptions of naive bayes text classifiers." Proceedings of the 20th international conference on machine learning (ICML-03). 2003.