COMP9318 Project Report

May 2018

By Shuning Zhao z3332916
Chunnan Sheng z5100764

Abstract

For this project we are required to devise an algorithm/technique to fool an SVM based binary classifier known as the target-classifier. To do achieve this, we are only allowed to make 20 changes to the testing samples.

Introduction

The dataset provided for this project are paragraphs of text extracts. We are given three files where class-1.txt contains 180 paragraphs for class 1, class-0.txt contains 360 paragraphs for class 0 and test_data.txt contains 200 paragraphs for class 1.

The files class-1.txt and class-0.txt are to be used for training and test_data.txt is to be used for testing. All data in the files provided are subsets of the training and testing datasets used for the final evaluation.

There is no further information available on the metadata such as where the data came from, what are the paragraphs about, what does the two classes 0 and 1 represent and how the samples given to us were sampled from the final evaluation data.

Methodology

Our approach to this project is to construct multiple SVM based models and evaluate its performance on the data provided. If we are satisfied with the performance of this model we will attempt to fool the model and submit the results for feedback.

For text data representation we tried bag of words with scoring methods such as binary, counts, frequencies and term frequency- inverse document frequency (TF-IDF) to search for the optimal results (Zhang, Jin and Zhou, 2010).

For bag of words with word counts, to avoid attributes with greater numeric ranges dominating those in smaller numeric ranges (Dodge, 2006) normalization methods such as standard score in the case of RBF kernel and feature scaling has been adapted to standardize the data (Jebara, 2004).

Since pre-processing of the data is not allowed all records of the training data will be loaded for model training. However, there is a great imbalance in the training data with 180 class 1 and 360 class 0. This could lead to potential issue as an SVM classifier trained on an imbalanced dataset can produce suboptimal models which are biased towards the majority class and have low performance on the minority class. (Japkowicz and Stephen, 2002)

Common methods of balancing the classes of SVM include different error costs (DEC), one class learning, zSVM, kernel modification methods, resampling and ensemble learning methods (Batuwita and Palade, 2013). Due to the constrains imposed by the training strategies, we will be only using ensemble learning methods for class balancing in this project.

The method used for ensemble class balancing is as follow:

1. Split the class 0 file to two sets of 180 paragraphs using stratified split in sklearn to preserve the original distribution of the data in each subset.
2. Train the first SVM model using the first half of class 0 paragraphs and all of class-1.txt.
3. Train the second SVM model using the second half of class 0 paragraphs and all of class-1.txt.
4. Since there are only 2 models, we cannot deploy majority voting for final decisioning. And due to restrictions on the training strategies we were unable to output the probabilities for model averaging. Hence, we just average the weights of the two SVM models for feature interpretation.

The kernels used in the project are linear, RBF and a customized kernel where we multiplied all class 1 data in the training data by the class 0 / class 1 ratio.

All other parameters for model training are obtained via grid search.

To decide what changes to make, it is mentioned in (Guyon et al., 2002) that a linear SVM creates a hyperplane that uses support vectors to maximise the distance between two classes. The absolute values of the coefficients in relation to each other can be used to determine feature importance for the data separation task. Where the large positive coefficients indicate the features contributing the most to the paragraph being classified to class 1 and the small negative coefficients indicate the features contributing the most to the paragraph being classified to class 0.

For RBF kernels the coefficients can be calculated by multiplying the support vectors and the dual coefficients. Those coefficients act as a decent approximation to the SVM-RFE feature selection for SVM with RBF Kernel (Liu et al., 2011) due to the training strategy constrains.

After working out the feature importance, we used two change strategies as follow:

1. Delete and Insert: Obtain the top 20 absolute values of the coefficients. Remove the corresponding word if the coefficient is positive. Insert the corresponding word if the coefficient is negative.
2. Delete Only: Remove words with the top 20 positive coefficients.

Results

The results for some selected models are in the table below:

Kernel	Text Data	Normalization	Class-Balance	Training Accuracy	Test Accuracy	Strategy	Feedback
Custom	Word Count	Feature Scaling	Ensemble	100%	95%	Delete Only	70.5%
Custom	Binary	Feature Scaling	Ensemble	100%	94%	Delete Only	68.5%
Linear	Binary	None	None	100%	61%	Delete and Insert	30%
Linear	Word Count	Feature Scaling	No balance	100%	62.5%	Delete Only	64%
Linear	Binary	Feature Scaling	None	100%	61%	Delete Only	62%
linear	TF-IDF	None	Ensemble	100%	99.5%	Delete Only	68.5%
RBF	Word Count	Standard Score	Ensemble	60%	65%	Delete Only	62%
RBF	TF-IDF	None	None	0%	0%	Delete Only	90.5%

Conclusion

For our submission we submitted the version with RBF kernel, TF-IDF encoding with no normalization or class balance. The parameters found by grid search were C = 1 and Gamma = Auto with delete only as the change strategy as it had the best performance from the online real-time feedback.

We believe the reason for the above model to have a better online feed-back performance than the other models tested is because the models we trained were incorrect guesses of the target-classifier.

Reference

- Batuwita, R. and Palade, V., 2013. Class imbalance learning methods for support vector machines。

- Dodge, Y. ed., 2006. *The Oxford dictionary of statistical terms*. Oxford University Press on Demand.

- Guyon, I., Weston, J., Barnhill, S. and Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. *Machine learning*, *46*(1-3), pp.389-422.

- Japkowicz, N. and Stephen, S., 2002. The class imbalance problem: A systematic study. *Intelligent data analysis*, *6*(5), pp.429-449.

- Jebara, T., 2004, July. Multi-task feature and kernel selection for SVMs. In *Proceedings of the twenty-first international conference on Machine learning* (p. 55). ACM.

- Liu, Q., Chen, C., Zhang, Y. and Hu, Z., 2011. Feature selection for support vector machines with RBF kernel. *Artificial Intelligence Review*, *36*(2), pp.99-115.

- Zhang, Y., Jin, R. and Zhou, Z.H., 2010. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, *1*(1-4), pp.43-52.