Final Project Report

Abhishek Kumar
Department of Computer Science
Stony Brook University
Stony Brook, NY
kumar1@cs.stonybrook.edu

Abstract

This document presents a brief report on the implementation of the paper `Finding Deceptive Opinion Spam by Any Stretch of the Imagination[1]`. The work is based on the study of *deceptive opinion spam* - fictitious opinions that have been deliberately written to sound authentic. The working implementation is restricted to a specific methodology adopted in the original paper.

1 Introduction

The main focus of the paper is detecting deceptive opinion spam (inappropriate or fraudulent reviews) through machine learning classifiers. The paper shows how ML classifiers namely Naive Bayes and SVMs have outperformed human level performance significantly. The gold-standard dataset used in the paper is readily available over the internet. The text categorization approach with unigrams as features has been trained using SVM classifiers in the implementation. The aim will be to achieve statistics such as accuracy, precision and recall very similar to what is already reported in the paper through nested cross validation experiments.

2 Implementation

2.1 Dataset

The gold standard data set is available for use at https://www.aclweb.org/anthology/P11-1032. This corpus consists of truthful and deceptive hotel reviews from 20 hotels in the Chicago area, described in [1]. Specifically, this corpus contains

(a) 400 truthful reviews from TripAdvisor.com

(b) 400 deceptive reviews from Amazon Mechanical Turk

The text data is encoded in POS format with unigrams and bigrams as features.

2.2 Algorithm

A specific part of the original paper uses SVM (Support Vector Machine) on *unigrams* and the same has been implemented in code. Features have been extracted using *unigrams* and which are lowercased and unstemmed. A ***tf-idf*** (term frequency–inverse document frequency) matrix is then constructed from these features which is then passed onto the SVM classifier for training. The implementation has been restricted to SVM with linear kernel function in the original paper. The working implementation is in the Python3 programming language developed on Jupyter Notebook.

3 Results and Discussion

The deception detection strategy adopted in the implementation are evaluated using a 5-fold nested cross-validation(CV) procedure(Quadrianto et al.,2009), where model parameters are selected for each test fold based on standard CV experiments on the training fold.The folds are selected in such a way (evident in the code) that the models are always evaluated on reviews from unseen hotels. A comparative analysis between the results obtained via my implementation and the original paper results are shown in Table 1. As can be seen in the table, the implementation numbers are very close to the reported numbers in the original paper. Following are some key assumptions undertaken while implementing the deception detection strategy.

- The term weighting scheme used in the implementation is tf-idf which is not mentioned clearly in the actual paper.

- Since the nested cross validation is used to tune the hyper-parameter C which is the

		TRUTHFUL			DECEPTIVE		
Approach(Unigrams - SVM)	Accuracy	P	R	F	P	R	F
Original Paper	88.4%	89.9	86.5	88.2	87.0	90.3	88.6
Actual Implementation	87.8%	87.2	88.7	87.9	88.5	87.0	87.7

Table 1: A comparative analysis of the SVM classifier performance for the unigrams approach based on nested 5-fold cross-validation experiments. Reported precision, recall and F-score are computed using a micro-average, i.e., from the aggregate true positive, false positive and false negative rates. **The results are reproducible in code**.

regularization parameter (controls the bias-variance trade-off), the domain of hyper-parameter to range over is not specified in the paper. So, the implementation uses a particular set of values to optimize over.

- The SVM implementaion is based on sklearn libsvm and differs from the SVM^{light} in the paper.

4 Conclusion and Future Work

There are many other aspects covered in the paper which have not been implemented and can be further extended upon. Text categorization with unigrams gives decent accuracy on the test data, being much higher than the baseline performance given by simple genre identification. This can be further improved upon by alternatively using bigrams and trigrams of the text data. Naive Bayes also has shown to be equally accurate in detecting opinion spam as reported in the paper. If using the SVM as the classifier, the following are ways in which the accuracy of the classifier may further be improved upon:

- Experimenting with different kernel functions in SVM : *rbf, nonlinear, sigmoid, polynomial* for better results.
- Optimizing the hyper-parameter C over a large range.
- SVM on bigrams and trigrams as features.

References

[1] M. Ott, Y. Choi, C. Cardie, and J.T. Hancock. 2011. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies..

[2] N. Quadrianto, A.J. Smola, T.S. Caetano, and Q.V. Le. 2009. Estimating labels from label proportions. The Journal of Machine Learning Research, 10:2349– 2374.

[3] G. Forman and M. Scholz. 2009. Apples-to-Apples in Cross-Validation Studies: Pitfalls in Classifier Performance Measurement. ACM SIGKDD Explorations, 12(1):49–57.