SHIVA KUMAR KANNEBOINA

MLND - Capstone Project Proposal

Title: Detecting Online Fake Reviews (Opinion Spam)

Date: 21/11/2017

Domain Background:

Almost everyone of us in this modern era depend on the online reviews before purchasing any product or booking any service via online. When deciding on whether to go for that product/service, these reviews plays a huge role and act as major source of information on the overall product, service or the organization which is providing that product/service. Due to drastic change in the technology in online market recent years, some companies are paying to people to write fake reviews on their product or service so that their reputation in the market will be high. This problem of detecting fake reviews was first addressed by a group of researchers at Cornell University, these guys have designed an algorithm by analysing the language used in legitimate and phony write-ups, their work details are published here. Here in this problem I am trying to design a similar model with my approach to detect fake reviews on hotels.

Problem Statement:

The main goal of this project is to find out if a particular review is fake or not by effectively using the Classification techniques available in Machine Learning space and find the best algorithm that accurately predicts by effectively training on large datasets available here. The algorithm is expected to learn from the large dataset and predict the future reviews. The major part of the problem is to find out a best algorithm that acts well on this dataset.

Datasets and Inputs:

Dataset/Corpus I want to use in this project can be downloaded from here. This corpus consists of truthful and deceptive hotel reviews of 20 Chicago hotels.

This corpus contains:

- ➔ 400 truthful positive reviews from TripAdvisor (described in [1])
- ➔ 400 deceptive positive reviews from Mechanical Turk (described in [1])
- ➔ 400 truthful negative reviews from Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor and Yelp (described in [2])
- ➔ 400 deceptive negative reviews from Mechanical Turk (described in [2])

Each of the above datasets consist of 20 reviews for each of the 20 most popular Chicago hotels. The files are named according to the following conventions:

- Directories prefixed with fold correspond to a single fold from the cross-validation experiments reported in [1] and [2].
- Files are named according to the format %c_%h_%i.txt, where:
    - **%c** denotes the class: truthful or deceptive
    - **%h** denotes the hotel: [Ex: affinia: Affinia Chicago (now MileNorth, A Chicago Hotel)]
    - **%i** serves as a counter to make the filename unique

Solution Statement:

To solve this problem I will use range of classification algorithms available in the Machine Learning(such as RandomForest classifier with GridSearchCV to start with, SVM, NaiveBayes Classifier, etc..) and compare the metrics like accuracy_score for each algorithm and pick a model that gives best results and use that algorithm for prediction. The solution is expected to use some NLP techniques for data pre-processing.

Benchmark Model:

In a test on 800 reviews of Chicago hotels, Cornell researchers have developed a computer software that's pretty good at detecting deceptive reviews with almost ~90% accuracy. This will be the aspirational target for this problem that I want to achieve.

Evaluation Metrics:

Model's accuracy will be judged based on the predictions that will be made on the dataset described above. Model's evaluation metric will be transcription accuracy on the test data in the corpus mentioned above. Accuracy is defined as correctly predicting whether a particular review is real or fake.

Project Design/Approach:

My approach to solve this problem is discussed in this section. I can use Parts Of Speech(POS) Tagging and Bag Of Words(BOW) techniques for addressing this problem. As like any other machine learning problem, first step is to download and preprocess the data which involves tasks like cleaning the noise from the corpus and prepare it such a way that it suffices with my solution needs. This preprocessing step might use some techniques like stop_words, lemmatization, POS Tagging, Bag Of Words, etc.. to prepare data.

- Will use the 70% of the data for the training and the remaining 30% can be used for testing purpose. And might use cross-validation techniques if required to split the data.

- Train and test various range of classifiers and note down the metrics for each model that is executed. Once I feel that my classifier is giving me better results, then I can choose that model and predict the reviews.

All reviews will be classified into four classes as below:

- **Class1:** POSITIVE: Any review that is marked as positive and it is truly positive will be classified as this class.
- **Class2:** HIGH POSITIVE: Any review that is marked as positive and it is deceptive will be classified as this class.
- **Class3:** NEGATIVE: Any review that is marked as negative and it is truly negative will be classified as this class
- **Class4:** HIGH NEGATIVE: Any review that is marked as negative and it is deceptive will be classified as this class.

The solution of predicting whether a given review is fake or real is obtained by using the following strategy:

- ➔ If the review falls into either POSITIVE(Class1) or NEGATIVE(Class3), then it is true review, i.e. **Review is not fake.**
- ➔ If the review falls into either HIGH POSITIVE(Class2) or HIGH NEGATIVE(Class4), then the review is fake, i.e. **Review is fake.**

References/Research Links:

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. M. Ott, C. Cardie, and J.T. Hancock. 2013. Negative Deceptive Opinion Spam. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.
3. http://myleott.com/op_spam/
4. http://news.cornell.edu/stories/2011/07/cornell-computers-spot-opinion-spam-online-reviews
5. http://www.ijerd.com/paper/vol12-issue4/Version-1/A1240108.pdf
6. http://www.vladsandulescu.com/opinion-spam-detection-literature-review/
7. http://courses.washington.edu/ling575/SPR2014/slides/OpinionSpam.pdf
8. https://arxiv.org/pdf/1107.4557.pdf