MACHINE LEARNING ENGINEER NANODEGREE

Capstone Proposal
Meena Joshi
July 20th 2017

DOMAIN BACKGROUND:

NLP (Natural Language Processing) is a field of study that focuses on the interactions between human language and computers. It is a way for computers to analyze, understand and derive meaning from human language in a smart and useful way. By utilizing NLP, we can organize and structure knowledge to perform tasks such as automatic text summarization, translation, named identity recognition, relationship extraction, sentiment analysis, speech recognition, topic segmentation and more. NLP algorithms are based on machine learning algorithms, i.e. it can rely on machine learning to automatically learn rules by analyzing a set of examples, and making the inference.

The ability of computers to understand human language has become more relevant with the exponential growth of data, with wide applications in real world. For example, categorizing news stories by topics, organizing emails into various groups (social, promotion, primary), etc. I am interested to built a prediction model that can accurately classify which texts are spam, to overcome the ever-increasing problem of unwanted texts.

PROBLEM STATEMENT:

Almost everyone gets unwanted messages, which costs time, money and resources to process, filter or manually delete. Given a set of text messages, the goal is to identify which texts are spams. The machine Learning algorithm can categorize the messages as spam or non-spam, by learning the rules based on the key features of the texts. The ability to detect linguistic patterns is an invaluable tool when applied to text messaging data.

The objective of this project is to automatically classify the text messages as spam or ham (legitimate), using a dataset of SMS messages in English of 5574 messages. I will first train a machine-learning model to learn to discriminate between ham and spam. Then I will use this trained model to classify arbitrary unlabeled messages as ham or spam.

DATASETS AND INPUTS:

I will be using the SMS Spam Collection, which is a public set of SMS labeled messages that have been collected for mobile phone spam research.
http://www.dt.fee.unicamp.br/~tiago/smsspamcollection/

The dataset has total 5574 English, real and non-encoded messages, with their tags as spam or ham. The messages are not in chronological order. Out of total 5574 messages, 4827 are ham and 747 are spams. The dataset is composed in one text file, where each line has two columns the correct class label (ham/spam) and the raw message. Below are some examples:

ham What you doing?how are you?
ham Ok lar... Joking wif u oni...
ham dun say so early hor... U c already then say...
ham MY NO. IN LUTON 0125698789 RING ME IF UR AROUND! H*
ham Siva is in hostel aha:-.
ham Cos i was out shopping wif darren jus now n i called him 2 ask wat present he wan lor. Then he started guessing who i was wif n he finally guessed darren lor.
spam FreeMsg: Txt: CALL to No: 86888 & claim your reward of 3 hours talk time to use from your phone now! ubscribe6GBP/ mnth inc 3hrs 16 stop?txtStop
spam Sunshine Quiz! Win a super Sony DVD recorder if you canname the capital of Australia? Text MQUIZ to 82277. B
spam URGENT! Your Mobile No 07808726822 was awarded a L2,000 Bonus Caller Prize on 02/09/03! This is our 2nd attempt to contact YOU! Call 0871-872-9758 BOX95QU

SOLUTION STATEMENT:

The goal of this project is to build a predictive model that can automatically classify ham or spam messages correctly. There will be some exploratory data analysis of the dataset with visualization, text-preprocessing steps. I will use python based Natural language toolkit (NLTK). Then the machine-learning algorithm will be used to train the model. I will use Naïve Bayes algorithm, which works well with textual data. The probabilistic model of Naïve Bayes classifier is based on Bayes theorem with an assumption that the features in the dataset are mutually independent. Bag of words features will be used to identify spams. The model will be used to make predictions for the labels of messages as spam or ham.

BENCHMARK MODEL:

The measure of success of the classifier is to check how accurate the prediction classes are. This dataset is strongly biased towards ham (4827 ham/ 747 spam). Even if I classify every message of the dataset as ham, I can get a minimum accuracy of 87%, which is obviously as indicator of a non-ideal classifier. So, I would choose benchmark model with accuracy 87% and my goal is to get more than 92% accuracy in Area Under ROC curve rather than the overall accuracy.

EVALUATION METRICS:

Now we need to determine how well our model will perform. There are some possible metrics for evaluating model's performance. The importance of the evaluation metrics depends on the task and the business effects of decisions based on the model. For our dataset, which is highly imbalanced (4827 ham and 747 spam) accuracy should not only be the correct evaluation metric as it will tell us nothing about the true predictive power of the classifier. For instance, even classifying each message as ham can get a minimum accuracy of 87%. So, I will use precision (which is the ratio of true positives and total sample which classifier predicted positive), recall (which is the ratio of true positives and total actual positives) and f1-score (which is the weighted average of precision and recall) along with accuracy as an evaluation metrics. I will use scikit learn's built-in classification report, which returns precision, recall and f1-score.

PROJECT DESIGN:

1. **Download dataset**- first step will be to download the dataset from UCI datasets.
2. **Exploratory Data Analysis**- Once the dataset is ready; will do some basic exploratory data analysis to get better understanding of the data.
3. **Text Preprocessing**- will convert the raw text messages into tokens, which includes splitting sentence to words, removing stop words, normalization and more. For this step I will use NLTK library, which is a standard library in Python to process text and has many useful features.
4. **Vectorization-** Now the tokens will be converted to vectors that a machine- learning algorithm can understand. I will use bag of words model, which learns vocabulary from all the documents, then models each document by counting the number of times each word appears. It will be done in 3 steps.
5. **Training a model-** Once messages are represented as vectors, we can train our classifier. There are many classification algorithms, which can be used here. I would prefer Naïve Bayes, which works very well for textual data.
6. **Evaluation of model-** will split the data into training and test set. The model will be trained on the training set and the final evaluation will be done on the test data, which will be true representative of predictive performance.
7. **Comparing models-** I will compare classifier's performance with other classification models to get the best model in terms of performance and computational efficiency.

REFERENCES:

https://en.wikipedia.org/wiki/Natural_language_processing
https://www.kaggle.com/c/word2vec-nlp-tutorial/details/part-1-for-beginners-bag-of-words
http://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html
https://github.com/udacity/machine-learning/blob/master/projects/capstone/capstone_proposal_template.md
http://www.nltk.org/book/
https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection