Machine Learning Nanodegree
NLP Capstone Proposal

Leonardo Cotti
June 12, 2017

Domain Background

The born of Natural Language Processing (a.k.a. NLP) is dated in the 1950s, in fact one of the first applications was a Georgetown experiment, in 1954, to translate sentences from Russian to English[1]. For three decades, the research of new NLP techniques focused on rule-based machine learning algorithms (such as Decision Trees), until a revolution happened in the 1980s. In these years the introduction of statistical model (such as hidden Markov model) and the availability of big and high-quality datasets (official translated documents of Canadian Parliament and European Union) led to dramatic improvements of translation models[2].

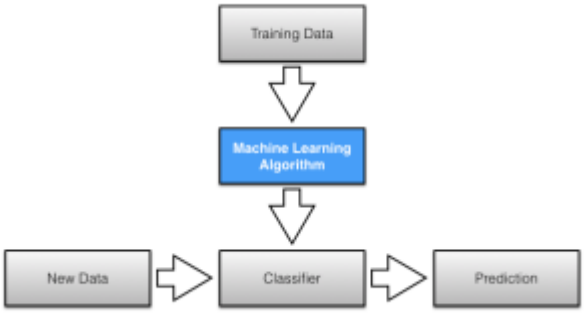
The trend of using more complex techniques in combination of big datasets has been continuing even today. Thanks to the availability of huge datasets and new techniques, such as vector word representation and deep learning[3], it was possible to achieve incredible results in many fields. For instance, in 2016, Google implemented a new Google Translate deep learning model that has outclassed each attempt made before. The outcome of the final model was comparable to human translation in term of translation quality[4].

In these days, a popular application of NLP is to create models for sentiment analysis. At first, sentiment analysis was used to prevent spam and fraud but now it is also applied to many business decisions making process[5]. Better understanding customer reviews, monitoring product's opinions on social media and understanding market trends are some of the business applications of NLP for sentiment analysis.

Most of the research papers of sentiment analysis are conducted in English and it is very difficult to find applications of machine learning in other languages. In fact, few NLP papers are available in my own language: Italian. That is why I will conduct a study and comparison to benchmark the feasibility of NLP techniques in the Italian language context.

Problem Statement

In this project, I will use Amazon.it reviews of products taken from different departments to create a Supervised model. I will create a Classifier to map reviews to the proper rating (1 to 5 stars). The structure of the problem is the standard Machine Learning Classification approach.



Input Data:	Matrix of numbers that represent the frequency of a given word in a sentence, derived from reviews using a bag-of-words representation
Classifier:	Classification model (such as SGDC, Naïve Bayes Classifier, Logistic Regression)
Prediction:	Discrete labels that represent if the review has a class 1 to 5 of rating

Datasets and Inputs

The reason because it is difficult to find applications of Machine Learning in the Italian language context is the lack of a ready-to-process dataset. There are only few datasets available for general regression problems, but there is no dataset available to process sentiment analysis. This means that I have had to create my own dataset.

I chose to focus on reviews of products because it is easy to find a sentiment, an opinion or words that classifies the quality of the product. The best website to find the best product reviews, full of words and with a good variety, is Amazon Italy.

I created a scraper to download around 11,500 reviews associated with a discrete rating labeled from 1 to 5. The reviews were taken from a balance mix of the book, fashion and electronics departments, I have chosen to focus not only on one department to avoid, at maximum, repetition of same words. The dataset allows me to split the reviews in a training set and a test set with enough number of elements to use.

I am confident to have found a good dataset because I can use a total of around 400,000 words that result in 37 words per sentence on average. On top of that a single word is repeated on average 15 times, it guarantees that the frequency of words is high enough to make the dataset representative.

Solution Statement

As stated before, the solution is to create a Classification Model that can predict sentiments of sentences (lists of words) using reviews of products related to the associated rating. I will try to find the best classifier using as many classifiers I can, then I will take the best and tune it to boost the performance at maximum.

To convert sentences in a matrix I will use the bag-of-words approach, that will convert the word list in a vector containing the frequency of words in a given sentence. Before the conversion I will analyze and clean the text with tools to eliminate factors of noise. Then, I will find the best model implementing different Classifiers: from Naives Bayes Classifier (highly used in NLP) to simpler Classifiers such as Logistic Regression or more complex Classifiers such as Stochastic Gradient Descent Classifier.

Benchmark Model

The scope of the project is to understand the feasibility of implementing sentiment analysis in the Italian language context, that is why the best way to benchmark will be to compare the same model starting from two different datasets: an English tested dataset and an Italian dataset created just for this scope.

Evaluation Metrics

The starting metric used in the project will be to draw the confusion matrix of the different models, then I will compute the Precision and Recall and finally I will directly compare the models using the F1-score. The F1-score will be the perfect metric to represent in one number the performance of the classifiers and it will be very useful to compare same classifiers but with training set in different languages.

Project Design

Tools:

- **Python 3** – Leading programming language in data science
- **Scrapy** – Open source framework for extracting data from websites
- **Scrapinghub.com** – Website that offers cloud solutions to scrape the web
- **Scikit Learn** – Open source machine learning library for Python

Workflow:

1) Identify a website with a consistent number with good variety of reviews in Italian
2) Create a Scrapy Spider[6] to crawl the data
3) Collect the data and clean it
4) Create a bag of words model as input for the Classifier using scikit-learn
5) Fit a Classifier to verify if the initial results are relevant enough (better than casual prediction)
6) If not go back to point 1) to improve the dataset
7) Try different Classifier and compare them to find the best one
8) Use grid search with cross validation to tune the best Classifier
9) Analyze the final model using confusion matrix, precision, recall, F1-score
10) Find a ready-to-train dataset in English with comparable size and diversity of the Italian one
11) Fit the English dataset with the same best Classifier
12) Compare the results of the two datasets and highlight eventual differences, pros/cons

[1] https://en.wikipedia.org/wiki/Georgetown–IBM_experiment
[2] https://en.wikipedia.org/wiki/History_of_natural_language_processing
[3] Stanford Course: CS224n - http://web.stanford.edu/class/cs224n/
[4] https://research.googleblog.com/2016/09/a-neural-network-for-machine.html
[5] https://hbr.org/2016/01/sentiment-analysis-can-do-more-than-prevent-fraud-and-turnover
[6] Scrapy Class to create a crawl process