Sentiment Analysis

Papadouli Vasiliki

February 20, 2022

1 Introduction

Natural Language Processing, is broadly defined as the automatic manipulation of natural language, like speech and text, by software.More specifically,the area of sentiment analysis , which is also known as opinion mining,falls into the broad category of text classification tasks where you are supplied with a phrase, or a list of phrases and your classifier is supposed to tell if the sentiment behind it is positive, negative or neutral. In this project the main goal is to implement different machine learning and deep learning algorithms in two datasets scraped by Trip Advisor website to classify hotel and restaurant reviews as positive or negative. This report is organised as follows: Section 2 describes assignment's datasets, Section 3 describes data preprocessing, Section 4 contains the description of machine learning and deep learning algorithms, Section 5 contains the evaluation report.

2 Data Description

The first dataset(TA_restaurants_curated.csv) has been produced by scraping Trip Advisor website and it contains information about restaurants for 31 cities in Europe. The scraper reads restaurants listings pages for each city and creates the dataset. It consists of 125.527 entries and 11 columns(Name, City, Cuisine Style, Rankin, Rating, Price Range, Number of reviews, Reviews, Detailed restaurant page (URL_TA), Identification of the restaurant inside the dataset (ID_TA)). However, we only kept reviews of restaurants for specific region of Greece (Athens). This dataset was retrieved by kaggle website in the link below: `https://www.kaggle.com/damienbeneschi/krakow-ta-restaurans-data-raw`

The second dataset(greek_hotel_reviews) has also been produced by scraping Trip Advisor with our web scraper (see WebScraper.ipynd). It consists of 2 columns (Reviews and Rating) and 7.642 raws. We scrape several popular greek hotel reviews like Grande Bretagne in Athens and Apanemo Hotel and Suites in Santorini.

Entries that contain missing information for restaurants (i.e. unreviewed restaurants or unrated review) displayed in the dataset as Nan.

3 Data Preprocessing

3.1 Handling null values

- At first, we replace missing and empty review values with 'No Review'.
- Then, we drop the missing values from the Rating column.

3.2 Data Cleaning

As it stands, our ML models cannot understand raw text data, so in order to train our models on this dataset we need to build a preprocessing process for our reviews samples. For that reason we implement the following steps for preprocessing text data:

1. Remove the puntuation marks.
2. Remove stopwords.

3. Disassemble the sentences to individual words by "splitting" on spaces.

4. "Stem" samples using the porter stemmer. This consists of converting similar (inflected/derived) words such as "love" and "loving" to a common stem, "lov" effectively compressing our vocabulary.

5. Reassemble the sentence.

3.3 Vectorization TF-IDF

Word frequencies in a document contain important information about the meaning of a certain linguistic item. A simple way of encoding sentences in numerical form is using a bag-of-words model, creating dictionaries of word to word-frequency mappings for each item. TF-IDF is an abbreviation for Term Frequency Inverse Document Frequency. This is very common algorithm to transform text into a meaningful representation of numbers which is used to fit machine algorithm for prediction.

3.4 Random Oversampling

Resampling involves creating a new transformed version of the training dataset in which the selected examples have a different class distribution. Random oversampling involves randomly selecting examples from the minority class, with replacement, and adding them to the training dataset.

4 Machine Learning Algorithms

For the purpose of this assignment we worked with a wide variety of machine learning algorithm and deep learning techniques. Below we present each of them briefly.

4.1 Logistic Regression

Logistic Regression is a classification algorithm which aims to o predict a probability that the answer to some question is 1 or 0, yes or no, true or false, good or bad etc. The general goal of this algorithm is to find the decision boundary that seperates better the data. More specifically, the objective function that the algorithm aims to maximize is $\prod |h_i|$. Roughly speaking, this optimization tends to position a hyperplane in the middle of two classes that offers greatest seperation of the two classes.

4.2 K Neighbors Classifier

KNC works by finding the distances(Euclidean Distance or Manhattan Distance or Minkowski Distance) between a query and all the examples in the data, selecting the specified number examples (K) closest to the query. Then votes for the most frequent label (in the case of classification).

4.3 SVM

Support Vector Machines, like Logistic Regression, focus on finding the hyperplane that splits better the data points into two classes. However the objective in this algorithm is to maximize the margin under the constraints that all data points must lie on the correct side of the hyperplane. The optimization function for this problem is described by $max_{w,b}\gamma(w,b)$, where γ stands for the margin with parameters w and b (slope and intercept of the hyperplane respectively).

4.4 Random Forest

The random forest algorithm is made up of a collection of decision trees, and each tree in the ensemble is comprised of a data sample drawn from a training set with replacement, called the bootstrap sample.Depending on the type of problem, the determination of the prediction will vary. For a regression task, the individual decision trees will be averaged, and for a classification task, a majority vote,this step is known as aggregation.

4.5 Gaussian Naive-Bayes

Naive Bayes is a family of probabilistic algorithms that take advantage of the probabilty and the Bayes theorem to predict the tag of an instance. They are probabilistic, which means that they calculate the probability of each tag and then output the tag with the highest one. The way they get these probabilities is by using Bayes Theorem, which describes the probability of a feature, based on prior knowledge of conditions that might be related to that feature.

4.6 AutoMl

Auto ml is a AutoML framework specialized on natural language processing. Yet, it can also be used for generic classification and regression problems. auto ml uses a fixed pipeline structure with fixed data cleaning, scaling and feature selection steps followed by a modeling stage.Grid search and genetic programming are supported as CASH solvers. It is not possible to adjust the grid size or number of individuals and generations, consequently it is not possible to influence the optimization duration. Besides the possibility to tune scikit-learn estimators, auto ml provides interfaces to other popular ML libraries like TensorFlow or XGBoost. To improve the overall performance, auto ml provides the possibility to train an ensemble of algorithms.

4.7 XGBoost

XGBoost is A a Scalable System for learning tree ensembles. Practically this algorithm applies the benefits of gradient boosting in decision trees to improve speed and performance. Boosting is an ensemble method that corrects the errors and the misclassifications of existing models. New models are added until there is no further improvement.

4.8 Long short-term memory (LSTM)

The Long Short-Term Memory or LSTM network is a recurrent neural network that is trained using Backpropagation Through Time and overcomes the vanishing gradient problem. LSTMs differentiated by common RNNs by using memory blocks that are connected into layers. A block has components that make it smarter than a classical neuron and a memory for recent sequences.This capability of LSTMs has been used to great effect in complex natural language processing problems such as neural machine translation.

4.9 BERT

Bidirectional Encoder Representations from Transformers or BERT for short, is o new method of pre-trained language representations that covers a wide variety of natural language processing tasks and leads to state-of-the-art results.

BERT, as we mentioned before, is a method of pre-training language representations, this means that a general-purpose "language understanding" is trained in a large corpus and then is used to fulfill NLP tasks. BERT outweighs previous methods because it is the only deeply bidirectional system for NLP tasks. Previous methods are either context-free or unidirectional contextual. Bidirectional contextual models instead generate a representation of each word that is based on the other words in the sentence.

BERT has two phases: pre-training and fine-tuning. Pre-training is a one-time procedure while fine-tuning is repeated for every task individually.

5 Experimental Evaluation

5.1 Restaurant reviews dataset

Algorithms	Accuracy	0 -	1 - F1-score
LR	0.91	0.10	0.95
KNN	0.97	0.13	0.99
RF	0.91	0.09	0.96
Naive Bayes	0.83	0.08	0.91
SVC	0.93	0.05	0.96
LSTM	0.98	0.00	0.99
AutoML	0.91	0.10	0.95
Xgboost	0.92	0.07	0.96
BERT	0.98	0.00	0.99

PREDICTION ACCURACY OF THE MACHINE LEARNING AND DEEP LEARNING MODELS USED

From the table above, we can conclude that the deep learning algorithms, Long Short Term Memory (LSTM) and Bidirectional Encoder Representations (BERT) from Transformers, give better performance. However, one can easily see that in the minority class (0 target) almost every model fails to predict correctly it's instances. This can be explained by the fact that our dataset was unbalanced and we had to do some random oversampling before training our models. It is worth mentioning though, that KNN seems to have the best F1 score for class 0 and equal to 13%.

5.2 Hotel reviews dataset

Algorithms	Accuracy	0 - F1-score	1 - F1-score
LR	0.95	0.54	0.98
KNN	0.96	0.47	0.98
RF	0.96	0.41	0.98
Naive Bayes	0.88	0.31	0.94
SVC	0.97	0.50	0.98
LSTM	0.96	0.00	0.98
AutoML	0.96	0.49	0.98
Xgboost	0.96	0.53	0.98
BERT	0.98	0.00	0.96

PREDICTION ACCURACY OF THE MACHINE LEARNING AND DEEP LEARNING MODELS USED

According to the above prediction metrics, it is safely to assume that Bidirectional Encoder Representations (BERT) estimator performs better concerning overall model accuracy. However fails to predict correctly instances from minority class (0%). On the contrary, Logistic Regression and XGBoost appear a considerable percentage of correct instances in the minority class. Yet from the results above it is clear that like BERT estimator, LSTM neural network could not predict instances from 0 class.