Universidad Autónoma de Madrid

Escuela Politécnica Superior

Máster Universitario en Investigación e Innovación en Tecnologías de la Información y las Comunicaciones (I2-TIC)

Web Mining 2019-2020

Lab assignments: Sentiment Analysis

Starting date: Wednesday, 15th April 2020

Submission deadline: Wednesday, 29th April 2020

In this laboratory session we are going to further study sentiment analysis techniques related to opinion mining and how to perform natural language processing (NLP) tasks.

A special type of text mining related to NLP and very similar from detecting whether a text or a reviewed is positive or negative is the task of detecting truthfull from deceptive texts (true or false/ invented or not).

There are several research approaches that have tackled this problem. Concretely in [1] authors design an intelligent system that behaves like a classifier. The system's input are police reports to classify and the exit of the system indicates whether it was a true or false report. False reports represent an expense of money and resources to the Police. For society, the existence of false reports represent a loss of police resources and a contamination of the police databases. Despite the seriousness of this issue today is a very common crime, since, unaware of the law, some citizens present robbery allegations with different goals, e.g. to collect money from insurances on a stolen item. The early detection of this type of report allows to focus the work of police inspectors in a more efficient and prevent from wasting resources and time that can be focused on real crimes. That is why a tool that allows detecting this type of reports could be useful.

In that work authors contrasted their approach to the approach of others with similar goals. The following datasets are public and used in paper [1] to contrast the success of the approach:

- **Positive Sentiment Hotel Opinions** This dataset, first introduced in [2], and used in [2,5,7] and [3] , is comprised of 400 deceptive and 400 truthful positive reviews on 20 hotels in the Chicago area. [1]'s results against the aforementioned state-of-the art approaches are shown in Table 1.
- **Positive and Negative Sentiment Hotel Opinions** This dataset is first introduced in [5] and extends the Positive Senti- ment Hotel Opinions dataset, by introducing 400 deceptive and 400 truthful negative reviews on 20 hotels in the Chicago area. Some authors consider the positive and negative dataset separately [3,5,7] (see Table 2), while others jointly [9,37] (see Table 3).
- **Abortion Opinions** This dataset, introduced and used in [4], presents 100 true and 100 deceitful opinions on the topic of abortion. [1]'s results against its original methodology are shown in Table 4.
- **Death Penalty Opinions** This dataset, introduced and used in [4], presents 100 true and 100 deceitful opinions on the topic of death penalty. [1]'s results against its original methodology are shown in Table 5.
- **Best Friend Opinions** This dataset, introduced and used in [4], presents 100 true and 100 deceitful opinions on the topic of death penalty. [1]'s results against its original methodology are shown in Table 6.
- **Extended Hotel, Restaurant, and Doctor Opinions** This dataset, first introduced in [6] , presents 940 true and 940 deceitful opinions on hotels (extending the Positive and negative Sen- timent Hotel Opinions dataset), 520 true and 200 deceitful opinions on restaurants, and 400 true opinions on doctors, obtained from different sources (i.e.,

Turkers, experts and customers). The three domains are considered separately in [6] (see Tables 7 –9), and jointly in [8] (see Table 10).

The following tables were reported in paper [1], that report the classification results (using different machine learning algorithms). When no paper is indicated it means that the approach followed is that of paper [1] when another paper is reported in the left it means that the result corresponds to the success measure reported in the corresponding paper using the approach described in that paper.

Table 1
Performance statistics of our models on the Positive Sentiment Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.975 (0.9617, 0.9847)	0.9800	0.9700	0.9703	0.9751	0.9955
SVM	0.9688 (0.9542, 0.9797)	0.9750	0.9625	0.9630	0.9689	0.9915
Random Forests	0.8475 (0.8207, 0.8717)	0.8600	0.8350	0.8390	0.8494	0.9326
Naive Bayes	0.8925 (0.8689, 0.9131)	0.8775	0.9075	0.9046	0.8909	0.9414
[2]	0.898	–	–	–	–	–
[5]	0.893	–	–	–	–	–
[3]	–	–	–	–	0.89	–
[7]	–	–	–	–	0.902	–

Table 2
Performance statistics of our models on the Negative Sentiment Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.95 (0.94, 0.97)	0.95	0.96	0.96	0.95	0.99
SVM	0.95 (0.94, 0.97)	0.95	0.95	0.95	0.95	0.99
Random Forests	0.82 (0.80, 0.85)	0.84	0.80	0.81	0.83	0.91
Naive Bayes	0.87 (0.84, 0.89)	0.85	0.88	0.88	0.87	0.91
[5]	0.86	–	–	–	–	–
[3]	–	–	–	–	0.865	–
[7]	–	–	–	–	0.872	–

Table 3
Performance statistics of our models on the Positive and Negative Sentiment Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.95 (0.93, 0.96)	0.95	0.95	0.95	0.95	0.99
SVM	0.95 (0.93, 0.96)	0.94	0.95	0.95	0.95	0.98
Random Forests	0.85 (0.83, 0.88)	0.86	0.85	0.85	0.86	0.93
Naive Bayes	0.88 (0.85, 0.90)	0.85	0.90	0.90	0.87	0.94
[5]	0.872	–	–	–	–	–
[3]	–	–	–	–	0.879	–

Table 4
Performance statistics of our models on the Abortion Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.80 (0.73, 0.85)	0.87	0.72	0.76	0.81	0.85
SVM	0.76 (0.69, 0.81)	0.82	0.69	0.73	0.77	0.84
Random Forests	0.73 (0.66, 0.79)	0.73	0.73	0.73	0.73	0.79
Naive Bayes	0.68 (0.61, 0.74)	0.93	0.42	0.62	0.74	0.83
[4]	0.70	–	–	–	–	–

Table 5
Performance statistics of our models on the Death Penalty Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.73 (0.67, 0.80)	0.81	0.66	0.71	0.75	0.61
SVM	0.73 (0.67, 0.80)	0.81	0.66	0.71	0.75	0.81
Random Forests	0.73 (0.66, 0.79)	0.81	0.65	0.70	0.75	0.63
Naive Bayes	0.61 (0.54, 0.68)	0.92	0.30	0.57	0.70	0.62
[4]	0.674	–	–	–	–	–

Table 6
Performance statistics of our models on the Best Friend Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.90 (0.85, 0.94)	0.90	0.91	0.91	0.90	0.97
SVM	0.90 (0.85, 0.94)	0.90	0.91	0.91	0.90	0.95
Random Forests	0.78 (0.72, 0.84)	0.74	0.82	0.80	0.77	0.86
Naive Bayes	0.82 (0.76, 0.87)	0.69	0.95	0.93	0.80	0.92
[4]	0.77	–	–	–	–	–

Table 7
Performance statistics of our models on the Extended Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.93 (0.92, 0.94)	0.94	0.92	0.93	0.93	0.98
SVM	0.92 (0.89, 0.95)	0.95	0.90	0.90	0.93	0.98
Random Forests	0.82 (0.80, 0.84)	0.83	0.80	0.81	0.82	0.90
Naive Bayes	0.85 (0.83, 0.87)	0.90	0.80	0.82	0.86	0.91
[6]	0.664	–	–	–	–	–

Table 8
Performance statistics of our models on the Restaurant Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.94 (0.91, 0.96)	0.96	0.92	0.92	0.94	0.98
SVM	0.92 (0.89, 0.95)	0.95	0.90	0.90	0.93	0.97
Random Forests	0.82 (0.78, 0.86)	0.83	0.81	0.82	0.82	0.90
Naive Bayes	0.86 (0.82, 0.89)	0.79	0.93	0.92	0.85	0.91
[6]	0.765	–	–	–	–	–

Table 9
Performance statistics of our models on the Doctor Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.98 (0.97, 0.99)	0.96	1.00	1.00	0.98	1.00
SVM	0.99 (0.98, 1.00)	0.98	1.00	1.00	0.99	1.00
Random Forests	0.94 (0.92, 0.95)	0.87	1.00	1.00	0.93	1.00
Naive Bayes	0.87 (0.84, 0.89)	0.87	0.86	0.85	0.86	0.91
[6]	0.647	–	–	–	–	–

Table 10
Performance statistics of our models on the Extended Hotel, Restaurant, and Doctor Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.95 (0.94, 0.96)	0.95	0.96	0.96	0.95	0.99
SVM	0.95 (0.94, 0.96)	0.94	0.96	0.96	0.95	0.99
Random Forests	0.84 (0.82, 0.85)	0.84	0.83	0.84	0.84	0.91
Naive Bayes	0.87 (0.85, 0.88)	0.90	0.84	0.85	0.87	0.93
[8]	–	–	–	–	–	0.907

The original papers explaining the approaches taken are included in a folder called biblio in case you want to consult them.

Assignment 1. Understanding text mining and NLP techniques

The described datasets are included in a folder named corpus. A simple NLP preprocessing pipeline that builds bags of words, and includes PosTagging, lematization, stopwords techniques, etc is given to you in a python file named FeatureExtraction. This file follows the pipeline described in [1] a produces for each given dataset 3 csv files, one with postagging information of each example in the dataset and two bag of words files to be used to classify one with positive and one with negative examples of the class to classify (truth or false).

You are requested to run this file for at least two of the given datasets. Describe the NLP pipeline executed, the steps taken, describe de bag of words files created and give at least 5 examples of how 5 different rows (cases) in the datasets are transformed step by step (by debugging the execution). You need to install the treetaggerwrapper[1][2] that is the package used for NLP tasks.

Assignment 2. Querying for entities with certain label

You are requested to implement a classifier for the two previously studied datasets. Examples in R that model initial versions of [1]'s approach are given so you can take a peep at the process. You should do your programming in python. For which you can use the well known libraries of nltk, scikit_learn or sklearn. You can either model any of the models described in the original papers or try to improve or change the methodology. Note that for example, bigrams, trigrams, word2vec, etc approaches have not been followed and you could give a try if you dare. You are asked to present a report describing your approach and a comparison table where you report the previously obtained results in that dataset (i.e. use the data from the tables above) against your results.

Deliverable

The deliverable of the previous lab assignments is an inform as requested in Assignment 1, the truth tables comparing your results against the original given tables along with the explanation of the used dataset and technique requested in Assignment 2 and your source code (without external libraries),. Put your name, surnames and email in the header of the file.

Submit it via Moodle within a zip file named **`wm1718-lab-NLP-XX.zip`**, where XX has to be replaced accordingly with the team id, e.g., 01, 02, ...

Grade

The grade of the lab assignments will be computed as follows:

- Assignment 1: up to 4 points.
- Assignments 1 and 2: up to 10 points

Bibliography

[1] Quijano-Sánchez, L., Liberatore, F., Camacho-Collados, J., & Camacho-Collados, M. (2018). Applying automatic text-based detection of deceptive language to police reports: Extracting behavioral patterns from a multi-step classification model to understand how we lie to the police. Knowledge-Based Systems, 149, 155-168.

[2] M. Ott , Y. Choi , C. Cardie , J.T. Hancock , Finding deceptive opinion spam by any stretch of the imagination, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, Association for Computational Linguistics, 2011, pp. 309–319 .

[3] L.C. Cagnina, P. Rosso , Detecting deceptive opinions: intra and cross-domain classification using an efficient representation, Int. J. Uncertain. Fuzziness Knowl. Based Syst. 25 (Suppl. 2) (2017) 151–174.

---

[1] https://treetaggerwrapper.readthedocs.io/en/latest/

[2] https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[4] R. Mihalcea , C. Strapparava , The lie detector: explorations in the automatic recognition of deceptive language, in: Proceedings of the International Joint Conference on Natural Language Processing, AFNLP, 2009, pp. 309–312.

[5] M. Ott , C. Cardie , J.T. Hancock , Negative deceptive opinion spam., in: Proceedings of the HLT-NAACL, 2013, pp. 497–501 .

[6] J. Li , M. Ott , C. Cardie , E. Hovy , Towards a general rule for identifying deceptive opinion spam, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 1, 2014, pp. 1566–1576 .

[7] D. Hernández Fusilier , M. Montes-y Gómez , P. Rosso , R. Guzmán Cabrera , Detection of opinion spam with character n-grams, in: Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, Springer, 2015, pp. 285–294 .

[8] B. Heredia, T.M. Khoshgoftaar, J.D. Prusa, M. Crawford, An investigation of ensemble techniques for detection of spam reviews, in: Proceedings of the IEEE International Conference on Machine Learning and Applications, ICMLA, 2016, pp. 127–133.