Sentiment Analysis in the Rest-Mex Challenge

Jessica-Alejandra Castillo-Montoya, Jonathan-Fernando Gómez-Pérez, Tania Rosales-Onofre, Marco-Antonio Torres-López, and Omar J. Gambino(✉)

Escuela Superior de Cómputo, Instituto Politénico Nacional, ESCOM-IPN, J.D. Batiz e/M.O. de Mendizabal s/n, 07738 Mexico City, Mexico
jjuarezg@ipn.mx

Abstract. In this paper, we describe our participation in the Rest-Mex 2022 forum for the Sentiment Analysis task. The objective of the task was to create a model capable of predicting the polarity of the sentiment expressed by a tourist's opinion, as well as the type of attraction visited. For this task, we followed two different approaches: a lexicon-based approach and a Machine Learning approach. In the lexicon-based approach, we use a dictionary with words that have a numerical value that specifies the association with some emotions or attractions. We trained a logistic regression model for the Machine Learning approach to predict sentiment polarity and attractions. Our proposal obtained a combined score for both tasks of 0.85, which is only 0.03 away from the best reported result.

Keywords: Sentiment analysis · Emotion lexicon · Machine learning

1 Introduction

Online platforms have allowed people to share opinions with other users about their experiences. In opinions, users express polarity about certain topic in the form of likes or dislikes, agreement or disagreement. These opinions are a valuable resource for many economic activities, because opinions can influence people's decisions [1]. Tourism is one of these economic activities in which opinions are important because people tend to express the experience they had when they visited a place, which may impact people interested in visiting the same site.

Given the above, efforts have been made to propose models that can automatically analyze opinions and determine the polarity expressed by users. Two main approaches have been followed to determine sentiment polarity: lexicon-based and Machine Learning.

Lexicon-based approach are usually based on lexical resources like sentiment lexicons, which are a list of words with sentimental attachment. Taboada et al.

O. Pichardo Lagunas et al. (Eds.): MICAI 2022, LNAI 13613, pp. 137–147, 2022.
https://doi.org/10.1007/978-3-031-19496-2_11

[2] created dictionaries of words annotated with their semantic orientation or polarity for classifying the polarity of different users' reviews. Each word in those reviews was compared to the words in the dictionaries in order to find a match; if they matched, the polarity of the words was used to determine the global _olarity of the review. In [3] a lexicon for determining sentiment polarity in Urdu language was used. The authors classify opinions as positive, negative or neutral with 89.03% of accuracy. Authors in [4] proposed a method for sentiment analysis considering aspects in opinions. They used two methods to generate lexicons for aspect-based problems—using a statistical method and a genetic algorithm—and obtained an improvement of 7.4% points of F-measure when compared with baseline method reported in [5].

The Machine Learning approach considers the task a classification problem, where classes are the polarities of the expressed opinions (i.e., positive or negative opinions). Algorithms are used to learn from data examples and then apply the learned model to unseen data. In [6] Naïve Bayes, Maximum Entropy and Support Vector Machine algorithms were used to classify sentiment polarity on a corpus of movie reviews in English. Even though the experiments obtained 82% of accuracy, the authors pointed out that the applied algorithms were not able to achieve results comparable to those reported for standard topic-based categorization, concluding that sentiment polarity is a more difficult problem than text categorization. On the other hand, hybrid approaches propose using lexicons and Machine Learning methods to classify sentiment polarity. In [7] the authors used two Spanish emotion lexicons combined with a Naïve Bayes classifier. The features provided by the lexicons allowed the classifier to increase the baseline accuracy, demonstrating that the combination of both approaches can lead to better performance.

In order to encourage the develop of computational models for Natural Language Processing in Spanish, IberLEF@sepln 2022[1] proposed an evaluation forum called Rest-Mex. The forum stated that using Machine Learning and Natural Language Processing can help promote tourism by generating mechanisms to identify the polarities of tourists' opinions. This paper describes our participation in the Rest-Mex forum for the Sentiment Analysis task. We developed two models, one using a lexicon and the other using a Machine Learning method. In the following section, the corpus and task are described (Sect. 2); then the method used is explained (Sect. 3); after that, we present the experiments and results (Sect. 4); and finally, conclusions and future work are discussed (Sect. 5).

2 Corpus and Task Description

The Rest-Mex forum provided a corpus for training models. There are 30,212 opinions, and the structure of the content is as follows:

- Title. Title of the opinion.
- Opinion. Opinion expressed by the user.

[1] https://sepln2022.grupolys.org/eventos/.

– Polarity. Sentiment polarity of the opinion.
– Attraction. Place visited by the user.

The corpus was collected from tourists who shared their opinion on TripAdvisor between 2002 and 2021.

The forum had two objectives. The first was to predict the polarity of opinion expressed by tourists traveling to a place in Mexico. The second objective was to predict the type of place visited by tourists (attractions). Polarity can have the following values: Very negative (VN), Negative (N), Neutral (NEU), Positive (P), Very positive (VP). The places visited by the tourist can be Hotel (H), Restaurant (R), and Attractive (A).

For the contest, the corpus was divided into a training set with 30,212 opinions and a test set with 12,938 opinions. Figure 1 shows the class distributions for both tasks, sentiment polarity and attractions in the training corpus[2]. As can be seen, the class distribution for sentiment polarity is unbalanced, with almost 69% of the opinions labeled as very positive. On the other hand, the distribution of attractions is also unbalanced, but to a lesser extent.

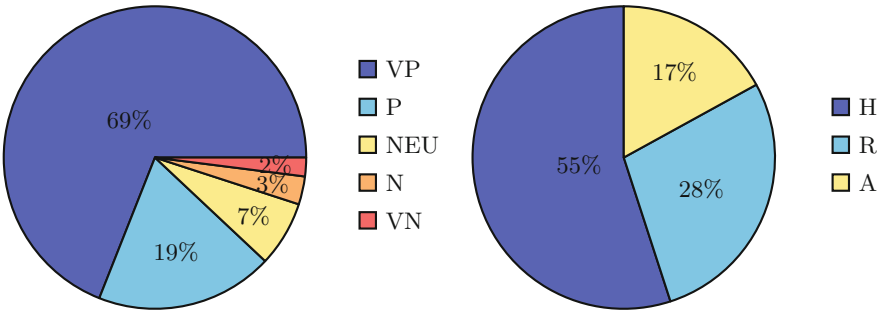

Fig. 1. Class distribution of sentiment polarity (left figure), class distribution of attractions (right figure)

Machine Learning methods have problems with unbalanced classes and this problem affects performance. This situation is described in Sect. 4.

3 Method

We used two approaches in the contest. One is based on a lexicon, and the other is based on Machine Learning algorithms. In this section, both approaches are described.

[2] Distributions of classes in the test set were not provided by the forum organizers.

3.1 Lexicon-Based Approach

To determine sentiment polarity we can use a dictionary of words attached to emotions. For English, numerous lexicons have been created over the years, for instance: SentiWordnet [8], OpinionFinder [9], Harvard inquirer [10] and LIWC [11]. Some English lexicons have been translated to Spanish and used for sentiment analysis in this language [8,10]. Authors in [12] created a dictionary composed of 2,036 words called Spanish Emotion Lexicon (SEL). For every word, the dictionary calculates the probability factor for affective use (FPA for its acronym in Spanish). This value indicates how often a word is used to express the following emotions: Joy, Surprise, Anger, Fear, Disgust, and Sadness. A word can be related to more than one emotion. This lexicon was selected for determining the polarity of opinions.

We follow a procedure based on the algorithm described in [7] to calculate the polarity. The first step was to tokenize the opinions; then, the text was lemmatized using Freeling [13]. The pseudocode to determine the sentiment polarity is described in Algorithm 1.

Algorithm 1: Algorithm to determine sentiment polarity using SEL

```
1  function getSentimentPolarity (o, SEL, PT);
   Input  : o is the opinion of a TripAdvisor user, SEL is the Spanish
            Emotion Lexicon, PT is the threshold defined to determine the
            sentiment polarity
   Output: sentiment polarity
2  words = getOpinionWords(o);
3  accumulatedFPAPositive = 0;
4  accumulatedFPANegative = 0;
5  foreach word in words do
6      if word in SEL then
7          fpaValues, emotion = getFPAValues(word);
8          if emotion in positiveEmotions then
9              accumulatedFPAPositive = accumulatedFPAPositive +
               fpaValues;
10         else
11             accumulatedFPANegative = accumulatedFPANegative +
               fpaValues;
12
13
14 end
15 emotionDifference = accumulatedFPAPositive - accumulatedFPANegative;
16 sentimentPolarity = getSentimentPolarity (emotionDifference, PT);
```

As can be seen, the pseudocode takes an opinion, the lexicon SEL, and a polarity threshold. Words in the opinion are separated and looked up in the lexicon. If the word is founded, the FPA values are obtained for the related

emotions (Joy, Surprise, Anger, Fear, Disgust, and Sadness). These values are accumulated for each word. Then, we obtain the difference between the FPA values of positive and negative emotions. For this, we consider the emotions Joy and Surprise as positive and the rest as negative. Finally, the sentiment polarity is obtained using a threshold. The threshold establishes the value ranges that the differences in emotions can have to be considered in the five possible polarities. The threshold values were empirically determined and using evolutionary computation. Both procedures are described in Sect. 4.

For the second task—places visited by tourists—we do not use the lexicon approach, so the description of the method used can be found below.

3.2 Machine Learning Approach

Sentiment analysis can be tackled as a text classification problem. A classifier uses a labeled dataset to train a model that learns from the data. In the Rest-Mex corpus each opinion is labeled with five different sentiment polarities, and these are considered the classes that the model should predict. As with the lexicon-based procedure, the opinions were tokenized and lemmatized. No stop words were removed. The text must be represented appropriately so that Machine Learning methods can use it. We tried different text representation like bag of words and TF-IDF. These representations were used to train a Logistic Regression classifier. Details of the experiments are described in Sect. 4.

A similar procedure was followed to determine the places visited by tourist, but the classes were the three different attractions (i.e., Hotel, Restaurant and Attractive) considered in the corpus. The following Section describe the experiments performed for this task.

4 Experiments and Results

We created a development set from the training corpus to perform the experiments. The development set had 80% (24,170 opinions) of data for training and 20% (6,042 opinions) for testing, instances of both sets were randomly selected. The corpus has the attributes title and opinion related to the sentiment expressed by users, so we concatenated them into a sentence and used it to determine sentiment polarity

4.1 Experiments with the Lexicon-Based Approach

As was explained in Sect. 3.1, our method uses the emotion difference between positive and negative emotions to determine sentiment polarity. The experiments performed are explained below.

Empirical Threshold Adjustment. Algorithm 1 uses different threshold values to determine sentiment polarity. The ranges of initial values were defined experimentally. Subsequently, information from the confusion matrix was used to determine the classes that generated the most errors and, based on this, the thresholds were modified. We tested with different ranges of values and calculated the accuracy obtained with the test set. We changed the ranges until the accuracy no longer improved. In Table 1 we show the results of the experiments. The difference (df) between positive and negative emotions (see line 13 of Algorithm 1) was used to determine different values of the sentiment polarity. As can be seen, experiment 3 obtained the best accuracy.

Table 1. Results of empirical threshold adjustment

Experiment	Threshold	Sentiment polarity	Accuracy
1	$df \geq 1$	VP	0.61
	$0.5 \leq df < 1$	P	
	$-1 \leq df < 0.5$	NEU	
	$-2 \leq df < -1$	N	
	$df < -2$	VN	
2	$df \geq 1$	VP	0.62
	$-1.3 \leq df < 1$	P	
	$-1 \leq df < -1.3$	NEU	
	$-2.6 \leq df < -1$	N	
	$df < -2.6$	VN	
3	$df \geq 0$	VP	0.67
	$-1.3 \leq df < 0$	P	
	$-1.8 \leq df < -1.3$	NEU	
	$-2.6 \leq df < -1.8$	N	
	$df < -2.6$	VN	
4	$df \geq 0.5$	VP	0.66
	$-0.7 \leq df < 0.5$	P	
	$-1.8 \leq df < -0.7$	NEU	
	$-2.6 \leq df < -1.8$	N	
	$df < -2.6$	VN	
5	$df \geq 0.5$	VP	0.66
	$0 \leq df < 0.5$	P	
	$-1.8 \leq df < 0$	NEU	
	$-2.6 \leq df < -1.8$	N	
	$df < -2.6$	VN	

Threshold Adjustment Using Evolutionary Algorithms. The adjustment of threshold values can be considered an optimization problem. Evolutionary algorithms have been used to solve optimization problems with good results [14]. The advantage of this type of algorithms is that the tuning process automatically tries different threshold values that improve accuracy instead of the manual tuning performed in previous experiments. There are several evolutionary algorithms such as particle swarm optimization, ant colony optimization, and genetic algorithm. Specifically, for the sentiment analysis task, evolutionary algorithms have been used for creating adaptive sentiment lexicons [15]. In [16], the authors used particle swarm optimization to label the words of a lexicon. In this work, we decided to use a genetic algorithm.

The genetic algorithm is inspired by Charles Darwin's theory of natural evolution. This theory establishes the survival of the fittest individual. The main elements of genetic algorithms are chromosome representation, fitness selection and operators [17]. For the implemented genetic algorithm[3] we set the following parameters.

- Number of generations: 50
- Crossover type: single point
- Mutation type: random

In Table 2 we show the results of the experiments. As can be seen, we obtained a 2% improvement in accuracy compared to the empirical approach. The threshold values were all negative, implying that the accumulated positive values are less than the negative ones. We consider this because the words used in the comments are more likely to match a negative emotion since there are four possible ones, while the positive ones are only 2.

Table 2. Results of genetic algorithm threshold adjustment

Threshold	Sentiment polarity	Accuracy
$df \geq -0.1$	VP	0.69
$-1.09 \leq df < -0.1$	P	
$-2.68 \leq df < -1.09$	NEU	
$-3.37 \leq df < -2.68$	N	
$df < -3.37$	VN	

4.2 Experiments with the Machine Learning Approach

We tested three text representations for the Machine Learning approach: bag of words with word frequency, binarized bag of words (presence or absence of

[3] We thank Gustavo-Alain Peduzzi-Acevedo, Edgar-Josue Varillas-Figueroa, Juan-Daniel Del-Valle-Pérez and Francisco-Javier Aragón-González for their help in implementing this algorithm.

a word), and TF-IDF. We used logistic regression as a classifier. After several experiments, the binarized version of the bag of words was selected because it obtained the best results.

In Fig. 1, we show that the class distribution of sentiment polarity is imbalanced. Of the five polarity classes in the corpus, 69% of the opinions have the VP polarity, while 31% of the remaining opinions have one of the other four classes. This situation usually affects the learning process of classifiers because the algorithms are biased toward the majority class examples while the minority classes are not well modeled [18]. Some algorithms help classifiers to deal with unbalanced data sets. Resampling methods—like undersampling and oversampling—are one the most used for this purpose [19]. Undersampling reduces the data by eliminating instances belonging to the majority class while oversampling replicates or generates new instances belonging to the minority class. In our experiments, we tested both resampling methods, and undersampling was selected because it obtained the best results. The resampling methods were implemented using the Imbalanced learn library [20].

The logistic regression classifier using the selected text representation and resampling method obtained 0.74% accuracy. Compared to the lexicon-based approach, the Machine Leaning approach had a 5% improvement; therefore, this model was selected for use with the test set. However, it is important to mention that the comparison is unfair because the classifier takes advantage of training examples while the first approach does not use this information.

The second task, which consists of determining the destination visited by the tourist, was also treated as a classification problem. The opinions were also tokenized and lemmatized. The selected text representation was a binarized bag of words. As shown in Fig. 1, the class distribution of attractions is unbalanced. Resampling methods did not improve accuracy and therefore were not used in the final model. We believe that resampling methods did not help because there are fewer classes (3) in the opinion polarity (5) and, in addition, the imbalance is smaller between classes. We used a Logistic Regression classifier and obtained 97% accuracy.

The contest rules allowed for two runs in the test set. We decided to create two versions of the trained model by making slight variations. Specifically, we changed the number of instances removed by the subsampling algorithm. We tried to generate a more balanced corpus in the first run by removing more instances labeled with the majority class. In contrast, in the second run, fewer instances were removed to reduce the imbalance but trying to preserve a similar distribution.

The final models for both tasks were used in the test set composed of 12,938 opinions. In the sentiment polarity task, the best run of our model obtained 73.52% of accuracy and 96.39% for the attraction prediction. In Table 3 we show the results of all participants of the contest. The results of our model are marked in bold. As can be seen, the second run that removed fewer instances had better performance than the first run in which more instances were removed. We believe eliminating instances to balance the corpus is helpful but may be too restrictive.

Forcing the corpus to be fully balanced does not allow the model to learn from the natural distribution of classes. On the other hand, reducing the imbalance to a lesser extent reduces the impact of bias in the classifier but preserves classes with a higher presence from which the trained model can learn.

Table 3. Official results of Rest-Mex 2022

Team	Final rank	Polarity acc	Attraction acc
UMU-Team-Run-1	0.8923	75.9854	98.9642
UC3M-Run1	0.8907	76.2523	98.8481
CIMAT MTY-GTO-Run1	0.8899	75.7845	98.8406
MCE_Team-Run2	0.8891	75.8503	98.8790
MCE_Team-Run1	0.8870	76.2909	98.6239
UMU-Team-Run-2	0.8855	74.1536	98.8483
GPI_CIMAT-Run1	0.8854	75.7072	98.1913
CIMAT2020_beto-Run1	0.8826	75.9740	97.8432
DCI-UG-Run1	0.8753	75.7690	96.5527
UCI-UC-CUJAE-Run2	0.8721	74.5284	97.9050
UCI-UC-CUJAE-Run1	0.8691	73.6858	97.4412
CIMAT2020_botextautoaugment-Run2	0.8690	73.9795	97.8432
DCI-UG-Run2	0.8662	74.6096	96.5527
ESCOM-IPN-IIA_run2	**0.8596**	**73.5275**	**96.3904**
GPI_CIMAT-Run1	0.8442	75.0734	92.4640
ESCOM-IPN-LCD_run2	0.8400	69.2456	94.7364
ESCOM-IPN-IIA_run1	**0.8341**	**72.9247**	**92.9741**
UPTC_UDLAP-Run1	0.8273	67.6147	96.5527
SENA Team	0.8029	65.2882	93.1133
DevsExMachina-Run1	0.7035	64.9868	82.021
DevsExMachina-Run1	0.6668	56.2528	84.6885
ESCOM-IPN-LCD_run1	0.5956	49.8144	67.1896
UPTC_UDLAP-Run2	0.5422	58.6489	47.4339
Majority class (baseline)	0.4568	70.0262	54.8771

According to the results in Table 3, our model ranked 14th out of 24[4] The final rank was calculated with a metric that combines results of both tasks; details of this metric can be found in the official web page[5]. As can be seen,

[4] Results were published in the official web page https://sites.google.com/cicese.edu.mx/rest-mex-2022/results?authuser=0.

[5] https://sites.google.com/cicese.edu.mx/rest-mex-2022/data-and-evaluation?authuser=0.

the difference between our model and the best-ranked one was only 0.03. We consider that, despite the simplicity of our model, it was very competitive.

5 Conclusions and Future Work

In this paper we reported our participation in the Rest-Mex forum. We explored two approaches for the sentiment polarity task. The first approach used a lexicon to determine five different polarities based on a threshold. Values of the threshold were calculated experimentally and using a genetic algorithm. The second approach used a Machine Learning method to classify polarity of the opinions. The latter approach performed best in the development set and was chosen for use in the test set. The same approach was used for determining the kind of place visited by the tourist. Our results placed us 14th out of 24 in the competition, with a difference of only 0.0327 points compared to first place. For future work, we propose the use of other evolutionary algorithms to improve the lexicon-based approach, as well as the use of a hybrid approach combining lexicon-based and Machine Learning methods. Further research on the use of methods for dealing with class imbalance is also proposed as well as the use of Deep Learning techniques.

Acknowledgments. We thank the support of Insituto Politécnico Nacional (IPN), ESCOM-IPN, SIP-IPN projects numbers: SIP-20220620, SIP-2083, SIP-20220925 COFAA-IPN, EDI-IPN and CONACyT-SNI.

References

1. Cheung, C.M., Lee, M.K., Rabjohn, N.: The impact of electronic word-of-mouth: The adoption of online opinions in online customer communities. Internet Res. (2008)
2. Taboada, M., Brooke, J., Tofiloski, M., Voll, K.D., Stede, M.: Lexicon-based methods for sentiment analysis. Comput. Linguist. **37**, 267–307 (2011)
3. Mukhtar, N., Khan, M.A.: Effective lexicon-based approach for Urdu sentiment analysis. Artif. Intell. Rev. **53**, 2521–2548 (2020)
4. Mowlaei, M.E., Abadeh, M.S., Keshavarz, H.: Aspect-based sentiment analysis using adaptive aspect-based Lexicons. Expert Syst. Appl. **148**, 113234 (2020)
5. Hu, M., Liu, B.: Mining opinion features in customer reviews. In: AAAI 2004, pp. 755–760. AAAI Press (2004)
6. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs Up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, vol. 10, Association for Computational Linguistics (2002)
7. Gambino, O.J., Calvo, H.: A comparison between two Spanish sentiment lexicons in the twitter sentiment analysis task. In: Montes-y-Gómez, M., Escalante, H.J., Segura, A., Murillo, J.D. (eds.) IBERAMIA 2016. LNCS (LNAI), vol. 10022, pp. 127–138. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47955-2_11

8. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: LREC, European Language Resources Association (2010)
9. Wilson, T., et al.: OpinionFinder: a system for subjectivity analysis. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP-2005) Companion Volume (software demonstration) (2005)
10. Stone, P.J.: The General Inquirer: A Computer Approach to Content Analysis. The MIT Press, Cambridge (1966)
11. Tausczik, Y.R., Pennebaker, J.W.: The psychological meaning of words: LIWC and computerized text analysis methods. J. Lang. Soc. Psychol. **29**, 24–54 (2010)
12. Rangel, I.D., Guerra, S.S., Sidorov, G.: Creación y evaluación de un diccionario marcado con emociones y ponderado para el español. Onomázein **29**, 31–46 (2014)
13. Padró, L., Stanilovsky, E.: FreeLing 3.0: towards wider multilinguality. In: Proceedings of the Language Resources and Evaluation Conference, Istanbul, Turkey, ELRA (2012)
14. Bartz-Beielstein, T., Branke, J., Mehnen, J., Mersmann, O.: Evolutionary algorithms. WIREs Data Min. Knowl. Disc. **4**, 178–195 (2014)
15. Keshavarz, H., Abadeh, M.S.: ALGA: adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs. Knowl. Based Syst. **122**, 1–16 (2017)
16. Machová, K., Mikula, M., Gao, X., Mach, M.: Lexicon-based sentiment analysis using the particle swarm optimization. Electronics **9**, 1317 (2020)
17. Sourabh, K., Singh, C.S., Vijay, K.: A review on genetic algorithm: past, present, and future. Multimed. Tools App. **80**, 8091–8126 (2021)
18. Fernández, A., García, S., Galar, M., Prati, R.C., Krawczyk, B., Herrera, F.: Learning from Imbalanced Data Sets. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98074-4
19. Estabrooks, A., Jo, T., Japkowicz, N.: A multiple resampling method for learning from imbalanced data sets. Comput. Intell. **20**, 18–36 (2004)
20. Lemaître, G., Nogueira, F., Aridas, C.K.: Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning. J. Mach. Learn. Res. **18**, 1–5 (2017)