Multimedia Tools and Applications
https://doi.org/10.1007/s11042-019-07877-7

CatSent: a Catalan sentiment analysis website

Pau Balaguer[1] · Ivan Teixidó[2] · Jordi Vilaplana[2] · Jordi Mateo[2] · Josep Rius[3] · Francesc Solsona[2]

Received: 3 October 2018 / Revised: 16 April 2019 / Accepted: 10 June 2019
Published online: 05 July 2019

Abstract
In this paper we investigate, analyze and compare sentimental analysis methodologies in Catalan tweets. The main goal is to develop a high-performance Catalan classifier. There are three main steps: Catalan language preprocessing tool, classification model and corpus training. The preprocessing tool is used for cleaning and extracting features from a document (or tweet). This is a key step due to the great morphological complexity of the Catalan language. The tool will remove empty words from the text and find the roots of other words. The classification algorithm will divide the tweet into "positive" and "negative" classes. To choose the best algorithm, five models are compared: Naïve Bayes, Maximum Entropy, Support Vector Machine, Decision Tree and Neural Networks. Finally, the corpus will be used for training and testing these methods. There is no known public corpus in Catalan, so we created one using a lexicon-based approach. This work aims to enable the tools to carry out sentiment analysis studies in the Catalan language. The last step is to develop a public web service with the best classification model achieved where users will be able to check its effectiveness.

Keywords Sentiment analysis · Natural language processing · Emojis · Catalan · Twitter

1 Introduction

Sentiment analysis [34], also known as opinion mining or emotion Artificial Intelligence (AI), is defined as the research field that studies computational opinions, feelings and emotions expressed in texts. The main goal is to systematically identify, extract, quantify, and study effective states and subjective information. Sentiment analysis aims to determine the attitude of a speaker, writer or other subject with respect to some topic or the overall contextual polarity or emotional reaction to a document, interaction, event, institution, government, etc.

Sentiment Analysis can be divided into two different learning methods: lexicon-based [11, 26] and Machine Learning [6, 18, 19, 25, 41]. In the one hand, in lexicon-based approaches, a

✉ Francesc Solsona
francesc@diei.udl.cat

Extended author information available on the last page of the article

piece of text message is represented as a bag of words. Following this representation of the message, sentiment values from a tagged dictionary are assigned to all positive and negative words or phrases within the message. A combining function, such as sum or average, is applied to make the final prediction regarding the overall sentiment of the message. In the other hand, in the Machine Learning approach, two learning types can be found [23]: supervised learning (Regression, Decision Tree, Linear, Rule-based and Probabilistic Classifiers) and unsupervised learning (Clustering, Anomaly Detection, Hebbian Learning and Learning Latent variable models, as example).

Due to the remarkable tool that the Twitter social network has become to know users' opinions about a variety of subjects in real time, many studies in this area are focused on this platform. The researchers in [13] present methods, apparatus, and computer-readable media for generating a website network graph to model one or more networks of websites relevant to subject matter of interest in a category that could be helpful. [37] presents a model with the *PoliTwi* tool that detects emerging political topics in Twitter sooner than other standard information channels.

This project is focused on binary classification of Catalan tweets (up to 140 Twitter messages) using different supervised learning models. Specifically, the algorithm models are Naïve Bayes [15], Maximum Entropy [24, 29], Support Vector Machine [7], Decision Trees [40] and Neural Networks [8, 16, 27, 39]. Since it is an initial approach to Catalan sentiment analysis, only classical models have been tested. Before testing different algorithms, data must be prepared and other preprocessing methods and tools must be implemented. Accordingly, the main goal of this project is not to test all the possible sentiment analysis approaches (classical and modern).

Due to the large number of frameworks offered for AI project development, most of the models chosen can seem misleadingly easy to implement. However, Neural Network models are relatively new in this area and adapting a model for this project could prove difficult, although it would be of interest to test and compare these models with the same labeled tweet-corpus.

Moreover, this paper aims to be a first step for sentiment analysis in Catalan. The main contribution is the study of the state of the art of the Catalan morphological analysis and the tools that can be used for sentiment analysis purposes. The project also presents a Catalan labeled corpus from which other studies can start.

To date, there is no known Catalan sentiment analysis study, the main aim of this work. In this sense, an important contribution is the method presented for text (tweet) preprocessing by applying studies of Catalan vocabulary and grammar. This consists of a morphological analysis, where infected (or sometimes derived) words are reduced to their stem, base or root form to achieve better results. Next, we tackle the construction of a valid and annotated Catalan corpus. We use this corpus to compare the performance of the most popular supervised classifiers applied to such a corpus in other languages as English [6, 18, 19, 25, 41], Spanish [10, 22, 33, 35] or Arabic [1]. Then, we will be able to compare Catalan, Spanish and English sentiment analysis performance based on supervised classifiers.

This project also aims to make an initial sentiment analysis website in Catalan, using the classification model which performs best. Then, the model obtained will be integrated into a public API to be used freely as a *Python* library.

Due to the lack of an annotated corpus in Catalan, it is very important to create one valid and large enough to achieve good results. Moreover, Twitter does not tag Catalan tweets, so the sentiment analysis procedure firstly obtains tweets geolocated somewhere in Catalonia.

Then, only those written in Catalan are selected (an external Catalan detection service will be necessary). Finally, these tweets are classified using a lexicon-based approach classifier and removing all tweets that do not have a clear polarity. The first two steps have already done in the #eMOVIX [12] project, where more than 1500 million Catalan tweets were stored in a 5TByte database. Our work focuses on the lexicon-based classification step using a subset of those tweets.

All the tools, code, dictionaries and corpus used for this project are available in the project *GitHub* Repository[1] and in the *API* Repository[2].

2 Related work

Here we brie y discuss examples of past research in sentiment analysis engines. Many studies have been developed to improve classification tools and propose new ones (basically for English and Spanish). For the lexicon-based approach, A. Moreo et al. [26] proposed a system that consists of an automatic Focus Detection Module and a Sentiment Analysis Module capable of assessing user opinions on topics in news items. In [11], a set of new feature selection schemes was proposed that use a content and syntax model to learn a set of features in a review document automatically. Furthermore, [36] also presents a detailed sentiment analysis methodology for generating a first sentiment score for a first entity based on a content source.

Furthermore, many researchers have studied classification models with different scenarios. In 2012, Marilyn A. Walker et al. [41] worked on classifying stance in on-line political debate. Peter C.R et al. [19] explored the effect of using five different types of features, classifiers and corpora.

Neural Networks are also very present in some projects. In [6], machine learning-based tools for the classification of personal relationships in biographical texts and the induction of social networks from these classifications was developed and tested. They used Support Vector Machine and Neural Networks during the project. In [25], Artificial Neural Networks, Naïve Bayes and Support Vector Machine methods were used for empirical comparisons between them.

With the continuous evolution of social networks, new and improved features have been added to sentiment analysis. Recently, Emojis are a new research focus of interest. For example, in 2017, the authors in [18] used Naïve Bayes and Maximum Entropy in combination with emojis, with the aim of providing companies with a clear idea of what customers really think about their brand.

The latest research projects in sentiment analysis are basically focused on Neural Networks. Researchers in [8, 16, 27, 39] tested Neural Networks in the sentiment analysis area using such different layers and methods as Associative Neural Networks, Convolutional Neural Networks and Recurrent Neural Networks. However, the main objective of these algorithms is more expensive processes like image processing.

Moreover, there are some multilingual sentiment classification studies where different languages are supported by a single model. These approaches present a very interesting point of view since, due to the Internet, all the world is immediately connected and many different

[1] GitHup repository: https://github.com/pbalaguer19/catalan-sentiment-analysis
[2] API repository: https://github.com/pbalaguer19/catsent-api/

languages coexist. On the one hand, [30–32] are good multilingual sentiment classification approaches using different languages. On the other hand, [3] studies the multilingual sentiment analysis for Basque and Catalan hotel reviews and [4] presents a project where Catalan and Spanish tweets debating about Catalan elections are analyzed by using multilingual sentiment analysis approaches, which offers novel views on how to study and understand communication in political debates.

In [21], the authors studied sentiment polarity detection in Spanish reviews combining supervised and unsupervised approaches. Luciana Dubiau et al. [10] evaluated the performance of Naïve Bayes, Support Vector Machine, Maximum Entropy, Decision Tree and Turney Polarity (non-supervised), the best results being achieved with Bayesian Networks methods. Furthermore, Eugenio Martínez et al. [22] achieved better results with the Support Vector Machine in sentiment analysis in movie reviews (using the review punctuation as the sentiment label).

In 2015, Ferran Pla et al. [33] presented a model based on Twitter in TASS (an evaluation workshop for sentiment analysis focused on Spanish). Finally, we highlight the work done in [35], where Naïve Bayes, Support Vector Machine and Decision Tree were tested on movie reviews. All of them achieved good results.

Moreover, language processing tools can be applied to other fields and sections, such as [14] that presents a machine learning processing for automatic keyword extraction approach for personalized health activities, or [38], with the project: automatic emotion-based music classification for supporting intelligent IoT applications.

3 Methods

This section briefly describes the methodology used. The technology used during the project is explained, as are the main design and programming criteria for the web server and, finally, how the classifier models work. The corpus used, the tweet preprocessing module and the website are also described, as is the API web service.

3.1 Technology

From among the large number of frameworks offered for AI project development, *Python* 2.7 was selected as the programming language. The *Django* framework and its *RESTful API* toolkit, *Django Rest Framework*, were also used for web-service creation.

The tweet preprocessing functions (basically for cleaning and feature extraction) used a list of Catalan stopwords and the synonym corpus described in Data section 3.2. Then, the *Freeling* framework [28] was used for the morphological study.

For the classification models, *NLTK* [20] and *sci-kit learn* [5], an open source machine learning software package for Python development were used. The packages used for scientific computing and data structured design were *NumPy*[3] and Pandas[4] which provide a fast, flexible, and expressive data structure.

TensorFlow[5], an open-source software library for machine learning was also used for Neural Network development.

[3] NumPy: http://www.numpy.org

[4] Pandas: http://pandas.pydata.org

[5] TensorFlow: https://www.tensorflow.org

3.2 Data

The tweets were collected from the *#eMOVIX* [12] project database, where tweets from around the world were sampled. We specifically used the Catalan subset obtained from an external language detection service[6]. This corpus needed to be labeled, so we used a lexicon approach based in an *ML-SentiCon* [9] file, where a large list of Catalan words is classified into positive and negative labels. This file also contains a sentiment classification of the most informative emojis [17] and their polarity. Table 1 shows some examples of Catalan words already classified by *ML-SentiCon* project and Table 2 shows some examples of the ranking of emojis, labeled by [17]. Based on this, the polarity of each tweet (defined by the formula (1)), and its value is the label of the final corpus. So, the final (Catalan) corpus used for training and testing the classifier models, which is 5.29 MB in size, contains 50,000 tweets, labeled in two categories: negative (0) and positive (1). Note that the Catalan tweets were classified into a binary classification (positive or negative) due to the models tested, since there are some approaches that only support binary labels. Also note that the corpus had some limitations since its generation was based on a lexicon approach.

The Corpus quality was obtained by using 500 random tweets (250 from the positive label and 250 from the negative label). Table 3 shows the results obtained.

The AnCoraVerb [2] dictionary was used for synonym detection (see Section 3.3).

Final corpus is available on the project *GitHub* Repository[7].

3.3 Operation

Algorithm 1 (in pseudo-code) explains the operating mode for training and testing each classifier model. There are two main steps: the first one is the preprocessing of each tweet and the second involves training and testing.

Algorithm 1 Classifier.

```
function Classifier
    /* PREPROCESSING STEP */
    for each tweet, sentiment in Corpus do
        tweet = remove twitter elements(tweet)
        tweet = find synonyms(tweet)
        tweet = remove stopwords(tweet)
        tweet = remove punctuation(tweet)
        tweet = morphological analysis(tweet)
        tweetList.append(tweet, sentiment)
    end for
    /* TESTING STEP */
    Training Corpus, Test Corpus = divide corpus(tweetList)
    for each train tweet, sentiment in Training Corpus do
        classifier.train(train tweet, sentiment)
    end for
    for each test tweet, sentiment in Test Corpus do
        classifier.test(test tweet, sentiment)
    end for
    classifier.getMetrics()
end function
```

[6] Language detection service: https://detectlanguage.com

[7] Project on Github: https://github.com/pbalaguer19/catalan-sentiment-analysis

Table 1 Some examples of Catalan labeled words used for lexicon-based classification

Word	Polarity
Apreci	0.625
Afortunat	0.75
Dolc	0.55
Abatut	−0.344
Angoixar-se	−0.687
Cruel	−0.417

polarity <0 (polarity >0) is considered a negative (positive) word.

The first loop focuses on the preprocessing step. Preprocessing is responsible for modifying the input data in order to improve the performance of the classifier. This step can be divided into different tasks:

1. remove twitter elements deletes user names (@USERNAME), hash tags (#hashtag) and URLs (http://...).
2. find synonyms unifies all words with the same meaning in the same word (one of them, randomly chosen).
3. remove stopwords deletes all the words that are in the stopword corpus.
4. remove punctuation removes all punctuation marks.
5. morphological analysis (also known as stemming) reduces infected (or sometimes derived) words to their word stem, base or root form. Section 3.1 describes the stemming module used for this step.

Different data resources were used in the preprocessing step. The *AnCoraVerb* [2] project offers a corpus that can be used for the Catalan synonym detection. We modified this corpus by adding the most common misspellings. The preprocessing step also performs the stopword extraction function to delete words that do not contribute to the sentence (stopwords). To detect these, we used the *Ranks* NL[8] and LaTeL[9] corpora, putting them together in one file. Therefore, finally a dictionary with 736 Catalan stopwords which can be deleted was finally obtained.

Table 4 shows some tweets before and after pre-processing. Note that user names have been replaced by "username" to preserve their anonymity.

In the training and testing step, we first divided the cleaned corpus into two different lists (with function divide corpus). The Training Corpus represents 75% of the total (37,500 tweets) and will be used for training the classifiers. The remaining 25% was used to test the classifier accuracy and obtain its performance (Test Corpus).

Once the corpus had been divided, each classifier was first trained in order to evaluate it. After finishing the evaluation, the last step was to show the obtained performance (described in section 3.6).

Once the model with better performance was found, 100% of the corpus was used to train it. The model obtained was used in the web application and its API.

[8] Ranks NL: http://www.ranks.nl/stopwords/catalan

[9] LaTeL: http://latel.upf.edu/morgana/altres/pub/ca_stop.htm

Table 2 Some examples of labeled emojis used for lexicon-based classification

Emoji	Polarity
😂	0.221
❤️	0.746
😄	0.421
😩	-0.368
😭	-0.093
😒	-0.374

polarity <0 (polarity >0) is considered a negative (positive) emoji

3.4 Corpus creation

After the lexicon corpus had been obtained (see section 3.2), the development of a simple lexicon-based approach for the corpus classification was easy. Given a tweet, its polarity P is calculated following formula (1).

$$P = \left(\sum_i pe_i\right) + \left(\sum_j ps_j\right), \tag{1}$$

where i is the number of emojis and j is the number of words contained in the text. Therefore, pe_i represents the sentiment of the emoji [17] and ps_j is the polarity of the word based on the ML-Senticon lexicon-approach of [9].

Note that both pe and ps can be any number between −1 and 1. Negative and positive numbers mean negative or positive sentiment respectively. Due to the importance of obtaining correct classifications, all emojis with $|pe_i| < 0.2$ and words with $|ps_j| < 0.3$ were discarded. In addition, tweets with $|P| < 0.2$ were also discarded. These decisions improved the quality of the corpus.

3.5 Sentiment model

3.5.1 Naïve Bayes

The Naïve Bayesian classifier works as follows. Suppose that there is a set of training data D, in which each tuple (in our case tweet) is represented by an n-dimensional feature vector, $X = x_1, x_2, \ldots, x_n$, corresponding to n measurements of attributes or features. Assume that there are m classes, $C_1, C_2, \ldots, C_m$ (in our case, there are two classes: positive and negative, so $m = 2$). Given

Table 3 Quality of training and testing corpus

Corpus	WCC	BC	SP	OL
Tweets	96.6%	3.4%	5.4%	2.8%

Percentage of Catalan tweets well-classified (WCC), badly-classified (BC), and those in Spanish (SP) and other languages (OL).

Table 4 Examples of tweet preprocessing

Tweet	Preprocessed tweet
Poca broma, fa 10 minuts era fins es llit i ara a sa peluqueria.	broma minut llit peluqueria
@USER Vaig publicar-ho al moment en que jo hi era;) te bona pinta, eh?	publicar el moment tenir bo pinta eh
Enhorabona @USER. Molt content per tu i tambe x @USER i els seus gols.	enhorabona molt content tambe x gol
Heu d ser claus per la proxima temporada	d clau proxima temporada
Casal d'Estiu 2016 i Curs d'Informatica a #Corbins! https://t.co/fQ9Y69hJ0S	casal estiu curs informatic corbins
Ahir va ser el dia mundial de l'ELA i des d'@esquerratortosa	dia mundial donar suport
vam donar tot el nostre suport #fesungestperlela https://t.co/1uOlQUF1z8	fesungestperlela

The first column contains the input text while the second displays the preprocessing-step output.

a tuple X, the classifier will predict that X belongs to C_i if and only if: $P(C_i|X) > P(C_j|X)$, where i,j ∈ [26, 34] and $i \neq j$. $P(C_i|X)$ is defined in formula (2).

$$P(C_i|X) = \prod_{k=1}^{n} P(x_k|C_i) \tag{2}$$

3.5.2 Maximum entropy

Maximum Entropy is a discriminatory classification model where the documents are described as a list of attributes, each being a constraint model. This model is based on selecting the probability distribution that satisfies all constraints and maximizes entropy. The classification probability is obtained as follows:

- For each word w and class $C_i \in C$ there is a joint feature $f_i(C_i,w)$, defined as the number of times that w occurs as class C_i.

- Via iterative optimization, a weight (λ_i) to each joint feature so as to maxi-

mize the log-likelihood of the training data is assigned.

- Given a document X and weights, the probability of class C_i is defined in formula (3).

$$P(C_i|X) = \frac{e^{\left(\sum_i \lambda_i f_i(C_i,X)\right)}}{\sum_{c' \in C} e^{\left(\sum_i f_i\left(c',X\right)\right)}} \tag{3}$$

3.5.3 Support vector machine (SVM)

The Support Vector Machine (SVM) is a supervised binary classification method. The training of this method is to find a hyperplane[10] that separates the vectors of attributes

[10] In geometry, a hyperplane is a division of a space into two parts

that represent different datasets into groups, these being the highest possible separation. The vectors that define the limits of the maximum separation between the classes are called support vectors. The classifier is based on formula (4) to predict the most likely class given to a document.

$$f(x) = sign\left(\sum_i \alpha_i x_i \cdot x + b\right), \tag{4}$$

x being the attribute vector (or features) of the document to be classified, α_i the weights of the vector attributes, x_i each support feature and b the independent term. The value −1 indicates that the document belongs to a class and the value +1 to the other, representing the hyperplane side of x.

3.5.4 Decision trees

Supervised classification method where the training goal is to build a decision tree with multiple paths. Each node searches for the attribute information that provides higher profits for its representing class. The tree grows to its maximum size and then is pruned to improve its performance. From this model, we can add decision rules about which classifiers it will be based on.

3.5.5 Neural networks

We used a deep convolutional neural network. Our approach is based on the approaches proposed by Kim [16] following the "Implementing a CNN for Text Classification in Tensorflow" *blog* post[11] and its simplified implementation *GitHub* repository[12].

This slightly simplified implementation works as follows: the first layer embeds words into low-dimensional vectors. The next layer performs convolutions over the embedded word vectors using multiple filter sizes. For example, sliding over 3, 4 or 5 words at a time. Next, we max-pool the result of the convolutional layer into a long feature vector, add dropout regularization, and classify the result using a softmax layer.

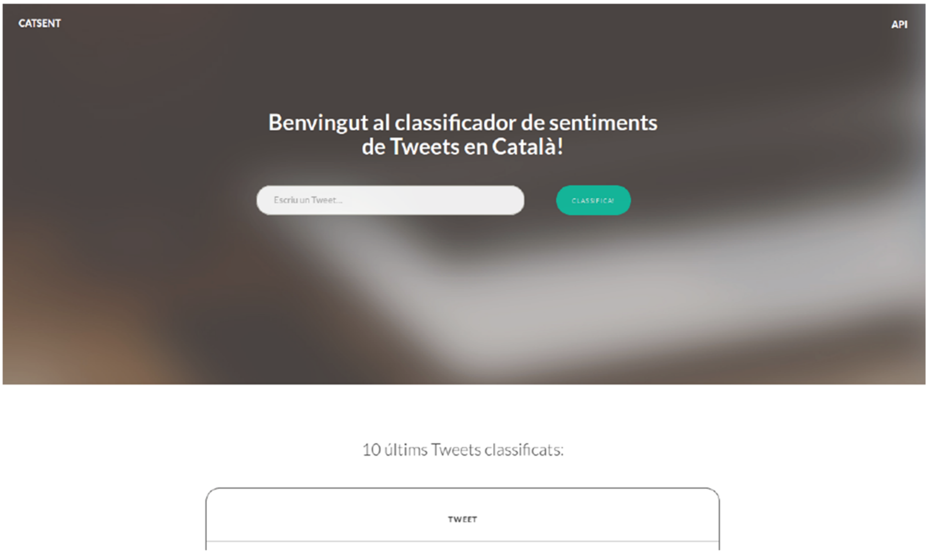
3.6 Performance metrics

The behavior of the classifiers was evaluated by calculating the Precision, Recall, and F-score of each model.

Precision measures the fraction of true positives among all positive predictions made by the method. *Recall* (or specificity) measures the fraction calculated by dividing the number of correct choices by the total number of choices available to the model. *Precision* and *Recall* are

[11] http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow

[12] https://github.com/dennybritz/cnn-text-classification-tf


Benvingut al classificador de sentiments de Tweets en Català!

10 últims Tweets classificats:

Fig. 1 CATSENT website interface

defined in formulas (5) and (6) respectively. *tp* (true positives) is the number of correctly predicted positive tweets, *fp* (false positives) is the number of negative tweets incorrectly predicted as positive and *fn* (false negatives) is the number of positive tweets that are incorrectly classified as negative. The *F-score* is an aggregate measure of the accuracy of a classifier that calculates a weighted average of *Precision* and *Recall* as defined in formula (7).

$$Precision = \frac{tp}{tp + fp} \tag{5}$$

$$Recall = \frac{tp}{tp + fn} \tag{6}$$

$$F\text{-}score = 2\frac{Precision \cdot Recall}{Precision + Recall} \tag{7}$$

Table 5 Naïve Bayes results

Naïve Bayes	Precision	Recall	F-score
Negative	81%	81%	81%
Positive	81%	82%	82%
Total	81%	82%	81%

Table 6 Maximum entropy results

Maximum Entropy	Precision	Recall	F-score
Negative	81%	80%	81%
Positive	80%	82%	81%
Total	81%	81%	81%

3.7 Web service and API

Web service will be created after testing all classifiers. This website will enable users to test the chosen model. There will be an input box where users can write a tweet and the website will return its polarity. There will also be a list with the last 10 classified tweets. Figure 1 shows the interface design of the described website.

The web service also offers an API where GET and POST HTTP methods can be called up. The GET Method returns a list of last classified tweets in JSON format. The POST method requires a tweet in the body section. The API will return the sentiment label of this tweet.

The services described are available at http://catsent.udl.cat/.

4 Results

Five models were tested and evaluated to find the best results. These were Naïve Bayes, Maximum Entropy, Support Vector Machine, Decision Tree and Neural Network.

All the frameworks used offer the chance to train and evaluate a given corpus. Therefore, the hardest job was to fit the corpus to each model since each framework required a different data format.

The tests were performed on a virtual machine with 4 Gigabytes of RAM with which the average execution time was high (about 2 h). All tests that exceeded a 10-h execution time were overlooked because they were considered too heavy. The following sections show the results and the analysis of each model.

4.1 Naïve Bayes

Due to the flexible and easy model that *NLTK* toolkit offers for Naïve Bayes (NB) classifier, it was established as the framework used for this algorithm. Table 5 shows the results obtained for this classifier.

Table 7 SVM results

SVM	Precision	Recall	F-score
Negative	74%	75%	75%
Positive	74%	73%	74%
Total	74%	74%	74%

Table 8 Decision tree results

Decision Tree	Precision	Recall	F-score
Negative	57%	97%	72%
Positive	89%	26%	41%
Total	73%	62%	46%

4.2 Maximum entropy

The *NLTK* toolkit was also used for this model, given the similarity between the Naive Bayes and Maximum Entropy (ME) algorithms. The difference between these two models is that the features are independent in Naïves Bayes but not in the Maximum Entropy. In other words, Maximum Entropy establishes a relation between the features. Table 6 shows the evaluation results.

4.3 Support vector machines

The *sci-kit learn* toolkit was established as the framework for Support Vector Machine (SVM) algorithm, and more specifically, the linear implementation was *LinearSVC*. The results of the tests are shown in Table 7. Better results were expected for this model and it is thought that the default parameters may influence these.

4.4 Decision tree

The Decision Tree (DT) model was also implemented with the *sci-kit learn* toolkit. In this case, tests were carried out for different tree depths, between 2 and 100. The highest performance scores were obtained with depth 16. The performance increased until depth 16 but then went down when the depth increased beyond 16. Table 8 shows the metrics performance percentages for this model.

4.5 Neural network

The implementation of the Neural Network was hardest due to its complexity and combinations. Finally, a *Github* repository[13] was found with a simple and editable code. Due to its simplified algorithm, the results were not as good as expected but they can be improved with more tests. The execution time in this case was around 8 h and the results are shown in Table 9.

4.6 Discussion

Table 10 shows a comparison between the classifiers studied where the Naïve Bayes and Maximum Entropy models performed best.

The performance of the Neural Network model was unexpectedly poor, since it required a long training time and the Precision was greater in the original paper in

[13] https://github.com/dennybritz/cnn-text-classification-tf

Table 9 Neural network results

Neural Network	Precision	Recall	F-score
Negative	80%	73%	77%
Positive	75%	81%	78%
Total	78%	77%	77%

Table 10 Summary of the classifier results

Classifier	Precision	Recall	F-score
Naïve Bayes	81%	82%	81%
Maximum Entropy	81%	81%	81%
SVM	74%	74%	74%
Decision Tree	73%	62%	46%
Neural Network	78%	77%	77%

English. Nevertheless, note that training and testing algorithms with tweets give less information to the model than reviews or documents as tweets only have 140 characters. Notably, some of the most informative features were emojis. The addition of emojis into social networks can greatly help the tasks of sentiment analysis, expanding the possibilities and improving the old ones.

The Naïve Bayes and Maximum Entropy methodologies are very similar and that is why their results are also close. The same framework was used for these models and this may be another reason for the similarity in performance. The SVM and Decision Tree presented the worst results. SVM gave poor results due to its configuration settings. However, the results obtained are considered good because there is no known previous study of this area in Catalan and therefore the time spent on other steps increased.

Table 11 shows a comparison between Spanish, English and the results obtained. For Spanish performances, precision from [10] was used. English column uses the best performances from [19, 27, 29, 40]. There is higher performance from Spanish and English than the ones obtained. However, these languages have been studied and tested for a long time. In our case, this is the first known study in the area of Sentimental Analysis in Catalan.

The corpus used has a low error rate due to the great restrictions (only clearly positive or negative tweets are part of it). The preprocessing step works well but takes a lot of time. The task that requires the most time is the substitution of synonyms, so it should be studied whether this is worth leaving or removing.

Table 11 Comparison between Catalan, Spanish and English precision performances

Classifier	Catalan	Spanish	English
Naïve Bayes	81%	95%	88%
Maximum Entropy	81%	95%	78%
SVM	74%	95%	95%
Decision Tree	73%	88%	80%
Neural Network	78%	–	86%

In summary, creating the corpus and input-data preprocessing tool was a good job but there is always room for improvement. In the classification models, it is convenient to test new algorithms, do more tests and compare them.

The classifier used for the web server will be the Bayesian network model since it had the best results. In addition, the model on the website will have 100% of the instances to train. So, its performance should be higher. The server will offer a graphic interface and also a RESTful[14] API with GET and POST methods.

5 Conclusions and future work

Catalan Sentiment Analysis is presented in this project. The three main steps are corpus creation, preprocessing tool development and classifier model testing. The model with the best performance metrics is used for the web application service, which offers a user-friendly interface and also a RESTful API. The main contribution of this work lies in the first public classifier in Catalan and the fact that this classifier is able to create new and better corpora for future projects. We also collected many Catalan resources for Sentiment Analysis that can be of great help for new studies.

There are three main issues that need to be solved in future work. First of all, the runtime for the tests is very long due to the preprocessing step. The second challenge is to improve the performance percentages, trying to achieve at least 90% Precision. Also, the last pending issue (and probably the hardest one) is to have a spellchecker for preprocessing since it is very common to find many misspellings. To improve the first issue, a cleaning module would only run once. Another approach might be to prioritize the preprocessing tasks and remove the lowest priority ones. The challenge of achieving a precision greater than 90% can easily be achieved with more tests, models and different corpora.

Including a spellchecker before training model classifiers could greatly help to improve performance since it is very common to find mistakes or abbreviations in some words. However, Catalan is very complex and it is difficult to obtain a good spellchecker and know the correct word. Accents, apostrophes, dialects and irregular verbs are some aspects of the high complexity of Catalan.

Since this is an initial approach to sentiment analysis in Catalan, the performance of this model should be discussed and tested in other environments. Moreover, due to the binary classification of the output sentiment, it is very difficult to find a practical application for the model. However, it is a very important initial step in order to encourage more projects in this direction.

In summary, we will continue with a deeper analysis of the corpus and the preprocessing step to try to improve its quality. Next, we will create new corpora with other methods to have more tests available. Finally, we will apply these two previous steps to different classification algorithms with better hardware and longer testing time.

Acknowledgements This work was supported by the Ministerio de Economía y Competitividad under contract TIN2017-84553-C2-2-R. IT, JV, JM, JR and FS are members of the research group 2017-SGR363, funded by the Generalitat de Catalunya. Besides, this research is partly supported by the European Union FEDER (CAPAP-H6 network TIN2016-81840-REDT). The research leading to these results has received funding from RecerCaixa.

[14] RESTful Web services are one way of providing interoperability between computer systems on the Internet.

References

1. N.A. Abdulla, N.A. Ahmed, M.A. Shehab, M. Al-Ayyoub. (2013) Arabic sentiment analysis: Lexiconbased and corpus-based. 2013 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT) IEEE
2. Aparicio J, Taule M, Mart MA (2008) AnCora-verb: two large-scale verbal lexicons for Catalan and Spanish. Proceedings of the XIII EURALEX international congress: ISBN 978–84–96742-67-3
3. Barnes J, Lambert P, Badia T (2018) MultiBooked: A Corpus of Basque and Catalan Hotel Reviews Annotated for Aspect-level Sentiment Classification. CoRR, abs/1803.08614
4. Bosco C, Lai M, Patti V, Pardo F, Rosso P (2016) Tweeting in the debate about Catalan elections. Proceedings of the tenth international conference on language resources and evaluation (LREC 2016)
5. Buitinck L et al. (2013) API design for machine learning software: experiences from the scikit learn project. arXiv:1309.0238
6. van de Camp M, van den Bosch A (2012) The socialist network. Decis Support Syst 53:761–769
7. Chen CC, Tseng YD (2011) Quality evaluation of product reviews using an information quality framework. Decis Support Syst 50:755–768
8. Chen T, Xu R, He Y, Wang X (2017) Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. Expert Syst Appl 72:221–230. ISSN 0957-4174
9. Cruz FL, Troyano JA, Pontes B, Ortega FJ (2014) ML-SentiCon: Un lexicon multilingüe de polaridades semánticas a nivel de lemas. Procesamiento del Lenguaje Natural :113–120
10. Dubiau L, Ale JM (2013) Analisis de Sentimientos sobre un Corpus en Español: Experimentación con un Caso de Estudio. ASAI :1850–2784
11. Duric A, Song F (2012) Feature selection for sentiment analysis based on content and syntax models. Expert Syst Appl 39:9166–9180
12. Feixa C, Rubio C, Ganau J, Solsona F (2015) L'Emigrant 2.0 : emigració juvenil, nous moviments socials i xarxes digitals. (Col·leccio Estudis ; 35), ISBN 9788439395348
13. Goeldi A (2011) Website network and advertisement analysis using analytic measurement of online social media content. U.S. patent no. 7,974,983
14. Huh JH (2018) Big data analysis for personalized health activities: machine learning processing for automatic keyword extraction approach. Symmetry (2018) 10(4):93
15. Kang H, Yoo SJ, Han D (2012) Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews. Expert Syst Appl 39:6000–6010
16. Kim Y (2014) Convolutional neural networks for sentence classification: arXiv:1408.5882
17. Kralj P, Smailovic J, Sluban M (2015) Sentiment of Emojis. PLoS One 10(12):e0144296
18. Kularathne SD, Dissanayake RB, Samarasinghe ND, Premalal LPG, Premaratne SC (2017) Customer behavior analysis for social media. IJAEMS 3(1). ISSN: 2454-1311
19. Lane P, Clarke D, Hender P (2012) On developing robust models for favourability analysis: model choice, feature sets and imbalanced data. Decis Support Syst 53:712–718
20. Loper E, Bird S (2002) NLTK: The Natural Language Toolkit. arXiv:cs/0205028
21. Mart n MT, Martínez E, Perea JM, Ureña LA (2013) Sentiment polarity detection in Spanish reviews combining supervised and unsupervised approaches: Expert Syst Appl
22. Mart nez E, Mart n MT, Perea JM, Urena~ LA (2011) Tecnicas de clasificacion de opiniones aplicadas a un corpus en Español. Procesamiento del Lenguaje Natural :163–170
23. Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: a survey. Ain Shams Engineering Journal: 2090–4479
24. Mehra N, Khandelwal S, Patel P (2002) Sentiment identification using maximum entropy analysis of movie reviews. Stanford University, USA
25. Moraes R, Valiati JF, Gaviao WP (2013) ~. Document-level sentiment classification: an empirical comparison between SVM and ANN. Expert Syst Appl 40:621–633
26. Moreo A, Romero M, Castro JL, Zurita JM (2012) Lexicon-based comments-oriented news sentiment analyzer system. Decis Support Syst 53:704–711
27. Nogueira dos Santos C, Gatti M (2014) Deep convolutional neural networks for sentiment analysis of short texts: international conference on computational linguistics
28. Padro L (2011) Analizadores Multilingües en FreeLing. Linguamática: ISSN 1647–0818
29. Patel D, Saxena S, Verma T (2016) Sentiment analysis using maximum entropy algorithm in big data: International Journal of Innovative Research in Science, Engineering and Technology ISSN: 2319–8753
30. Petz G et al (2012) On text preprocessing for opinion mining outside of laboratory environments. In: Huang R, Ghorbani A, Pasi G, Yamaguchi T, Yen and Neily, Jin, Beijing (eds) Active media technology, lecture notes in computer science, LNCS 7669. Springer, Berlin Heidelberg, pp 618–629

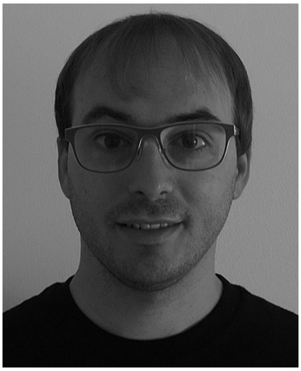
31. Petz G et al. (2013) Opinion mining on the web 2.0 - characteristics of user generated content and their impacts. Lecture notes in computer science LNCS 7947. Heidelberg, Berlin Springer :35–46
32. Petz G et al (2015) Computational approaches for mining user's opinions on the web 2.0. Inf Process Manag 51(4)
33. Pla F, Hurtado LF (2015) ELiRF-UPV en TASS 2015: Análisis de Sentimientos en Twitter. TASS :75–79
34. Qu Y, Shanahan J, Wiebe J (2004) Exploring attitude and affect in text: Theories and applications. AAAI Spring Symposium. Technical report SS-04-07. AAAI Press, Menlo Park, CA
35. Ramirez M, Carrillo M, Sanchez A (2015) Combinación de clasificadores para el análisis de sentimientos. Research in Computing Science :193–206
36. Rehling JA, Dignan TG (2013) Detailed sentiment analysis. U.S. Patent No. 8,463,595
37. Rill S, Reinel D, Scheidt J, Zicari RV (2014) Politwi: early detection of emerging political topics on twitter and the impact on concept-level sentiment analysis. Knowl-Based Syst 69:24–33
38. Seo YS, Huh JH (2019) Automatic emotion-based music classification for supporting intelligent IoT applications. Electronics (2019) 8(2):164
39. Stojanovski D, Strezoski G, Madjarov G, Dimitrovski I (2015) Twitter sentiment analysis using deep convolutional neural network: HAIS 2015, Bilbao, Spain
40. Suresh A, Bharathi CR (2016) Sentiment classification using decision tree based feature selection. IJCTA 9(36):419–425
41. Walker MA, Anand P, Abbott R, Fox JE, Martell C, King J (2012) That is your evidence?: classifying stance in online political debate. Decis Support Syst 53:719–729

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Pau Balaguer received the B.S. and M.S degrees in computer engineering from the Escola Politècnica Superior of the Universitat de Lleida (UdL), Spain, in 2017 and 2019 respectively. He gained a research position at Institut de Recerca Biomèdica de Lleida (IRBLleida). He's currently working at EURECAT as a researcher. His research interest include big data, machine learning and social network analysis.

Ivan Teixidó received the ETIS and B.Sc. in computer engineering de-grees from the Escola Politècnica Superior of the Universitat de Lleida (UdL), in 2007 and 2009 respectively. He also received M.S. in applied science to engineering from the Escola Politècnica Superior of the Universitat de Lleida (UdL), in 2011. He currently holds a position of research support staff at UdL. His research interests include cloud computing; parallelization of algorithms; social networks analysis; market analysis and big data.

Jordi Vilaplana received the B.S., M.S. and Ph.D. degrees in computer science from the Universitat of Lleida, Spain, in 2009, 2011 and 2015 respectively. Currently, he has a posdoctoral position in the Department of Computer Science at the University of Lleida (Spain). His research interest include eHealth, mHealth, cloud, parallel and distributed processing, big data and data analysis.



Jordi Mateo received the ETIG and B.Sc. in computer engineering degrees from the Escola Politècnica Superior of the Universitat de Lleida (UdL) in 2006 and 2012 respectively. He also received M.S. in computer engineering from the Escola Politècnica Superior of the Universitat de Lleida (UdL) in 2013. He received the Ph.D. degree in 2019. Currently, he has a posdoctoral position in the Department of Computer Science at the University of Lleida (Spain). His research interested and expertise is in O.R. (stochastic optimization); parallelization of algorithms; cloud computing and big data.

Josep Rius received the B.S., M.S. and Ph.D. degrees in computer science from the Universitat of Lleida, Spain, in 2006, 2008 and 2012 respectively. He also received the B.S. and M.S. of Business Administration in 2009 and 2011 re-spectively. He leaded the Research division of a Software company named ICG Software. Currently, he is an associate professor in the Department of Business Administration at Universitat de Lleida (Spain). His research interest includes parallel and distributed computing, cloud infrastructures, big data and data analysis.

Francesc Solsona received the B.S., M.S. and Ph.D. degrees in computer science from the Universitat Autònoma de Barcelona, Spain, in 1991, 1994 and 2002 respectively. Currently, he is a full professor in the Department of Computer Science at the University of Lleida (Spain). He is the cofounder of the Hesoft Group company. His research interest include distributed processing, cluster computing, coscheduling, administration and monitoring tools for dis-tributed systems, cloud computing, linear programming, big data, data analysis, social networks and bioinformatics.

Affiliations

Pau Balaguer[1] • Ivan Teixidó[2] • Jordi Vilaplana[2] • Jordi Mateo[2] • Josep Rius[3] • Francesc Solsona[2]

Pau Balaguer
pbalaguer19@gmail.com

Ivan Teixidó
iteixido@diei.udl.cat

Jordi Vilaplana
jordi@diei.udl.cat

Jordi Mateo
jmateo@diei.udl.cat

Josep Rius
josepmaria.rius@aegern.udl.cat

[1] Parc Científic i Tecnològic de Lleida, Building H3, ground floor, 25003 Lleida, Spain

[2] Department of Computer Science, University of Lleida, Jaume II 69, 25001 Lleida, Spain

[3] Department of AEGERN, University of Lleida, Jaume II 73, 25001 Lleida, Spain