Proceedings of the SALLD-2 Workshop @LREC2020, pages 9–18
Language Resources and Evaluation Conference (LREC 2022), Marseille, 24 June 2022

Movie rating prediction using sentiment features

João Ramos, Diogo Apóstolo, Hugo Gonçalo Oliveira
CISUC, DEI, Universidade de Coimbra, Portugal
uc2017254040@student.uc.pt, japostolo@student.dei.uc.pt, hroliv@dei.uc.pt

Abstract

We analyze the impact of using sentiment features in the prediction of movie review scores. The effort included the creation of a new lexicon, Expanded OntoSenticNet (EON), by merging OntoSenticNet and SentiWordNet, and experiments were made on the "IMDB movie review" dataset, with the three main approaches for sentiment analysis: lexicon-based, supervised machine learning and hybrids of the previous. Hybrid approaches performed the best, demonstrating the potential of merging knowledge bases and machine learning, but supervised approaches based on review embeddings were not far.

Keywords: Motive Rating Prediction, Sentiment Analysis, Supervised Machine Learning, Linked Data

1. Introduction

Sentiment Analysis (SA) has been applied to determine the sentiment conveyed by people in various situations. For instance, it can be useful for recommender systems, which may exploit the sentiment expressed by an user for items they have consumed, predict their sentiment for other items, and recommend those for which a positive sentiment is predicted. One particular application centres on the use of the sentiment conveyed by the words, as features for predicting the scores of movies or product reviews (Schuller and Knaup, 2010; Kapukaranov and Nakov, 2015; Agarwal et al., 2015; Cernian et al., 2015). Another popular application is sentiment analysis in social media publications (Rosenthal et al., 2017; Jovanoski et al., 2015).

Most of the previous adopt a very specific pipeline, presented in Figure 1. They start by either choosing a pre-existing dataset or creating one. The dataset is then preprocessed to be more easily analysed with sentiment extraction methods, often based on a sentiment lexicon, supervised machine-learning, or a hybrid of both. In some literature (Kapukaranov and Nakov, 2015; Schuller and Knaup, 2010), sentiment analysis is not the final goal, and the predicted sentiment is used as the input for another task. This is the case of our work, where sentiment is used for predicting movie review scores.

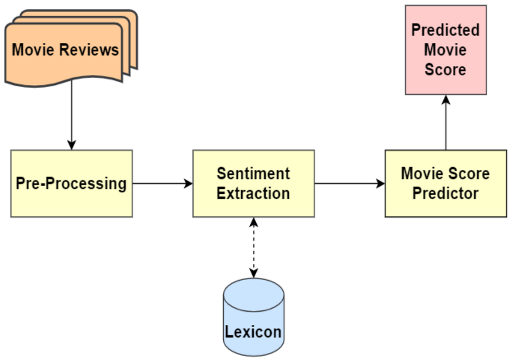

Figure 1: Typical pipeline

In order to better understand the impact of exploiting sentiment features for our goal, we experiment with the three different approaches: lexicon-based, supervised machine learning (SML) and hybrids of the previous. Our main contributions are:

- The creation of a new lexicon, Expanded OntoSenticNet (EON), which combines information from two sentiment resources, SenticNet (Cambria et al., 2010) and SentiWordNet (Esuli and Sebastiani, 2006);
- Experimentation with the recent IMDB movie review dataset (Pal et al., 2020);
- Attempting to predict the score of a review, not just the polarity, as most approaches do;
- Confirmation that sentiment features are useful for the prediction of review scores.

This paper is organized as follows: Section 2 overviews datasets and approaches for sentiment analysis; Section 3 describes the dataset and lexicons used in this work; Section 4 is on the setup of the experiments conducted, including details on implementation and parameterization; Section 5 reports and discusses the outcomes of the experiments; Section 6 concludes with the main take-aways and future work.

2. Background and Related Work

In this section, the details of the pipeline in Figure 1 is further elaborated upon, starting with a brief overview of the typical datasets used, followed by an explanation of each step.

2.1. Datasets

IMDB movie review datasets have been made available [1][2], with reviews and information like the publication date and name of the author. However, reviews

[1] https://www.kaggle.com/mantri7/imdb-movie-reviews-dataset/activity

[2] https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

are generally labelled as negative or positive, sometimes assuming a direct mapping between scores and polarity. We argue that, even if sentiment contributes to the score, they are different things. An exception is a dataset where reviews have a user-given score between 1 and 10 (Pal et al., 2020).

This kind of dataset can be collected from movie review websites like IMDB or Metacritic, which contain reviews in different languages (Schuller and Knaup, 2010; Kapukaranov and Nakov, 2015; Denecke, 2008). An alternative source of professional reviews is Rotten Tomatoes(Pang and Lee, 2005).

A similar methodology can and has been adopted for the creation of datasets for sentiment analysis in social networks (Jovanoski et al., 2015; Lobo and Pandya, 2019; Neethu and Rajasree, 2013).

2.2. Sentiment Extraction

Sentiment extraction refers to the application of natural language processing (NLP) for identifying and extracting subjective information in source materials. It is extensively applied to comments, posts and reviews, as a way of acquiring people's opinions about a subject (Shi et al., 2019). Sentiment extraction can be roughly separated into three main approaches: lexicon-based, supervised machine learning, and a hybrid.

2.2.1. Lexicon-Based Approaches

Semantic lexicons compile words and expressions together with sentiment-related information, such as the typical polarities they transmit. Lexicon-based approaches for sentiment classification resort to such resources for acquiring the polarity of words, which they combine towards sentence or document sentiment. The performance of these approaches is thus highly dictated by the quality of the lexicon, its size and how well it fits the specific problem. Lexicons are too resource-intensive to handcraft and, without the help of automatic methods, may fail to have a great coverage. To minimize this problem, one can start with a small dictionary of sentiment words and their polarity, and expand it iteratively through the analysis of: other available lexicons (Hu and Liu, 2004; Kim and Hovy, 2004); or corpora, e.g., based on co-occurrence statistics like PMI (Church and Hanks, 2002).

SentiWordNet (Esuli and Sebastiani, 2006), SenticNet (Cambria et al., 2010), GeneralInquirer (GI) (Stone et al., 2007), LIWC (Tausczik and Pennebaker, 2010) and VADER (Hutto and Gilbert, 2014) are among the most popular sentiment lexicons. SentiWordNet and SenticNet are known as valence-based, because they assign a continuous score for each word, not just a label (e.g., positive or negative). More specifically, in SentiWordNet each word has three scores: positivity, negativity and objectivity, and the sum of the three must add up to 1. SenticNet covers over 10,000 concepts, each with a score between -1 (negative) to 1 (positive). The VADER lexicon is based on LIWC, ANEW and GI, complemented by a list of western emoticons, sentiment related acronyms and slang. Though not considered by most of the other lexicons, these additions are a relevant for sentiment extraction. The new vocabulary was examined and given a score by multiple people. The VADER tool uses the VADER lexicon to calculate the polarity of sentences with four scores: negative, positive, neutral and compound.

Another lexicon (Agarwal et al., 2015) was built from SenticNet, SentiWordNet and GI. An ontology was created from ConceptNet (Speer et al., 2016) and other ontologies with domain-specific content. Towards sentiment extraction, document features are matched to the ontology, and their relevance is considered to be proportional to their distance to the root of the ontology. The final polarity of an opinion word is the result of *lexicon polarity*×*height of ontology*. Results show that the use of a context-specific ontology provides better results overall.

After choosing a lexicon, the polarity of a sentence can be computed by aggregating the sentiment values of included concepts that also occur in a sentiment lexicon. The polarity of all the sentences in a document will contribute to the overall polarity of the document. For example, SentiWordNet has been used for assigning a positivity, negativity and objectivity score to each sentence, from which the overall score was computed with logistic regression (Denecke, 2008). Similar approaches using SentiWordNet were adopted in other works (Bhoir and Kolte, 2015; Cernian et al., 2015)

Another work (Schuller and Knaup, 2010) explored GI and WordNet (Miller et al., 1991) for sentiment extraction simultaneously with the target of the sentiment. Out of the resulting expressions, the relevant ones are selected with the help of ConceptNet, to finally compute the document polarity score.

2.2.2. Supervised Machine Learning Approaches

An alternative to lexicons, which are not always suitable or available, is supervised machine learning (SML). These, however, require annotated data, which, for sentiment extraction, means textual documents and their manually-assigned polarity.

Moreover, to be exploited by SML approaches, documents generally have to be represented as numeric vectors, which can be obtained with algorithms such as TF-IDF, Doc2Vec, or more recently, sentence transformers. These may, however, make the interpretation of the results harder, if possible.

Traditional text classification algorithms have been used for determining the polarity of the document, including SVM, Naive Bayes and kNN (Yasen and Tedmori, 2019; Baid et al., 2017a; Baid et al., 2017b). Some test a range of classifiers, and assess the results with measures like accuracy, precision, recall, F1 and AUC. When tested in movie reviews, Random Forests proved superior to the remaining classifiers (Yasen and Tedmori, 2019). In the same scenario, SML approaches were also compared with lexicon-based (Schuller and Knaup, 2010). SML used a bag of n-grams representation and relied on an SVM to determine the polarity of the text document.

Experiments were also conducted to predict the score given by the user in the review, again with bag of n-grams features and a regression algorithm (SVR). SML was superior to the lexicon approaches, both in F1 and accuracy, but both methods had much difficulty for classifying negative reviews.

With the Deep Learning boom, there was a push to explore deep neural networks for sentiment extraction. Similarly to some traditional approaches, these models take embedded documents as their input, but they are more adequate for the large number of inputs the embedding generates. Recurrent Neural Networks (RNN) (Tang et al., 2015) were used for generating word representations from word to sentence level and then from sentence to document level, and applied to sentiment analysis. This resulted in better accuracy than previous approaches in several datasets. Traditional word embeddings (Word2Vec, GloVe, FastText) were also explored as the input of a Convolutional Neural Network (CNN), achieving the best accuracy in comparison to other tested algorithms (Vizcarra et al., 2018). LSTM networks were experimented in this task, with improvements achieved by ATAE-LSTM (Wang et al., 2016), an attention-based LSTM, which extracts features from each sentence and analyses the sentiment polarity of each aspect. Yet, since 2018, as it happens for other NLP tasks, the trend is to fine-tune neural language models based on transformers. Here, BERT performs especially well for sentence sentiment analysis (Habimana et al., 2020).

2.2.3. Hybrid Approaches

In hybrid approaches, the sentiment of a document is extracted with the help of both lexicons and content features, such as the number of positive/negative/objective sentences (Kapukaranov and Nakov, 2015). Document or sentence embeddings may be further exploited (Kapukaranov and Nakov, 2015; Keerthi Kumar et al., 2018; Kim et al., 2019).

For example, dependency parsing was combined with machine learning (Poria et al., 2014). Dependency-based rules are used for better capturing the role of a concept within a sentence and, if concepts are found in SenticNet, their polarity is obtained from this resource. Otherwise, an Extreme Learning Machine classifier, trained on a movie review dataset, is used to guess the sentence polarity.

Also, for movie reviews, content features (e.g., words, bigrams, emoticons) were exploited together with aggregated positive and negative scores of words, according to an automatically-generated lexicon, also considering meta information about each movie (e.g., actors, genre, director) (Kapukaranov and Nakov, 2015). From them, experiments were conducted for predicting the rating of the review with a SVM classifier or regression (SVR or logistic regression). A similar approach was adopted in the domain of social media sentiment analysis (Jovanoski et al., 2015).

A different task is to predict the success of movies from their plot summaries (Kim et al., 2019), also considering their sentiment. More precisely, classification considers the sentiment score of a document, computed with the VADER lexicon for each sentence, and its representation by ELMo (Peters et al., 2018) embeddings.

2.3. Current Challenges

Some authors have confirmed that using a general purpose sentiment lexicon like General Inquirer, with no context specific information, leads to a poor performance (Schuller and Knaup, 2010). This can be minimized both by the creation of larger lexicons, e.g., by merging existing ones and adding domain-specific information. An alternative is to adopt machine learning, which may also exploit lexicon features. Here, the lack of context may also result in more false positives (Schuller and Knaup, 2010), so it is recommended that training data is on the application domain. Moreover, to further increase performance, a larger set of features can be exploited, including meta information about the domain (Kapukaranov and Nakov, 2015).

We should add that much work with movie reviews aims at classifying polarity, i.e., whether a review is positive or negative. Even if, sometimes, the ground truth is obtained by converting the rating directly (Pang et al., 2002; Maas et al., 2011), classifying the polarity is not exactly the same problem as predicting the rating. As such, it would be interesting to further research on actually predicting the rating, e.g., with a regression algorithm, as others have done (Kapukaranov and Nakov, 2015; Schuller and Knaup, 2010).

3. Data

This section is on the data used in our experimentation, namely the dataset and the lexicons.

3.1. Dataset

We used a subset of the "IMDB Movie Reviews Dataset" (Pal et al., 2020), which originally contained nearly 1 million movie reviews from 1,150 different movies, across 17 genres[3]. For each review, the following features are provided:

- **username:** which identifies the review's author;
- **rating:** a score in the 1–10 interval, given by the author to the movie;
- **helpful:** the number of people that found the review helpful;
- **total:** the number of people who classified the review either as helpful or unhelpful;
- **date:** the date the review was written in;
- **title:** the title of the review, usually a short sentence that summarizes the author's opinion;

[3] https://ieee-dataport.org/open-access/imdb-movie-reviews-dataset

- **review:** a text review describing the opinion of the author about the movie.

For illustrative purposes, Figure 2 shows two entries of the dataset.

Our goal was to predict the rating by exploiting features extracted from the review. It is important to note that the rating distribution is not balanced.

However, using the full dataset would be impractical for the available time and computational power. We thus worked on a random selection of 10,000 instances of the dataset, including about 7,500 reviews rated higher than 5 and 2,500 rated 5 or lower (see Figure 3). Afterwards, the dataset was split into a cross-validation and a held-out evaluation set. The former contained 90% of the instances and the latter contained the remaining 10%. The cross-validation set was used to tune the parameters of the SML algorithms, which were then tested in the evaluation set. Lexicon-based approaches, which do not require training, are evaluated on the evaluation set.

3.2. Lexicons

The lexicons explored in this work were SenticNet, more precisely, its ontology version, OntoSenticNet, and SentiWordNet. Having in mind the benefits of combining lexicons, we created a new ontology, Expanded OntosenticNet (EON), with information from both. From OntoSenticNet, we extracted the 'polarity' annotation, a score between -1 (negative) and 1 (positive) available for each word and expression. From SentiWordNet, we used the 'positive' (SWN_Pos) and 'negative' (SWN_Neg) scores, each ranging from 0 to 1. This way, each entry in the lexicon would have at most three sentiment-related scores.

OntoSenticNet is represented in RDF/OWL and was queried with RDFLib[4]. SentiWordNet scores were obtained with the NLTK[5] interface available for querying this resource.

In EON, words or expressions that are in only one of the lexicons stay only with the annotations from the lexicon they are in. SentiWordNet words with only objectivity scores were not considered, as they would only add noise to the predictions.

EON is available in RDF[6]. To look up the polarity scores, we use the SPARQL query in Listing 1. Table 1 illustrates the possible results, with examples for different tokens. The 'polarity' column comes from OntoSenticNet, and the other two come from SentiWordNet.

```
SELECT ?SenticConcept ?text ?polarity ?SWN_Positive
?SWN_Negative WHERE {
 ?SenticConcept :text ?text.
 ?SenticConcept :text <token> .
 ?SenticConcept :polarity ?polarity .
 ?SenticConcept :SWN_Positive ?SWN_Positive.
 ?SenticConcept :SWN_Negative ?SWN_Negative.
}
```

Listing 1: SPARQL query for retrieving polarities from EON

Table 1: Example results for SPARQL query in Listing 1, for different tokens.

Token	polarity	SWN_Pos	SWN_Neg
abhorrent	-0.44	0.00	0.75
good	0.66	0.69	0.00
food	0.03	0.00	0.04

4. Experimentation Setup

This section details the setup of the conducted experiments, namely on: data preprocessing, tested approaches, and parameterization of the algorithms, also covering the adopted evaluation metrics.

4.1. Preprocessing

The reviews were prepossessed with Python's NLTK package. This step included: the removal of HTML tags; sentence splitting; the removal of punctuation and stopwords; tokenization and lemmatization.

4.2. Experimented Approaches

Experimentation was performed with three different groups of approaches described here.

4.2.1. Lexicon Approach

In order to get the polarity of the reviews, EON is queried, with RDFLib, for each lemmatized token in the document. Whenever the token is in EON, the three polarity values (polarity, SWN_Pos, SWN_Neg) are obtained with the query in Listing 1. The token sentiment score s is calculated according to equation 1, where p is the polarity value in OntoSenticNet, and swp and swn are respectively SWN_Pos and SWN_Neg.

$$s = \frac{(p + \frac{(swp - swn)}{2})}{2} \tag{1}$$

Seven different options were tested for aggregating token sentiment values in a sentence sentiment, namely:

- **Mean:** mean value for all tokens;
- **Max:** highest token value (positive or negative);
- **Max 3/5:** mean of 3/5 highest token values;
- **Neg 2/3/4:** mean of all token values, but with negative values weighting twice/three times/four times as more as positive;

Since review scores range from 1 to 10, the result of the previous methods, which range from 0 to 1, was mapped to the 1–10 interval. Two different mapping

[4] https://rdflib.readthedocs.io/en/stable/

[5] https://www.nltk.org/

[6] https://github.com/DiogoApostolo/EON

username: red95king **rating:** 1 **helpful:** 3 **total:** 8 **date:** 10/01/2002
title: *The Moronic & Ridiculous*
review: *This move was so dumb I don't even know where to begin. Put next to this, films "Stone Cold", "Harley Davidson and the Marlboro Man", and "Road House" look like cinematic masterpieces. If only it were true that you could roll a car 12 times at 100 miles per hour and come out with hardly a scratch. Granted there are some outstanding stunts, but not enough action overall to offset the non-sense plot and 3rd rate acting. Don't get me wrong I consider Vin Diesel a pretty good actor, but the script sounds like it was written for (or perhaps by) 8 year olds. Vin, your talents were wasted buddy. Watch "Grand Prix" instead.*

username: Shervin1982 **rating:** 4 **helpful:** 0 **total:** 0 **date:** 16/05/2003
title: *Neo has to choose!*
review: *I wouldn't call it a movie, rather a sequence of actions. If you're looking forward to watching fight scenes for over an hour, this is a must see. But the movie as a whole, is very poor and aimless. Martix reloaded compare to its prequal is very disappointing.*

Figure 2: Reviews for movies "The Fast and the Furious" (2001) and "The Matrix Reloaded" (2003).

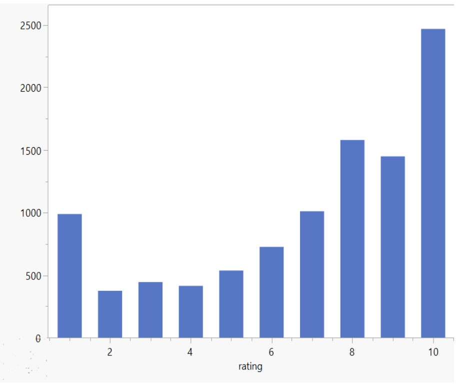

Figure 3: Distribution of labels in the dataset

functions were tested for this, namely: (1) splitting the sentiment score space into 10 equal intervals and use that as a basis to calculate the regression (Linear); (2) create intervals proportional to the frequency of each review score in the dataset (Frequency-Sensitive).

4.2.2. Supervised ML Approach

The SML approaches can be divided in two main steps: (i) Vectorization; (ii) Regression. The vectorization step takes the output of preprocessing and represents the documents into numeric vectors to be used by the regression algorithm. For this, we experimented with both Doc2Vec (Le and Mikolov, 2014), using *gensim*[7], and TF-IDF, using *scikit-learn*[8], which were fit to the training set. For the regression, we opted for Support Vector Regression (SVR), available in *scikit-learn*, because it is a popular option for this purpose in the literature (Yasen and Tedmori, 2019; Baid et al., 2017a; Baid et al., 2017b), particularly in the prediction of movie review scores (Kapukaranov and Nakov, 2015).

4.2.3. Hybrid Approach

For the hybrid approach, SVR is also used, but in can be trained in: polarities obtained from the lexicon; the previous concatenated to the document embedding. Token aggregation and embedding methods are chosen according to the best results of the pure lexicon and SML approaches.

4.3. Algorithm Parameters

For SVR, the C and $epsilon$ hyperparameters were tuned. For C, tested values ranged from 2^{-3} to 2^4, and for $epsilon$, from 0 to 4.
For Doc2Vec, we experimented with different vector sizes to conclude that 200 was the one to use. We also experimented with 10, 100, 200 and 500 training epochs. For TF-IDF, we cut the maximum number of features produced by the algorithm, as there were close to 40,000 different tokens across all text documents. We experimented with keeping only the 500, 1000, 1500 and 2000 most important tokens.

4.4. Evaluation Metrics

We compare the performance of the different approaches on the evaluation set, mentioned earlier. Performance is evaluated in terms of Mean Squared Error (MSE), Mean Absolute Error (MAE) and Pearson Correlation (ρ) between the predicted and the gold rating.

5. Experimentation and Results

For each family of approach, at least one experiment was made in order to determine which is the best. For SML, multiple experiments were run to tune the SVR parameters and to select the best embedding method, between Doc2Vec and TF-IDF.
As for the lexicon approach, we measured the performance of the algorithm based on the lexicon, the token aggregation function, the sentiment aggregation function, and the mapping function. Experiments were made using EON, but also OntoSenticNet alone.
For the hybrid approach, we selected the best performing methods in the lexicon and SML approaches. Experiments with and without the use of vectorization were also conducted.

5.1. Lexicon Approach

We conducted the pairwise analysis of the lexicon approach for each variable, but for the sake of presenting

[7] `https://radimrehurek.com/gensim/`
[8] `https://scikit-learn.org/`

the acquired information in a digestible way, we analyse each variable individually. First, we analyse the impact of the token aggregation option on performance, reported in Table 2. This was computed with EON, the frequency-sensitive mapping function and, since we needed a document score for computing the metrics, the Mean was used for sentence aggregation. For all metrics, the best performance was achieved with the Neg 4 aggregation, suggesting that negative opinions are more important for the sentence sentiment. Following this, we decided to use Neg 4 for token aggregation in further experimentation.

Table 2: MSE, MAE, Correlation based on the token aggregation function

Token Aggr.	**MAE**	**MSE**	ρ
Mean	2.428	10.068	0.170
Max	2.453	10.190	0.130
Max 3	2.427	10.457	0.217
Max 5	2.420	10.107	0.227
Neg 2	2.402	9.536	0.214
Neg 3	2.404	9.351	0.206
Neg 4	**2.395**	**9.017**	**0.244**

We then analysed sentence aggregation. Table 3 reports the performance for each option, using Neg 4 for token aggregation and EON. Here, Max 5 achieved slightly better results in MAE and ρ, while Neg 4 got a better MSE. As the differences were low, we decided to opt also for Neg 4 for sentence aggregation.

Table 3: MSE, MAE, Correlation based on the Sentence aggregation function

Sentence Aggr.	**MAE**	**MSE**	ρ
Mean	2.394	9.087	0.226
Max	2.452	9.502	0.186
Max 3	2.390	9.173	0.238
Max 5	**2.389**	9.042	**0.247**
Neg 2	2.400	8.835	0.247
Neg 3	2.411	8.785	0.242
Neg 4	2.421	**8.775**	0.237

After selecting both the token and sentence aggregation methods, we used them for checking whether the created lexicon, EON, was a better option than OntoSenticNet. The figures in Table 4 show that EON performs better, confirming the benefits of using a larger lexicon, resulting from the combination of two slightly different, possibly complementary, ones.

Table 4: MSE, MAE, Correlation based on the lexicon

Lexicon	**MAE**	**MSE**	ρ
OntoSenticNet	2.443	9.640	0.211
EON	**2.421**	**8.775**	**0.237**

Lastly, we examined the performance of the two functions proposed for mapping sentiment values (0–1) to the review score (1–10). Figures in Table 5 show that, for all metrics, the Frequency-Sensitive function leads to a substantially better performance than the Linear.

Table 5: MSE, MAE, Correlation based on the mapping function.

Mapping	**MAE**	**MSE**	ρ
Linear	3.177	12.880	0.177
Freq.-Sensitive	**2.421**	**8.775**	**0.237**

Following the experiments with the lexicon, these were our main decisions:

- When aggregating both token sentiment in sentence scores and sentence scores into document scores, negative sentiment scores are weighted four times more than positive (Neg 4). The effectiveness of this option can be the consequence of an imbalance in the lexicons, especially in OntoSenticNet, where positive terms are abundant and many seemingly neutral terms (e.g., “frequent" or “pick") have high positive scores.
- EON is a better option than OntoSenticNet alone. Including information from two different sources of knowledge enables to compute polarities that better reflect the real sentiment connotation of each word. This provides empirical evidence that the creation of broader sentiment lexicons, by merging already available ones, is effective.
- To map from the 0–1 interval that the approaches output to the 1–10 of the reviews, use a frequency-sensitive mapping function instead of its linear counterpart. This makes sense because the distribution of the review scores in the dataset is not linear, being more skewed towards the middling values of the scale.

5.2. Supervised ML Approach

Figure 4 shows the variation of MSE for different values of the SVR hyperparameters, C and $epsilon$, in experiments using Doc2Vec (left), with 10, 200 and 500 epochs, or TF-IDF (right), with 500, 1000 and 2000 maximum features.

Increasing the number of epochs leads to lower MSE for the majority of the SVR parameters. On the other hand, the increase from 200 to 500 does not lead to improvements. For TF-IDF, increasing the number of features helps to decrease MSE slightly. Specifically, going from 500 to 1000 features improves the performance more clearly than from 1000 to 2000, where it seems to almost stagnate. As for the best SVR parameterization, we can see that the best results are obtained with $C > 2$ and $epsilon$ between 0 and 1.

Figure 5 compares TF-IDF and Doc2Vec with the best parameters obtained previously (2000 and 200 respectively). Overall, they seem to perform equally in the best case scenarios, while for parameters where the performance degrades, the errors of Doc2Vec are lower.

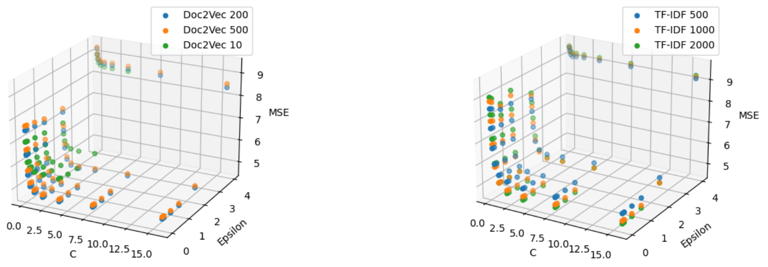

Figure 4: MSE for Doc2Vec and TF-IDF for multiple SVR parameters in the cross-validation set.

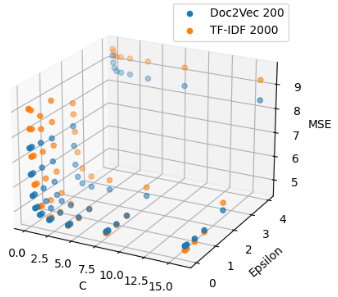

Figure 5: Comparison of the best parameters of Doc2Vec and TF-IDF in the cross-validation set.

Table 6 summarizes the best results obtained for Doc2Vec and TF-IDF in terms of MSE, MAE and ρ. While MSE is the same as in Figure 5, here it is possible to observe similar results for MAE and ρ, where both embedding methods obtain close results, even if TF-IDF has a slight advantage.

Table 6: MSE, MAE, Correlation for embedding method in the cross-validation set.

C/Epsilon	Emb.	MAE	MSE	ρ
4/0.01	Doc2Vec	1.694	4.691	0.684
4/0.01	TF-IDF	1.688	4.662	0.687

Based on the previous results, we set $C = 4$ and $epsilon = 0.01$ for the SVR. For the embedding, we opted for 200 epochs for Doc2Vec and 2,000 maximum features for TF-IDF. With these parameters, the approaches were tested in the evaluation set, with results in Table 7.

Table 7: MSE, MAE, Correlation for the embedding method in the evaluation set

C/Epsilon	Emb.	MAE	MSE	Corr.
4/0.01	Doc2Vec	1.953	5.879	0.623
4/0.01	TF-IDF	1.712	4.886	0.686

The difference in performance between both embedding approaches becomes more apparent in the evaluation set. While TF-IDF achieves a similar performance to cross-validation, Doc2Vec increases MAE and MSE significantly. TF-IDF is therefore be the embedding method used in the hybrid approach experiments.

Following the experiments with the SML approach, we have the following observations:

- When using Doc2Vec, we were expecting a positive correlation between performance and the number of training epochs. Indeed, 200 epochs leads to a lower error in regression than 10, but also than 500, which might be a consequence of overfitting to the training dataset.
- For TF-IDF, the more features are considered, the lower the error.
- Even though both embedding options performed similarly in cross-validation, this did not hold up in the evaluation set. This suggests that, by using a larger representation, TF-IDF is able to better embed a more varied set of documents.
- The SVR is used with the following hyperparameters: $C = 4$ and $epsilon = 0.01$.

5.3. Hybrid Approach

The hybrid approach combines the lexicon and SML approaches, using the best parameters for each, selected after the results of the previous sections. Four hybrid configurations were tested, where different document representations were used with the SVR, namely: EON sentiment values (Tok.); EON sentiment values concatenated to the TF-IDF vector (Tok. + TF-IDF); sentence sentiment scores, obtained with the Neg 4 function (Sent.); and sentence sentiment scores concatenated to the TF-IDF vector (Sent. + TF-IDF). In each of those experiments, zero padding was applied in order to assure equal input size. This procedure was required because sentences with different number of tokens, and reviews with different number of sentences, cause discrepancies in instance input size. Table 8 shows the results obtained.

Table 8: MSE, MAE, Correlation for the Hybrid Approaches in the cross-validation set.

C/Eps.	Method	MAE	MSE	ρ
0.5/1	Tok.	2.357	8.729	0.127
4/0.01	Tok. + TF-IDF	1.726	4.871	0.670
0.5/1	Sent.	2.318	8.848	0.211
4/0.01	Sent. + TF-IDF	1.666	4.630	0.691

Figures show that the vector representation is fundamental for a good performance, as without them it would not be much different from the lexicon approach. The sentence sentiment score also leads to better results than the polarity values extracted directly from the lexicon. As such, using as input the sentence sentiment score and the vector representation showed to be the best hybrid approach.

Following this, the best hybrid approaches were tested in the evaluation set, with results in Table 9. As it happened in the cross-validation, the best results are achieved with sentence + TF-IDF. However, the performance of both degrades, especially for Token + TF-IDF.

Table 9: MSE, MAE, Correlation for the Hybrid Approaches in the evaluation set

C/Eps.	Method	MAE	MSE	Corr.
4/0.01	Tok. + TF-IDF	1.798	5.308	0.6524
4/0.01	Sent. + TF-IDF	1.699	4.879	0.686

These experiments showed that:

- There are no clear improvements between learning regression from the polarities obtained from the lexicon or applying equation 1. This was somewhat expected, following the difference between the SML and lexicon approach, implying that the vector representation has more discriminant power.
- When polarities from the lexicon are concatenated with the embedding of the documents, there are improvements, but they are minimal.
- Using the aggregated token polarity for each sentence, instead of all the individual token polarities, performed slightly better. A possible cause is the increase of dimensionality when using all the tokens. Moreover, the number of tokens across all documents varies greatly, so the vectors must be zero padded to make the representation valid for the SVR. The same process must also be done for the sentences. However, the amount of padding is much lower, so less noise is inserted. This was backed up by the results in the evaluation set.

6. Conclusion

In this paper, we reported experiments with the three popular approaches for sentiment analysis in movie reviews: lexicon-based, supervised machine learning (SML), and hybrid approaches. In an attempt to create a more complete knowledge source for sentiment analysis, a new lexicon, EON, was created, by merging OntoSenticNet and SentiWordNet. Moreover, for each approach, experiments were made to identify the best parameters with cross-validation. The actual comparison was run in an evaluation set with data not used before. Evaluation was based on three metrics: MAE, MSE and Pearson correlation (ρ).

Out of the three approaches, hybrid yielded better results, but the difference was not substantial when compared to the SML approaches. The lexicon approach performed the worst in all metrics, mainly due to coverage and contextual issues. We further noticed that the lexicon is skewed towards positive polarities. This introduces error, making it difficult to accurately predict the true rating of the movie reviews. Overall, this behaviour matches the results found in literature (Shi et al., 2019), where pure lexicon-based approaches tend to perform worse than SML or hybrid approaches.

Despite its poor results, EON outperformed SenticNet, backing up the claim that both the size and quality of the lexicon is of extreme importance. As such, we believe that it would be important to repeat the experiment with a better lexicon, more relevant to the context of movie reviews.

SML, based on SVR, performed much better, confirming that there is more information to be extracted in the raw data than in the lexicons, as the bias towards positive tokens is not present in the vector embedding.

The hybrid approach lead only to minor improvements. This may be due to: (i) the information provided by the lexicons is flawed, as mentioned previously; (ii) traditional SML models, like an SVR, are not ideal for this kind of analysis, as the dimensionality of the data is very large and finding relations between tokens can be too complex for this model.

It is also worth noting that hybrid approaches that only use sentiment values from the lexicons perform similarly to the pure lexicon approaches, further backing up the hypothesis that content features are essential, and that lexicon scores are flawed representations of the true sentiment value of the tokens, or at least not suitable for the movie review domain. Finally, even though the errors are high considering the scale used for all approaches, supervised and hybrid approaches have relatively high correlation, indicating that, at least, the relative order of the predicted ratings is close to the real ones.

A natural step further would be to adopt state-of-the-art approaches for sentiment analysis, and text classification in general. The focus would, of course, be on deep neural networks, specifically RNNs (Tang et al., 2015) or Transformers, possibly starting with pre-trained language models (Yin et al., 2020), which should achieve better results and would allow for a better consideration of the contexts where words are used.

7. Bibliographical References

Agarwal, B., Mittal, N., Bansal, P., and Garg, S. (2015). Sentiment analysis using common-sense and context information. *Computational intelligence and neuroscience*, 2015:715730, 03.

Baid, P., Gupta, A., and Chaplot, N. (2017a). Sentiment analysis of movie reviews using machine learning techniques. *International Journal of Computer Applications*, 179:45–49, 12.

Baid, P., Gupta, A., and Chaplot, N. (2017b). Sentiment analysis of movie reviews using machine learning techniques. *International Journal of Computer Applications*, 179:45–49, 12.

Bhoir, P. and Kolte, S. (2015). Sentiment analysis of movie reviews using lexicon approach. In *2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pages 1–6.

Cambria, E., Speer, R., Havasi, C., and Hussain, A. (2010). SenticNet: A publicly available semantic resource for opinion mining. In *AAAI Fall Symposium: Commonsense Knowledge*, volume FS-10-02 of *AAAI Technical Report*. AAAI.

Cernian, A., Sgârciu, V., and Martin, B. (2015). Sentiment analysis from product reviews using sentiwordnet as lexical resource. *2015 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages WE–15–WE–18.

Church, K. and Hanks, P. (2002). Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16, 07.

Denecke, K. (2008). Using sentiwordnet for multilingual sentiment analysis. *2008 IEEE 24th International Conference on Data Engineering Workshop*, pages 507–512.

Esuli, A. and Sebastiani, F. (2006). SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy, May. ELRA.

Habimana, O., Li, Y., Li, R., Gu, X., and Yu, G. (2020). Sentiment analysis using deep learning approaches: an overview. *Science China Information Sciences*, 63(1):1–36.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 168–177, New York, NY, USA. Association for Computing Machinery.

Hutto, C. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225, May.

Jovanoski, D., Pachovski, V., and Nakov, P. (2015). Sentiment analysis in twitter for macedonian. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 249–257.

Kapukaranov, B. and Nakov, P. (2015). Fine-grained sentiment analysis for movie reviews in Bulgarian. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 266–274, Hissar, Bulgaria, sep. INCOMA Ltd. Shoumen, BULGARIA.

Keerthi Kumar, H. M., Harish, B. S., and Darshan, H. (2018). Sentiment analysis on imdb movie reviews using hybrid feature extraction method. *International Journal of Interactive Multimedia and Artificial Intelligence*, InPress:1, 01.

Kim, S.-M. and Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, page 1367–es, USA. ACL.

Kim, Y. J., Cheong, Y. G., and Lee, J. H. (2019). Prediction of a movie's success from plot summaries using deep learning models. In *Proceedings of the Second Workshop on Storytelling*, pages 127–135, Florence, Italy, August. ACL.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–1188–II–1196. JMLR.org.

Lobo, V. and Pandya, B. (2019). Sentiment analysis of Twitter data to predict the performance of movies. In *International Conference on Intelligent Systems and Communication Networks*.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June. Association for Computational Linguistics.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. (1991). Introduction to wordnet: An online lexical database*. 3, 01.

Neethu, M. and Rajasree, R. (2013). Sentiment analysis in Twitter using machine learning techniques. pages 1–5, 07.

Pal, A., Barigidad, A., and Mustafi, A. (2020). Identifying movie genre compositions using neural networks and introducing genrec-a recommender system based on audience genre perception. In *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, pages 1–7. IEEE.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, page 115–124, USA. ACL.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the*

2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), pages 79–86. ACL.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *CoRR*, abs/1802.05365.

Poria, S., Cambria, E., Winterstein, G., and Huang, G.-B. (2014). Sentic patterns: Dependency-based rules for concept-level sentiment analysis. *Knowledge-Based Systems*, 69:45–63.

Rosenthal, S., Farra, N., and Nakov, P. (2017). SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada, August. ACL.

Schuller, B. and Knaup, T. (2010). Learning and knowledge-based sentiment analysis in movie review key excerpts. In *Proceedings of the Third COST 2102 International Training School Conference on Toward Autonomous, Adaptive, and Context-Aware Multimodal Interfaces: Theoretical and Practical Issues*, page 448–472, Berlin, Heidelberg. Springer-Verlag.

Shi, Y., Zhu, L., Li, W., Guo, K., and Zheng, Y. (2019). Survey on classic and latest textual sentiment analysis articles and techniques. *International Journal of Information Technology & Decision Making*, 18(04):1243–1287.

Speer, R., Chin, J., and Havasi, C. (2016). Conceptnet 5.5: An open multilingual graph of general knowledge. *CoRR*, abs/1612.03975.

Stone, P., Bales, R., Namenwirth, J., and Ogilvie, D. (2007). The general inquirer: A computer system for content analysis and retrieval based on the sentence as a unit of information. *Behavioral Science*, 7:484 – 498, 10.

Tang, D., Qin, B., and Liu, T. (2015). Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432, Lisbon, Portugal, September. ACL.

Tausczik, Y. R. and Pennebaker, J. W. (2010). The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Vizcarra, G., Mauricio, A., and Mauricio, L. (2018). A deep learning approach for sentiment analysis in spanish tweets. In Věra Kůrková, et al., editors, *Artificial Neural Networks and Machine Learning – ICANN 2018*, pages 622–629, Cham. Springer International Publishing.

Wang, Y., Huang, M., Zhu, X., and Zhao, L. (2016). Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas, November. ACL.

Yasen, M. and Tedmori, S. (2019). Movies reviews sentiment analysis and classification. 04.

Yin, D., Meng, T., and Chang, K.-W. (2020). SentiBERT: A transferable transformer-based architecture for compositional sentiment semantics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3695–3706, Online, July. ACL.