ECE657A project report: Sentiment analysis on the *Drugs.com* Dataset

Natalia Hoyos Velasquez, Juan Manuel Gomez Gonzalez, Ruoxuan Xu

***Abstract*— There is a growing popularity and availability of opinion resources like forums and review sites that open new opportunities to actively analyze the public's opinion or feelings towards objects of interest. Data mining approaches such as sentiment analysis of large amounts of text data has captured the attention of different researchers around the world as potential applications range from the academical to the medical and commercial field. In the pharmaceutical field, getting valuable insight from drug reviews of thousands of users is promising, as it would lead to an improvement in the way the public health system is monitored as well as helping in the post-marketing pharmaceutical surveillance.**

The main challenge of applying sentiment analysis in the medical field is that most algorithms and models are domain specific and fail to perform well when applied in a different field. In this work, different methodologies to do the classification polarity of drug reviews are explored. Aiming to obtain the accuracy (92.24%) achieved by in Felix Graer et al as reference, the application of multiple approaches are divided in two types: Lexicon-based approach and machine learning-based approach, using the *drug.com* dataset [1].

***Index Terms*— sentiment analysis, drug reviews, machine learning, supervised learning approach, lexicon-based approach, text classification, neural networks.**

I. Introduction

Review sites found on the internet have been a great source of information for knowledge discovery. These sites contain data related to the preferences and experiences that the users of certain products have, in order to gain an overview of the wider customer opinion behind the use of those products. This type of knowledge becomes medically useful when prescribing pharmaceutical drugs, as it could give the doctors insight into what patients have perceived during the course of their treatment, at the same time giving patients a better understanding of what effects they might experience with the medications they are taking. It can also help as a guide that helps the prescriber make a more accurate decision on the treatment and type of drugs to use on each patient,while having a better understanding of what adverse effects other users have felt with the use of those pharmaceuticals.

Furthermore, in the field of pharmacovigilance, implementing sentiment analysis focused on drug reviews could help governmental agencies make informed decisions on health policies related to the approval or reassessment of certain medications, as well as improve the government oversee of the communitys public health by allowing the access to a type of collective health experience [2].

The objective of this work is to categorize the positive, neutral and negative feedback of customers using different pharmaceutical products, with the use of a model built to obtain the polarity value of drug reviews in a way that allows for future extraction of data from non-structured sources of information.

A. Sentiment Analysis

Sentiment analysis is the discipline that studies peoples opinions, attitudes and emotions towards object of interest such as products, services and topics. It usually involves applying Natural Language Processing (NLP) and text analysis techniques to identify, extract and interpret subjective information from a text. Sentiment analysis can also be referred to as Opinion extraction, or Sentiment mining [3].

Sentiment analysis ends up to be a classification in most of the cases, which is the problem of reporting a sentiment tag (e.g: positive or negative) given an opinionated text object.

Nowadays customers are much more inclined towards reviews from others to buy the products. Analyzing the data from those customers' reviews to understand their sentiments is an essential field. The rapid growth of information available in social media and the advances in NLP and machine learning techniques is presented as a good opportunity to mine these review data in different domains such as movie reviews, product reviews, and political environment [4]–[9], [15].

B. Pharmaceutical Product Safety

Doing sentiment analysis from drug reviews to understand the opinions of certain medicines among buyers benefits both the customers and the supervision departments to ensure pharmaceutical product safety. The applications of mining sentiments in medical domain include detecting side-effect for pharmacovigilance system [10] and user recommendation system using healthcare social media [11].

Currently, medical product safety mostly rely on clinic trials and specific test protocols which are done within a limited number of test subjects and in a limited time. Nevertheless, Pharmacovigilance has being evolved form passive surveillance to active surveillance and consequently post marketing drug surveillance is needed for a more comprehensive evaluation of drugs. There exist some approaches like clinical decision support systems (CDSS) and therapy recommendation systems, but they all require structured data with intense preparation which makes it harder for practical use. Therefore, exploring techniques for sentiment analysis in unstructured data such as social media, web forums and review sites is very meaningful.

C. Literature Review: Related Work on Sentiment Analysis

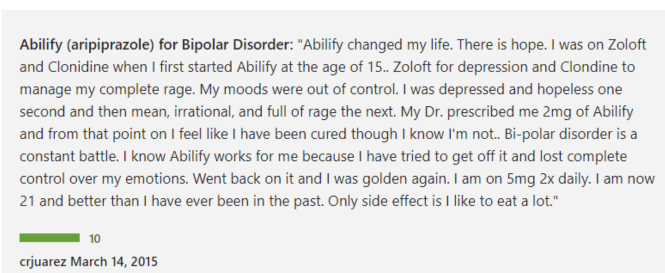
Sentiment analysis techniques development has quickly increased in recent years due to the advances in techniques such as NLP and machine learning. While many of the researchers focus on assigning sentiment polarity to documents, there are also other researchers aim at more specific task. For example, finding the sentiment of words [12], subjective expressions [13], [14], subjective sentences [15] and topics [16], [17].

Different techniques have been applied to predict the sentiment polarity. Many approaches tackling adverse drug reaction(ADPs) or side effect identification are lexicon-based relying on mapping relevant terms and phrases from user data to specific vocabularies from lexicons [18], [19]. Other methods of sentiment polarity classification train a machine learning algorithm. The machine learning approaches applicable to sentiment analysis mostly belong to supervised learning in general and text classification in particular which have achieved great success recently. Gopalakrishman et al. analyzed drug satisfaction from patients using supervised learning sentiment analysis approach. They classified three levels of polarity comparing SVM with neural network based methods, finding that the neural network outperforms the SVM method [20]. In 2011, Andrew L.Maas et al.present a model that uses a mix of unsupervised and supervised learning techniques to learn word vectors capturing semantic term-document information as well as rich sentiment content [21]. Hybrid approaches combines both lexicon-based and machine learning approaches together by increasing the complexity of the model. [22]–[24].

Domain issue is also a challenge for sentiment analysis. The model or classifier trained in one domain often does not perform well in a different field. Many research studies are working on improving domain adaption or cross-domain sentiment analysis [1], [25], [26]. In [27] a comprehensive systematic literature review on cross-domain sentiment analysis is presented.

II. *drug.com* DATASET

The *drugs.com* dataset is based on an automatic web crawl made by Grer et al on the *drugs.com* website in which they performed sentiment classification and obtained an accuracy of 92.6% [1]. The intention of this work is to explore different algorithms and recreate the accuracy they obtained. The process the authors used was to obtain the raw HTML related to the information of reviews from different pharmaceuticals. This reviews contained the name of the drug, the condition the drug treats, the review text, the rating given by the reviewer, the date of the review, and the amount of people that found that particular review helpful. Figure 1 shows a sample review for the Abilify medication from a user on the website.



Abilify (aripiprazole) for Bipolar Disorder: "Abilify changed my life. There is hope. I was on Zoloft and Clonidine when I first started Abilify at the age of 15.. Zoloft for depression and Clondine to manage my complete rage. My moods were out of control. I was depressed and hopeless one second and then mean, irrational, and full of rage the next. My Dr. prescribed me 2mg of Abilify and from that point on I feel like I have been cured though I know I'm not.. Bi-polar disorder is a constant battle. I know Abilify works for me because I have tried to get off it and lost complete control over my emotions. Went back on it and I was golden again. I am on 5mg 2x daily. I am now 21 and better than I have ever been in the past. Only side effect is I like to eat a lot."

10

crjuarez March 14, 2015

Fig. 1. A sample review from *drug.com*

For this study, the review and the rating were taken into consideration. The data-set has a clear separation, made by Grer et al, between the training and testing data, with 161297 and 53766 samples respectively.

III. METHODOLOGY

A. Data Pre-Processing

1) Pre-analysis of the data: In order to make a better comparison between this work and [1], the dataset was divided in the same way as the cited paper into three categories, positive, neutral and negative. Positive ratings were those that had a rating of 8, 9 and 10, while the neutral reviews were those with a rating of 4, 5, 6 or 7. Finally, the negative reviews were those that had a rating of 1, 2 or 3. The results of this segregation can be seen in Figure 2.

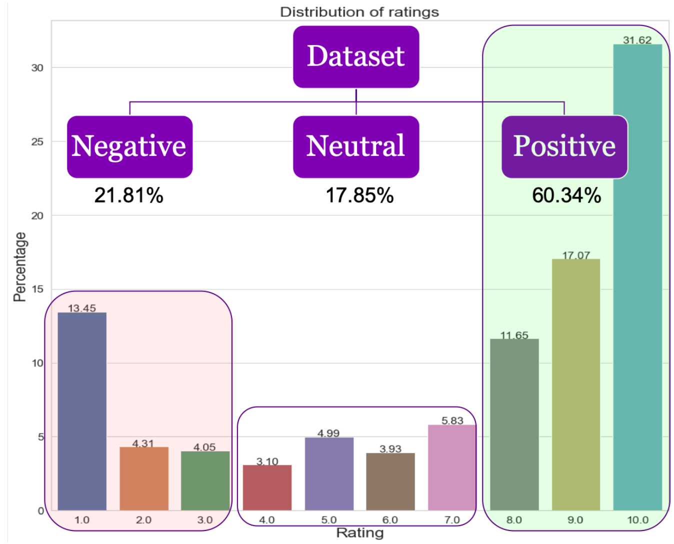


Fig. 2. Distribution of reviews from *drug.com* dataset

Figure 3 and Figure 4 show word clouds for the positive and negative reviews, respectively. The size of the word indicates the relative frequency compared to other words, where the bigger the word, the more frequent it is in the reviews.

Fig. 3. Word cloud for positive reviews

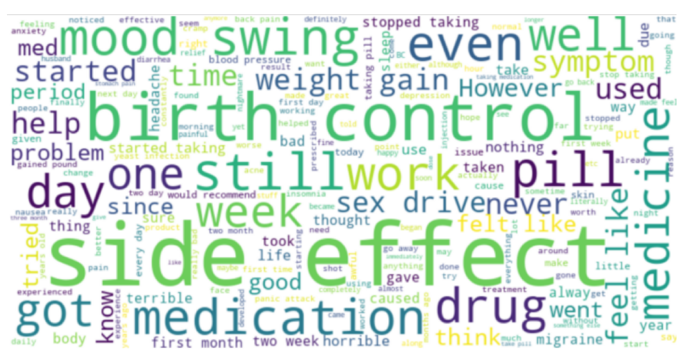


Fig. 4. Word cloud for negative reviews

It is worth noting that words like "Side Effect"and "Birth Control" appear in both word clouds. This indicates that these words are predominant in both types of reviews and thus would not contain meaningful information related to the sentiment of a particular review.

2) Data cleaning: Data cleaning was necessary to remove some non-useful information and artifacts found in the reviews corpus. First the HTML character entity references were converted to the correct display form, followed by the conversion of the text to lower case (except for the *VADER* classification, which will be mentioned later on).

Removal of numbers, extra spaces, non-alphanumeric characters and stop words was also done on the review dataset. In general, stop words are considered common words of a language, e.g. I, me, the, etc. that, although containing useful information for other applications, are too ubiquitous in the text to have an inherent sentiment and therefore can be discarded for the case of sentiment classification.

3) Tokenization: Tokenization is the process in which the different words found in the documents of a corpus are demarcated and separated from other words, creating what can be called tokens. During this process punctuation is also removed. All the relevant tokens still remain as a sample even though they each are separate from one another.

Two tokenization methods were used in this work. The first one is count vectorization, which builds an array of all the words present in the complete corpus of the dataset, and for each word assigns either a 1 or 0, depending on if the word appears in the respective sample or not. The second tokenization method is Term Frequency-inverse Document Frequency (tf-idf), which, in general terms, adds more value to words that appear repetitively in a sample but not in many samples, highlighting important and unique words.

B. Lexicon-based Sentiment Analysis

Most of the work in sentiment analysis is based on two methods: Lexicon based and supervised machine learning.

The lexicon-based approach evaluates the words based on a predefined opinion lexicon to determine the sentiment behind each word and thus the general sentiment of the complete text. The main advantage of lexicon based algorithms is that they can be employed directly on the dataset without the need of training data. However, this causes the lexicon based approaches to be generally static, and as such suggests that they might not work well for specific domains like the medical field.

NLTK's *Valence Aware Dictionary and sEntiment Reasoner* (*VADER*) implementation is used as a representative of the lexicon-based approach to do the polarity classification [28]. NLTK's VADER is a module of the NLTK python library based on lexicons of sentiment-related words. In this approach, each of the words in the lexicon is rated as to whether it is positive or negative, and in many cases, how positive or negative. It also takes into account if the word is preceded by a negation, letter case of the words and even the amount of exclamation points in the sentence. *VADER* returns a compound score with the sum of all the scores it interpreted in a given document, where -1 indicates that the document is extremely negative and +1 indicates that the document is extremely positive.

C. Supervised Learning Sentiment Analysis

1) Fundamental supervised learning models: Another approach to sentiment analysis relies on the use of training machine learning algorithms for sentiment polarity classification.

The following seven classifiers were used, in combination with either count vectorization or tf-idf tokenization, to classify the documents as positive, neutral or negative: Logistical Regression, Perceptron, linear super vector classifier (Linear SVC), AdaBoost, Decision Tree, Random Forest and Nearest Centroid.

These classifiers were trained with the reviews found in the dataset and with the ratings label as the output using the default parameters found in the scikit-learn library. The count vectorization and tf-idf tokenization techniques were compared, as well as using one-gram or up to four-grams in the tokenization. The best two classifiers based on accuracy were chosen and its hyperparameters were optimized.

2) Artificial Neural Networks: A further approach to sentiment classification is through the use of Artificial Neural Networks (ANN). ANNs is a connectionist model inspired by interconnected neurons in biological systems where its architecture and style of processing were used as motivation to let the machine learn the inner relations of data [29]. Since 2012 deep learning has taken over the machine learning field in the quest of trying to explore more advanced learning methods.

ANN's complexity and computational cost increase with the number of input features and therefore using an sparse matrix as input might be problematic. Therefore, to apply ANN another way of text encoding has to be explored and Word embedding becomes useful. In word embedding words from the vocabulary are mapped to vectors of real numbers and tries to do it in a way that similar words are close and opposite words are far from each other.

Fig 5 shows how the word embedding works and are able to learn the relation between words trying to build certain functions that can better represent the inner connections between them.

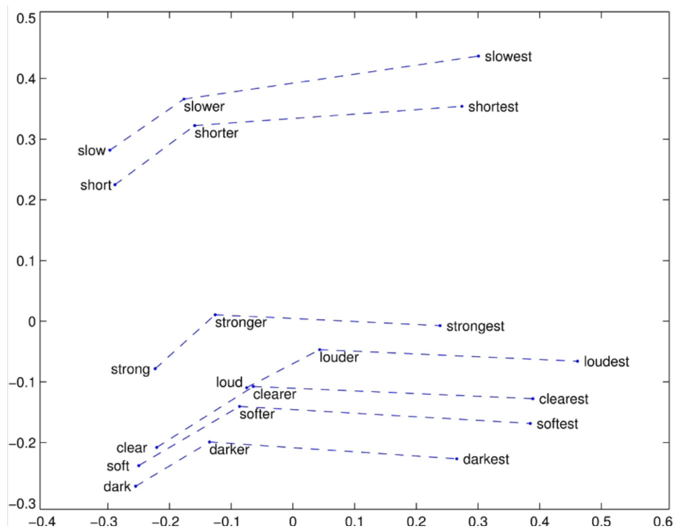


Fig. 5. Vectorization (comparative and superlative words)

Three different methods were used for the word embedding or word vectorization. First, a word embedding based on the documents found on the review corpus was trained and labeled ("No-Glove"). The second method was based on using the words present on the review corpus that also appear on the *GloVe* word embedding ("Glove-No Train"). *GloVe* is a pre-trained vector model developed by researchers in Stanford, built based on the Wikipedia 2014 and Gigaword 5 corpus (specifically the 100 dimension version) and contains 6 Billion tokens with a total vocabulary of 420k words [30]. The third approach is based on using the same *GloVe* word embedding technique as in the second method, but allowing a further training of the embeddings based on the relations present in the dataset's reviews ("GloVe-Train").

With these three possible inputs, four ANN models were used. The first model was a multi-layer perceptron (MLP) with a 128 neurons hidden layer, as shown in Figure 6. This model used a sigmoid activation in the Dense layer, followed by the pooling layer, a dropout layer with a dropout value of 0.25 and finished with a dense layer with a softmax activation function.

The second model was a Long Short Term Memory (LSTM) ANN, as shown in Figure 7. LSTM is a special type or Recurrent Neural Network with loops between the layers that allow information to persist. The LSTM ANN has the advantage that it can remember past states and use that information for the current state being evaluated.

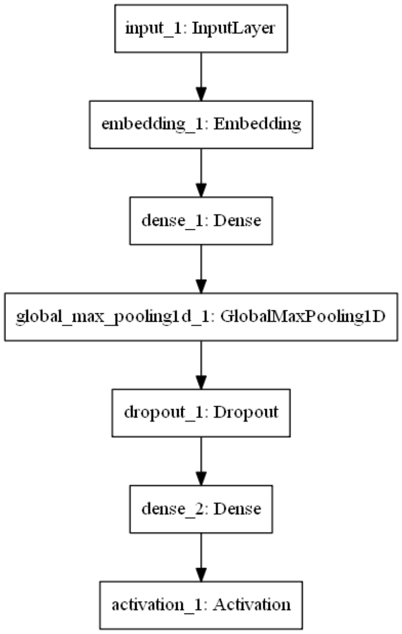


Fig. 6. MLP Model

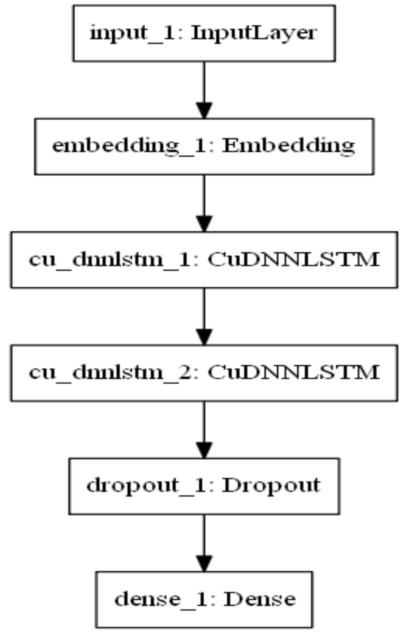


Fig. 8. Two layer LSTM Model

This property helps the ANN get a sense of context similar to the way humans remember and connect the words in a text.

This model had a 200 neurons LSTM layer, followed by a dropout layer with a dropout value of 0.5 and finished with a dense layer with a sigmoid activation function.

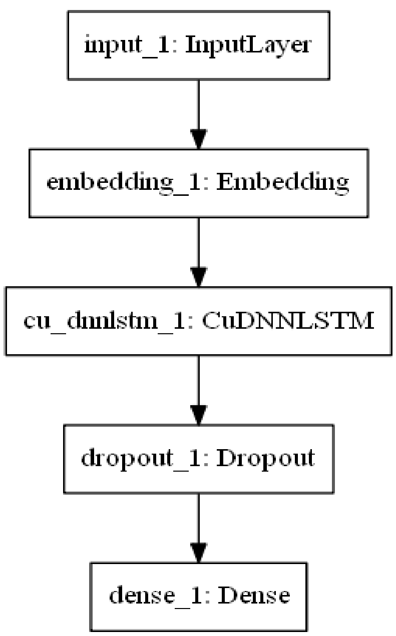


Fig. 7. One Layer LSTM Model

The third model was a two layer LSTM model based on Aleksandr Sboev's work in 2017 [31] and can be seen in Figure 8. The model considered in this work used two LSTM layers with 200 neurons each, followed by a dropout layer with a dropout value of 0.5 and finished with a dense layer with a softmax activation function.

The fourth and last model was based on Zhang and Wallace's work in 2015, in which they built a Convolutional Neural Network (CNN) model following the logic that as adjacent pixels share information that is important, words might also have a similar relation [32]. The evaluated model can be seen in Figure 9.

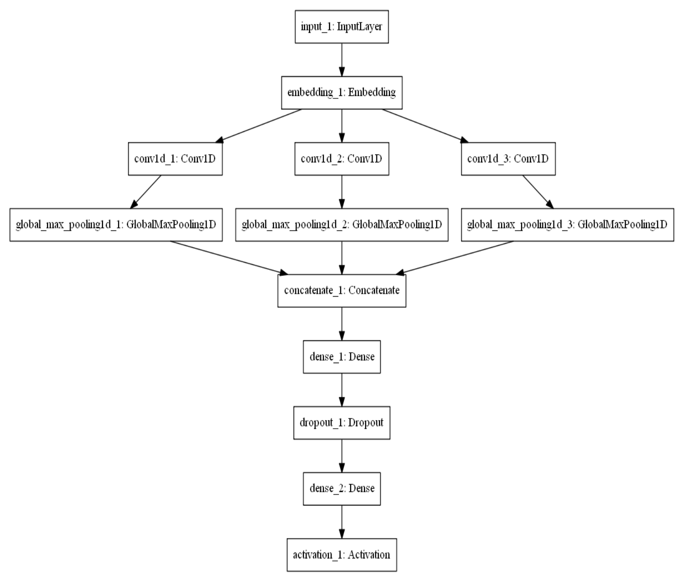


Fig. 9. CNN Model

The idea behind Zhang and Wallace's work is that a one dimensional convolutional layer might act similar to the way that n-grams work in words, and thus recreate the effect of words preceding and following a particular word that is being assessed. The model considered used three separate 1 dimensional convolutional layers with *ReLu* activation functions, with filter values of 2, 3 and 4, with one value for each filter. Afterwards, each layer has a maxpooling layer and is subsequently concatenated in the next layer. Next, a dense layer with and a *ReLu* activation function is applied to the model, followed by a dropout layer with a dropout value of 0.5 and a dense layer with another *ReLu* activation function and an L2 kernel regularization with a value of 0.2

IV. Results and Discussion

A. Lexicon-based Sentiment Analysis

Figure 10 shows the results obtained when *VADER* is used as the sentiment classifier. The overall *VADER* accuracy is about 56% which is better than a random classifier but is far from the goal accuracy of 92.24%. The classifier works well when dealing with positive reviews, but has difficulty classifying neutral reviews.

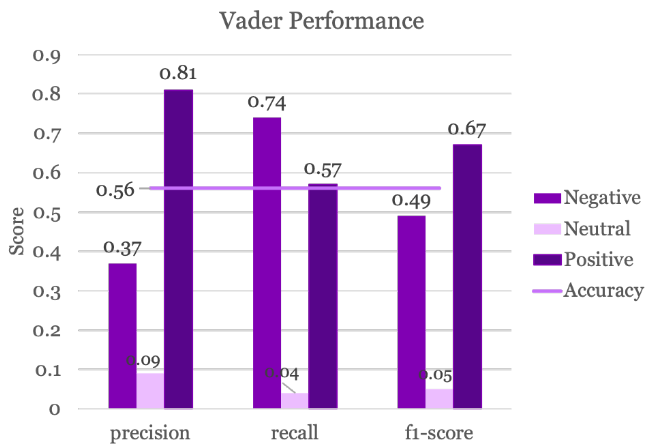


Fig. 10. Accuracy of the *VADER* approach

To sum up, *VADER* does not achieve the target accuracy when dealing with the drug reviews in the dataset. First, as was mentioned before, *VADER* is pre-trained which means that it might not be able to handle new words, such as symptoms and diseases, and thus can misclassify some review. This might mean that *VADER*'s accuracy is domain specific, and thus cannot be applied to any text document without testing its performance first.

Another possibility of the reasonably bad performance that *VADER* had is that it is not sensitive to verb tenses. This means that *VADER* can't tell the difference between a past sentiment and a current sentiment. When doing drug reviews, patients tend to explain their past symptoms and conditions before staring to take the drug and use what *VADER* might consider as a negative sentiment. Nevertheless, the review might be positive because the medication made the symptoms disappear. An example can be seen in the review on Figure 1.

Currently, most of the lexicon-based work is done using an automatic construction of new lexicon models with the help of machine learning methods and a specific dataset. This approach would certainly have a better performance because the lexicon constructed will be specific for the particular application, but at the same time, it sacrifice *VADER*'s simplicity since a training is needed.

B. Supervised learning approach

Figure 11 displays the obtained accuracy using the seven machine learning classifiers evaluated with the different tokenization methods (count vectorization and TF-IDF) while using the 1-gram approach. Among the seven methods assessed, the linear SVC and the decision tree achieved the best results. Nonetheless, the maximum accuracy obtained with this classifiers was 83.2%, which still has room to improve.

Three strategies could be used to improve the accuracy:

- Use N-grams to have a grasp on the text structure.
- Optimize the hyper-parameters of the algorithms.
- Explore more complex classifiers.

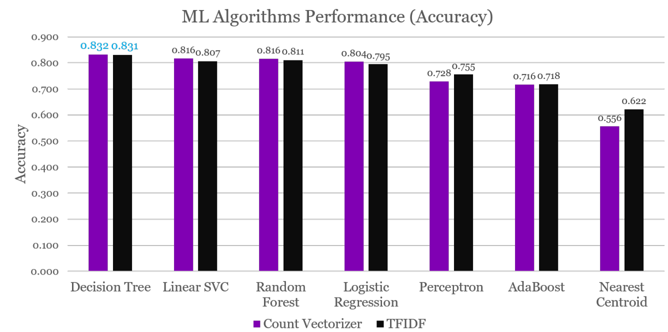


Fig. 11. Accuracy values for the different Machine learning algorithms using 1-gram

First, n-gram was performed as a strategy to get a better accuracy. Up to 4 n-grams were implemented for the classification, and the results can be seen in Figure 12. The best results were obtained using linear SVC with tf-idf vectorization and logistic regression with count vectorization, with an accuracy of 91.8% and 91.6% respectively. It is worth noting that each of these results was obtained using a different tokenization technique, which would mean that there is no clear advantage of using one over the other and thus the tokenization technique should always be considered when doing sentiment analysis.

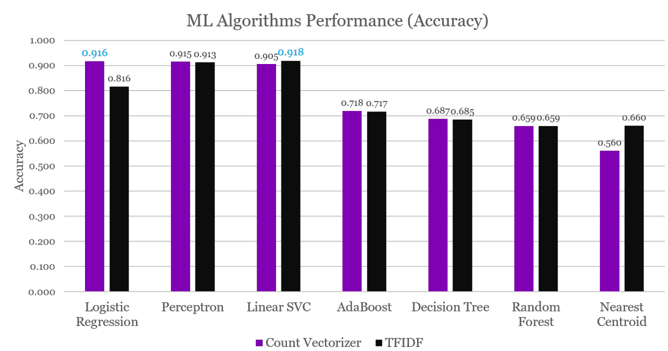


Fig. 12. Accuracy values for the different Machine learning algorithms using n-gram

The achieved performance with the n-grams increased impressively being at only 0.6% from the target accuracy. The following strategy used to try to improve the results was to tune the hyper-parameters of the two classifiers that best performed.

For the LinearSVC, C was the hyper-parameter to tune. The value of C tells the LinearSVC how soft or hard the margin should be. A higher value of C penalizes more the misclassified data points and makes the boundary harder. A high value of C makes the classifier to be more sensitive to noise, make it unsolvable because it doe not find a hyper-plane to perfectly separate the samples, or make it more prone to overfitting.

For the logistic regression, the hyper-parameter tuned was the, which in this case, is the inverse of regularization strength. The smaller the C the less likely are the parameters to be perturbed by small variations in the input data and the less likely it is that the classifier will overfit to the training data.

The tuning of the hyper-parameter C for the was made with a logarithmic scale from 0.001 to 100 and then tuned to the first decimal. The value for C with the best performance was $C = 4.9$ and $C = 2.8$ for Linear SVC and Logistic regression respectively. Unfortunately, the improvement in the accuracy was small and neglectable.

For the ANN classification, the results for each tested classifier with each word embedding technique can be seen in Figure 13, with the best result for each classifier highlighted in blue.

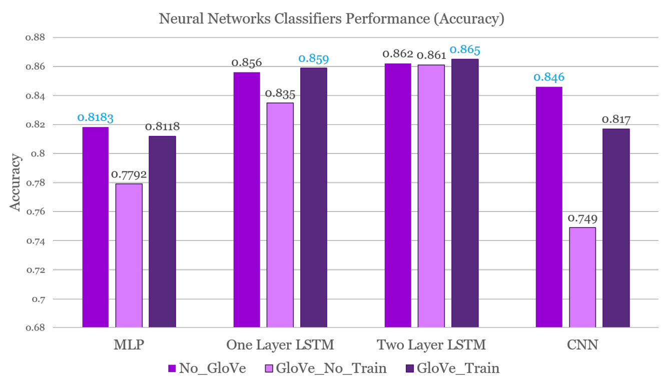


Fig. 13. Accuracy values for the different ANN techniques with its distinct word embedding approach

The best result for MLP, One Layer LSTM, Two Layer LSTM and CNN were 81.83%, 85.9%, 86.5%, and 84.6%, respectively. The best results were obtained while using no *GloVe* embedding or while using *GloVe* embedding with training. This might indicate that even though the *GloVe* model is a good approach at not having to train a group of word embeddings, some domain specific information might be needed in order to make a better sentiment prediction. Furthermore, even though the results are good compared to the other machine learning classifiers that were previously tested, no ANN technique outperformed the previous results using the tokenization approach.

In accordance with text classification research the LSTM models should outperform the MLP and CNN techniques, and this works confirmed it as seen in the figure. The two layer LSTM's accuracy also surpassed the score obtained by the one layer LSTM, which is again what was expected from the literature review. The CNN approach outperformed the MLP classifier, consistent with the theory behind the way the classifier was devised, in which it is able to represent some relations between the words found in a document.

Even though a drop-out layer and a kernel regularization were included, the ANN models seemed to do overfitting after a few epochs and this might explain why the performance of the models was lower than what was expected. Lowering the learning rate of the algorithm might help prevent overfitting. Further testing to tune the dropout and more extensive use of regularization layers are also recommended.

A summary of the best techniques of each approach performed in this work can be seen in Figure 14

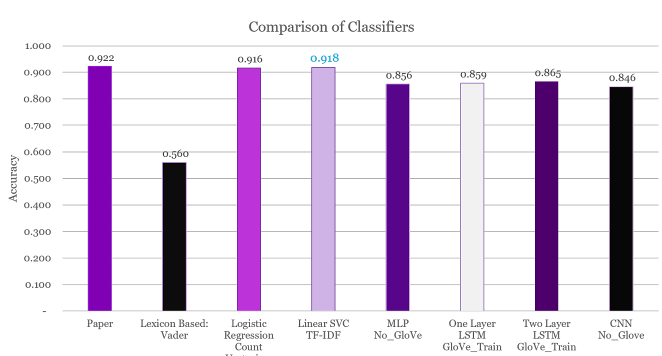


Fig. 14. Accuracy values for the best techniques of each evaluated category

Linear SVC with tf-idf tokenization and using n-grams was the best sentiment analysis classifier, with an accuracy of 91.8%, followed by Logistic regression using count vectorization using n-grams and afterwards by the two layer LSTM, with an accuracy of 91.6% and 86.5%, respectively. Although the accuracy reported by Grer et al. in [1] was not achieved, the attained accuracy was close by a margin of 0.4%. The classifier reported by them as the most accurate was a Logistical Regressor, which coincides with the second best found in this work. The worst performing classifier with an accuracy of 56% was the lexicon based VADER.

C. Conclusion and reflection

Different approaches to the sentiment classification on the reviews of the *drugs.com* dataset were analyzed on this work. We used the dataset to do the polarity classification among the negative, positive and neutral classes attributed to the reviews and their corresponding rating. The accuracy of 92.24% achieved by Grer's work used as baseline [1]. First, lexicon-based approach was studied and a representative examples was given using *VADER*. Other techniques evaluated also involved the use of different tokenization techniques in combination with the use of 1-gram and up to 4-gram combinations, as well as seven different machine learning algorithms. Further fine tuning of the classifiers' hyper-parameters was made to enhance the performance. Artificial neural networks models were also evaluated, in combination with the use of a word embedding specific to the dataset and using the *GloVe* word embedding, with and without trainable weights. The best results were obtained using traditional machine learning techniques.

Lexicon based approaches like VADER might not work as expected, as it seems it is a domain specific approach. VADER was easy to use and other lexicon models created by experts or manually trained using the specific domain can be directly and effortless used. However, once the model is trained it becomes static and blindly relying in it can be inconvenient.

Basic machine learning models view all the characters as isolated elements, and can not reflect the text connection and structure. Using the n-gram approach helps to get a grasp of the text construct and it did enhance the performance of the classifiers.

Artificial neural network techniques were evaluated but did not perform to their expectation, possibly due to an overfitting problem that will be assessed in further work. A Multi-Layer Perceptron, as well as two Long Short Term Memory recurrent neural networks and a convolutional neural network approach were evaluated. The best accuracy of the different models was obtained using the LSTM algorithms, followed by the CNN and finally the MLP. This coincides with the expected performance of each model in relation to their capacity from text analysis.

Further work should include tuning the ANNs hyper-parameters to prevent overfitting. Performing feature selection before applying the classification algorithm is another interesting future work that can help to identify the words that better describe the sentiment of the text. To improve the lexicon-based approach, it can be compelling to find a method that can identify and analyze the verb tense of a text and how the sentiment is changed by it. Another topic that is worth exploring is to use an ensemble of neural networks to increase the accuracy obtained by the models. Further work could also focus in the use of a hybrid approach, in which traditional machine learning approaches are combined with ANN. Finally, after refining the sentiment analysis algorithm, some testing on the overall generalization of the algorithm to see if it can be applied cross-domain can be interesting.

During this work, a better understanding of sentiment analysis was acquired, and an updated review of the past breakthrough works and challenges current researchers are working on in this field was performed. Some Natural Language Processing methods and text mining techniques were learned and many more were introduced. This work served as an opportunity to consolidate the knowledge learned in class.

References

[1] Grer, Felix, et al. "Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning." Proceedings of the 2018 International Conference on Digital Health. ACM, 2018.

[2] Mishra, Arpit, Ankit Malviya, and Sanchit Aggarwal. "Towards automatic pharmacovigilance: Analysing patient reviews and sentiment on oncological drugs." 2015 IEEE International Conference on Data Mining Workshop (ICDMW). IEEE, 2015.

[3] B. Liu, Sentiment Analysis and Opinion Mining, San Rafael, California, USA: Morgan and Claypool Publishers, 2012.

[4] Martnez-Cmara, Eugenio, et al. "Sentiment analysis in Twitter." Natural Language Engineering 20.1 (2014): 1-28.

[5] Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.

[6] Turney, Peter D., and Michael L. Littman. "Measuring praise and criticism: Inference of semantic orientation from association." ACM Transactions on Information Systems (TOIS) 21.4 (2003): 315-346.

[7] Beineke, Philip, et al. "Exploring sentiment summarization." Proceedings of the AAAI spring symposium on exploring attitude and affect in text: theories and applications. Vol. 39. Palo Alto, CA: The AAAI Press, 2004.

[8] Knig, Arnd Christian, and Eric Brill. "Reducing the human overhead in text categorization." Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006.

[9] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," Foundations and Trends in Information Retrieval, vol. 2, no. 1 - 2, pp. 1 - 135, 2008.

[10] Korkontzelos, Ioannis, et al. "Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts." Journal of biomedical informatics 62 (2016): 148-158.

[11] Jiang, Ling, and Christopher C. Yang. "User recommendation in healthcare social media by assessing user similarity in heterogeneous network." Artificial intelligence in medicine 81 (2017): 63-77.

[12] Hatzivassiloglou, Vasileios, and Kathleen R. McKeown. "Predicting the semantic orientation of adjectives." Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics. Association for Computational Linguistics, 1997.

[13] Kim, Soo-Min, and Eduard Hovy. "Determining the sentiment of opinions." Proceedings of the 20th international conference on Computational Linguistics. Association for Computational Linguistics, 2004.

[14] Wilson, Theresa, et al. "OpinionFinder: A system for subjectivity analysis." Proceedings of HLT/EMNLP 2005 Interactive Demonstrations. 2005.

[15] Pang, Bo, and Lillian Lee. "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts." Proceedings of the 42nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2004.

[16] Nasukawa, Tetsuya, and Jeonghee Yi. "Sentiment analysis: Capturing favorability using natural language processing." Proceedings of the 2nd international conference on Knowledge capture. ACM, 2003.

[17] Kanayama, Hiroshi, Tetsuya Nasukawa, and Hideo Watanabe. "Deeper sentiment analysis using machine translation technology." COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics. 2004.

[18] Goeuriot, Lorraine, et al. "Sentiment lexicons for health-related opinion mining." Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium. ACM, 2012.

[19] Leaman, Robert, et al. "Towards internet-age pharmacovigilance: extracting adverse drug reactions from user posts to health-related social networks." Proceedings of the 2010 workshop on biomedical natural language processing. Association for Computational Linguistics, 2010.

[20] Gopalakrishnan, Vinodhini, and Chandrasekaran Ramaswamy. "Patient opinion mining to analyze drugs satisfaction using supervised learning." Journal of applied research and technology 15.4 (2017): 311-319.

[21] Maas, Andrew L., et al. "Learning word vectors for sentiment analysis." Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1. Association for Computational Linguistics, 2011.

[22] Ye, Qiang, Ziqiong Zhang, and Rob Law. "Sentiment classification of online reviews to travel destinations by supervised machine learning approaches." Expert systems with applications 36.3 (2009): 6527-6535.

[23] Thompson, Joseph J., et al. "Sentiment analysis of player chat messaging in the video game StarCraft 2: Extending a lexicon-based model." Knowledge-Based Systems 137 (2017): 149-162.

[24] Molina-Gonzlez, M. Dolores, et al. "A Spanish semantic orientation approach to domain adaptation for polarity classification." Information Processing and Management 51.4 (2015): 520-531.

[25] Bollegala, Danushka, Tingting Mu, and John Yannis Goulermas. "Cross-domain sentiment classification using sentiment sensitive embeddings." IEEE Transactions on Knowledge and Data Engineering 28.2 (2016): 398-410.

[26] Zhang, Yuhong, et al. "Cross-domain sentiment classification-feature divergence, polarity divergence or both?." Pattern recognition letters 65 (2015): 44-50.

[27] Al-Moslmi, Tareq, et al. "Approaches to cross-domain sentiment analysis: A systematic literature review." IEEE Access 5 (2017): 16173-16192.

[28] NLTK Project, "nltk.sentiment.vader – NLLTK 3.4documentation," Sphinx, 17 November 2015. [Online]. Available: https://www.nltk.org/_modules/nltk/sentiment/vader.html.[Accessed3April2019].

[29] Gurney, Kevin. An introduction to neural networks. CRC press, 2014.

[30] J. Pennington, R. Socher and C. D. Manning, "GloVe: Global Vectors for Word Representation," 2014. [Online]. Available: https://nlp.stanford.edu/pubs/glove.pdf. [Accessed 4 April 2019].

[31] Sboev, Aleksandr, et al. "Deep Learning Network Models to Categorize Texts According to Author's Gender and to Identify Text Sentiment." 2016 International Conference on Computational Science and Computational Intelligence (CSCI). IEEE, 2016.

[32] Zhang, Ye, and Byron Wallace. "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification." arXiv preprint arXiv:1510.03820 (2015).