Evaluating Content Features and Classification Methods for Helpfulness Prediction of Online Reviews: Establishing a Benchmark for Portuguese

Rogério Figueredo de Sousa, Thiago Alexandre Salgueiro Pardo
Interinstitutional Center for Computational Linguistics (NILC)
Institute of Mathematical and Computer Sciences, University of São Paulo
Av. Trabalhador São Carlense, 400 – 13.566-590 – São Carlos – SP – Brazil
rogerfig@usp.br, taspardo@icmc.usp.br

Abstract

Over the years, the review helpfulness prediction task has been the subject of several works, but remains being a challenging issue in Natural Language Processing, as results vary a lot depending on the domain, on the adopted features and on the chosen classification strategy. This paper attempts to evaluate the impact of content features and classification methods for two different domains. In particular, we run our experiments for a low resource language – Portuguese –, trying to establish a benchmark for this language. We show that simple features and classical classification methods are powerful for the task of helpfulness prediction, but are largely outperformed by a convolutional neural network-based solution.

1 Introduction

The concern to facilitate users' decision-making is common in most e-commerce platforms. The possibility for customers to publicly provide product reviews is one of the consequences of this concern. This functionality allows future customers to read reviews from other customers and take their buying decision. Despite being useful, the amount of generated data is very large, making it impossible for a human to read them all. Moreover, a large part of this data can be considered unwanted, containing poorly written texts, vague opinions and texts of dubious quality (Kim et al., 2006), making it difficult to find relevant content.

The helpfulness voting functionality that some e-commerce platforms adopt tries to address the above problem, ranking the reviews and showing the most helpful ones to the customers. However, manual voting has some drawbacks, as new helpful reviews take time to get enough votes and gain a visible position. The solution is to automatically predict the helpfulness of reviews.

Despite the usefulness of the task of helpfulness prediction and its practical implications, literature has shown that it is a challenging open issue in Natural Language Processing (NLP). Performance results vary drastically across domains and there are several different features and classification methods in the area, as discussed in (Sousa and Pardo, 2021).

This paper aims to investigate such issues and to identify relevant features and methods for helpfulness prediction. We provide a qualitative and quantitative study of the impact of key content features in two different domains (apps and movies). By content features, we mean those that are related to the information that can be extracted directly from the review, such as the text and the "stars" given by the author. We also perform a comparative study of various classical and deep machine learning classifiers. We show that simple features and classical classification methods may be powerful for the task, but they are largely outperformed by a convolutional neural network-based approach, which reaches a f1-score of 0.90 for apps and 0.74 for movies. It is also relevant to cite that we run our experiments for a low resource language – Brazilian Portuguese –, bringing relevant contributions for NLP for Portuguese and establishing a benchmark for the task.

The rest of the paper is organized as follows. Section 2 shows the main related work. In Section 3, we describe the experimental setting adopted in this work. Section 4 reports the achieved results and Section 5 brings some final remarks.

2 Related Work

The main research line in review helpfulness prediction aims to predict the helpfulness score for a set of reviews. The helpfulness score is defined as shown in Equation 1 and can be used as the target for regression, binary classification, or ranking. The score regression aims to predict the helpfulness score $h \epsilon [0, 1]$. For binary classification, a threshold is applied in helpfulness score (e.g., $h > 0.5$)

Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis, pages 204 - 213
May 26, 2022 ©2022 Association for Computational Linguistics

and all reviews with a helpfulness score above the threshold are classified as helpful; otherwise, they are classified as not helpful. Review ranking seeks to order the reviews by their helpfulness according to a reference ranking.

$$h = \frac{helpful\ votes}{helpful\ votes + unhelpful\ votes} \quad (1)$$

In order to understand the helpfulness of online customer reviews, researches have performed several analyses. It is worth mentioning classical works like the ones of Kim et al. (2006) and Zhang and Varadarajan (2006) that introduce many types of features for helpfulness prediction. Kim et al. (2006) split the features in 5 categories, all considered to be content features: Structural, Lexical, Syntactic, Semantic and Meta-Data Features. They build a model for a regression task and a model for a ranking task using the SVM algorithm. Using a dataset of reviews on two products (MP3 players and Digital Cameras) extracted from Amazon.com, the best results are achieved with the combination of length, unigram and number of stars features. In a similar way, Zhang and Varadarajan (2006) propose three categories of features, also for a dataset extracted from Amazon.com. Their features include Lexical Similarity (Cosine similarity over TF-IDF vectors), Shallow Syntactic Features (Proper nouns, Modal verbs, Interjection, etc.) and Lexical Subjectivity Clues (Subjective adjectives, Subjective nouns, etc.). The authors model two regressors using SVR (Support Vector Regression) and SLR (Simple Linear Regression) techniques, obtaining the best results by combining all the features.

Zeng et al. (2014), in addition to the features already used by Kim et al. (2006), propose the use of Trigrams, Comparison Expressions ("Compare to" or "ADJ + er than"), Degree of detail and Pros and Cons. Using an SVM classifier, the authors address the helpfulness prediction task as a three-class classification: Helpful positive reviews, Helpful negative reviews, and Unhelpful reviews. Furthermore, by running a series of experiments with one less feature each time, they found that the "detail" feature is the most important one, followed by length, number of stars and unigram.

More recently, researchers are using more robust methods for helpfulness prediction. It is the case of Xu et al. (2020), that use BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) along with the features of Star Rating and Product Type. With this combination, the authors model a Neural Network to predict the helpfulness score for reviews extracted from Amazom.com. Wang et al. (2020) also use BERT, but the authors add more features (Number of Words, Number of Sentences, Rating, etc.) than Xu et al. (2020) and compare the BERT-based approach to SVM and CNN models. The neural network-based classifiers achieved similar results to SVM using all features. Wu and Wang (2019) propose the use of syntactic features along with BERT sentence embeddings to helpfulness classification. The work compares some CNN models with BERT and perform an ablation study with all syntactic features. Their results showed high recall but very low precision values. In terms of f1-score, BERT achieved the best results and the main feature was Star Rating.

All these researches have in common the use of content features. The results of methods using handcrafted features were better or very close to state-of-the-art classifiers (using BERT and CNN, for instance). In such setting, this paper aims at further exploring such issues, specially for the context of Portuguese, a low resource language. We present our experiment setting in what follows.

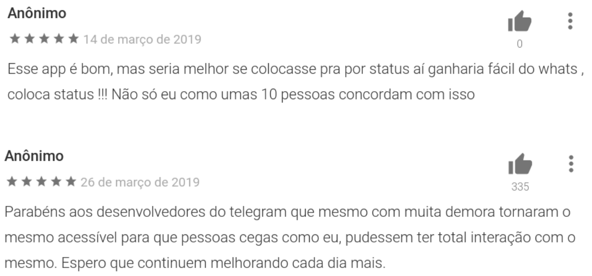
3 Experiment Setting

3.1 Data Overview

We adopt the dataset of Sousa et al. (2019) that includes reviews written in Portuguese for two very different domains: Movies and Apps. While movie reviews are usually largely subjective and passionate, app reviews tend to be more objective and focus on technical aspects. The dataset (namely UTLCorpus) contains a total of $2,732,538$ reviews ($1,833,691$ for movies and $898,847$ for apps).

Figure 1 presents two examples of reviews extracted from the corpus (from the apps domain). The first is considered not helpful, while the second is helpful. According to the creators of the corpus, the helpfulness status is based on the number of votes the reviews received (0 and 335 helpful votes, respectively) and the posting time (more than 5 days).

As the authors report, each review includes the review text, number of stars given by its author, the number of helpfulness votes, and publication time, among some other information. As shown in Table 1, the UTLCorpus is highly unbalanced. We address the unbalancing problem using an under-



Anônimo

★★★★★ 14 de março de 2019

0

Esse app é bom, mas seria melhor se colocasse pra por status aí ganharia fácil do whats , coloca status !!! Não só eu como umas 10 pessoas concordam com isso

Anônimo

★★★★★ 26 de março de 2019

335

Parabéns aos desenvolvedores do telegram que mesmo com muita demora tornaram o mesmo acessível para que pessoas cegas como eu, pudessem ter total interação com o mesmo. Espero que continuem melhorando cada dia mais.

Figure 1: Examples of reviews

sampling approach, randomly removing samples of the majority class. Due to the amount of data, we decided not to carry out the oversampling strategy. Besides the class balancing information, the details of tokens and types in the table show us that the average size of movie reviews is much bigger than that of apps. This difference can make the movies' reviews more challenging than the apps' reviews. Section 4 will further elucidate this assumption.

	Movies	**Apps**
# reviews	1, 833, 691	898, 847
# movies or apps	4, 283	243
# types	1, 828, 647	419, 713
# tokens	60, 177, 264	11, 919, 636
Avg. of Tokens p/ doc	32.7994	12.9384
Helpfulness Label	*helpful: 20%*	*helpful: 5%*

Table 1: UTLCorpus numbers. The helpfulness label refers to the percentage of reviews labeled as helpful.

For our experiments, which we report in the next section, we have randomly split our dataset in three parts: 70% for training, 20% for testing, and 10% for development.

3.2 Features

The literature on online review helpfulness explores several features. The researchers often split the features in two big groups: Content and Context features. The content features are related to the information that can be extracted directly from the review, such as the text and the "stars" given by the author. Context features are those extracted from outside the review, such as reviewer information. (Ocampo Diaz and Ng, 2018; Almutairi et al., 2019; Arif et al., 2018). Most of these features are used in domains such as products, books, hotels and so on. We desire to experiment them in apps and movies domains, which are the domains available in the dataset that we adopted in this work and that are remarkably different (which interests us in this paper).

We selected and adapted several content features to the Portuguese language. This process involved finding resources and tools that could support the use of the features in the target language. Table 2 summarizes the implemented features.

We explored the features in machine learning classification solutions. We performed a selection of the best features employing three different strategies. The first method of feature selection is the classical Information Gain (Kozachenko and Leonenko, 1987), which produces values from 0 (no information) to 1 (maximum information) for each feature. The features that contribute with more information are selected to the experiments. The second well-known method for feature selection is using the Random Forest classifier (Breiman, 2001), which is a meta estimator that uses several tree-based classifiers in various subsamples of the dataset to classify the target. Due to its characteristic of using decision trees, it can indicate the importance of features used in the classification process. The third method for feature selection consists in using the correlation values of the features with the helpfulness classes. The previous work of Sousa and Pardo (2021) presents studies of correlation among the feature values and helpfulness status using the correlation coefficients of Pearson and Spearman. Using these correlations, we order the absolute values and select the features with better values.

In addition to the previous features, we also test Term-Frequency (TF) and Term Frequency-Inverse Document Frequency (TF-IDF) techniques to generate specific text features and compare the results of the handcrafted features with these two well-known baseline features. It is important to mention that all feature values were normalized for the experimentation process. Table 3 shows an overview of all the features used in this paper.

We comment on the machine learning classifiers and report the achieved results in the next section.

4 Results

We explored the following classical classification strategies in this work: Naive Bayes (NB), Support Vector Machines (SVM), Decision Tree (DT), Random Forest (RF), Neural Network Multilayer Perceptron (NN) and a Dummy Classifier. More sophisticated (deep) strategies that we tested are a BERT-based classifier and a Convolutional Neural

Feature	Description	Portuguese Resource/Tool
Average Sentence Length (Avg-SL)	Average sentence size in terms of words (Liu et al., 2007; Lu et al., 2010)	spaCy with portuguese language model
Number of Sentences (Num-S)	Total of sentences in the review (Liu et al., 2007; Lu et al., 2010)	
Number of Words (Num-W)	Total of words in the review (Kim et al., 2006; Mudambi and Schuff, 2010)	
Star Rating (Star-R)	The review-assigned product star rating (Huang et al., 2015)	-
Readability Features (READ)	Measure how easy a text is to read and include the following features: Automated Readability Index (ARI), Coleman-Liau Index (CLI), Flesch Reading Ease (FRE), Flesch-Kincaid Grade Level (FKGL), Gunning fog index (GFI) and SMOG (Dubay, 2004; Ghose and Ipeirotis, 2011)	Readability features based on (Antunes and Lopes, 2019)
Spelling Errors (SPELL)	Number of misspelled words in review (Ghose and Ipeirotis, 2011)	Number of words not found in Wiktionary[1] and Unitex-PB lexicons (Muniz, 2004)
Dominant Terms (Dom-Terms)	Presence of important terms in reviews, considering their specificity for the domain (Tsur and Rappoport, 2009)	We use the NILC Corpus (Nunes et al., 1996) to calculate the frequencies of words that do not belong to the domains
Product Aspects (Prod-Feat)	Presence of product aspects in the reviews (Kim et al., 2006; Hong et al., 2012; Liu et al., 2007)	We manually extract the features of texts from the corpus development set.
Sentiment Words (SENT)	Number of words that express sentiments (Kim et al., 2006) according to the following categories of the LIWC dictionary (Pennebaker et al., 2001): Negate, Swear, Affect, Posemo, Negemo, Anxiety, Anger and Sad	We used a Portuguese version of LIWC dictionary (Balage Filho et al., 2013)
Sentiment Divergence (Sent-Div)	Difference between the general sentiment about the movie/app and the sentiment expressed by the author of a review (Hong et al., 2012)	Sentilex sentiment lexicon (Silva et al., 2012)
Subjectivity (SUB)	The probability of a review being subjective (Ghose and Ipeirotis, 2011)	
Morpho-Syntactic Tokens (SYN)	Number of tokens with the following Part-of-Speech tags: Noun (N), Verb (V), Adverb (ADV) and Adjective (ADJ). It also includes counting for open class words (Open) (Kim et al., 2006)	NLPNet POS-Tagger (Fonseca and Rosa, 2013)
Star Deviation (Star-Dev)	Difference between the number of stars in a review and the average star rating for the movie/app (Hong et al., 2012)	-

Table 2: List of content features

Network (CNN).

4.1 Feature Selection

As explained before, we performed feature selection using the techniques of Information Gain and Random Forest. Figures 2a and 2b show the results of feature ranking for the apps domain, while Figure 2c and 2d show the results for movies domain. We performed the classification for the top 8 features of each method of feature selection. As an alternative, we also selected the most correlated features to helpfulness status using the Pearson and Spearman values.

4.2 Classification Results

We divided the process of training classifiers in some distinct phases. In the first phase, we trained the classifiers considering the feature selection methods against the TF and TF-IDF techniques. This phase shows us the best sets of features and the best classifiers for both types of features: handcrafted and TF/TF-IDF features. In the second phase, we merged the handcrafted features with the TF/TF-IDF ones. This feature combination process

Feature category (number of features)	Description
Handcrafted Content Features (29)	The content features adapted from previous literature works.
Information Gain (8)	The handcrafted content features selected by Information Gain technique.
Random Forest (8)	The handcrafted content features selected by Random Forest Classifier.
Correlation Coefficients (8)	The handcrafted content features selected by the intersection of correlation coefficients.
Baseline TF (500)	The features selected by TF method.
Baseline TF-IDF (500)	The features selected by TF-IDF method.

Table 3: Overview of the features

consists of concatenating the vectors of each text (i.e., TF or TF-IDF vectors) with the vectors of each group of features, both with the same weight. Finally, in the third phase, we decided to use the results of the second phase to model voting-based ensemble classifiers. The classifiers with good results and fewer errors in common were selected to compose the ensembles. The chosen classifiers for the ensembles were Decision Trees and Neural Networks for apps, and Decision Trees and Random Forest for movies. Ensembles with three classifiers obtained similar results (never higher) to those with two classifiers, so we only report the results for ensembles of two classifiers[2]. Finally, in a fourth phase, we used a BERT-based classifier over a pre-trained Portuguese model (Souza et al., 2020) for both domains and a CNN using the GloVe[3] (Hartmann et al., 2017; Pennington et al., 2014) embeddings as input features.

The results referring to the first phase are shown in Figures 3a and 3b, where we show F1 scores (the best ones are written in the chart). Notice that we show in the charts the *F1-Measure* that is the average F1 score for the two classes. One may see that, for apps, the best results were 72%, which may be achieved with simple TF features with SVM and Random Forest; for movies, the best results were 63% for TF-IDF, with the same classifiers. Overall, for both domains, there were no significant performance differences for the two classes.

When we merge the two big groups of features (handcrafted and TF/IDF features), the results are better, as one may see in Figures 3c and 3d. Considering the best situation, apps classification achieved 78% with correlation-based feature selection and TF for SVM (results 8.3% better than before); movies achieved 66% with all the features and TF-IDF for SVM too (4.7% better). Again, SVM showed to be a distinctive technique, with stable classification performances for the two classes.

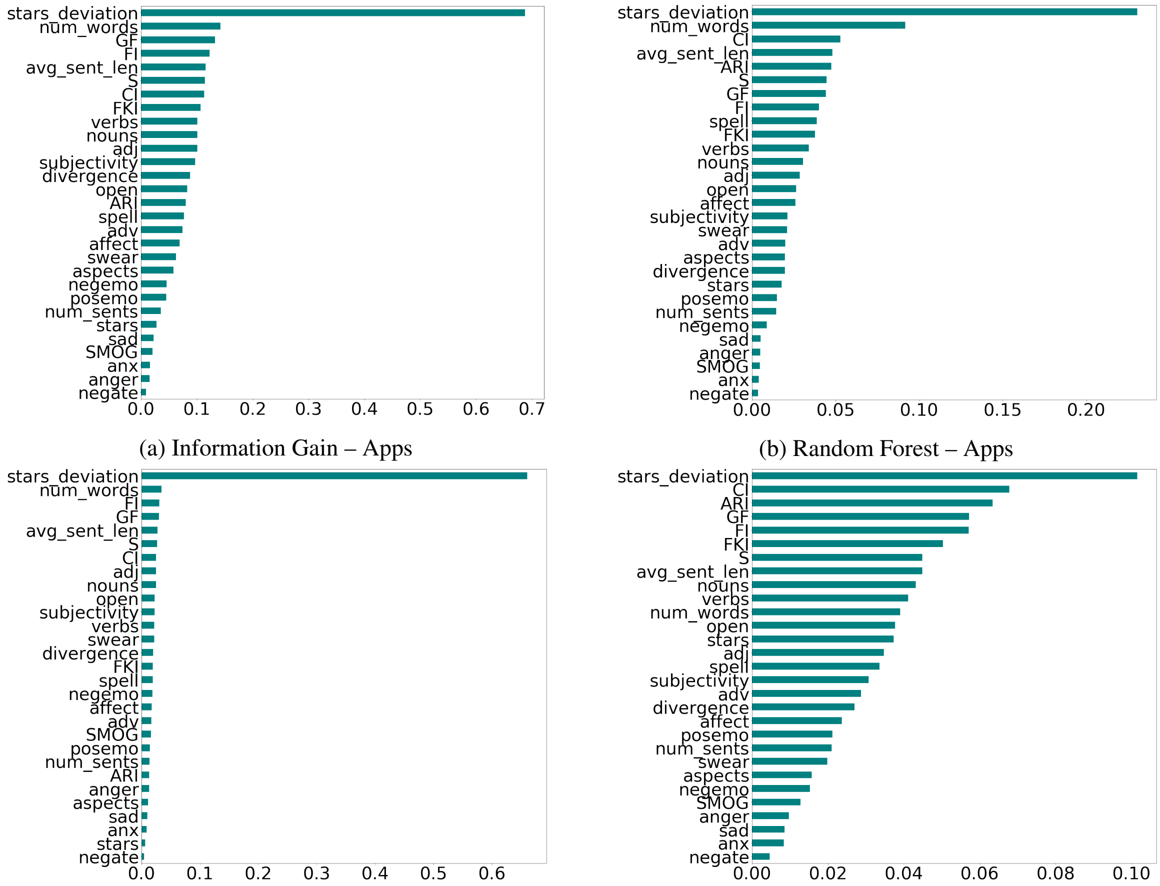
The results for our ensemble, the BERT-based[4] and the CNN classifiers are shown in Figure 3e. For better understanding, the X-axis in Figure 3e mentions the use of the handcrafted features along with BERT (BERT-PT+Hand). For this strategy, we appended all handcrafted features to *CLS* vector ($768 + 29\ dimensions$), and then the method proceeds normally, using the resulting vector in the next layer to perform the classification. In the same way, for clarification, the strategy BERT-PT+CNN was modeled to merge the BERT architecture to CNN, presented before. We used the four last layers of BERT as features for CNN. The fine-tuning of BERT model was made at the same time as the CNN training. Figure 4 shows the architecture of the CNN.

Despite BERT being a new standard technique in the NLP area, it achieved results very similar to those presented by the ensemble. In the application domain, BERT shows a slight drop in performance. Further investigation is needed to find out why the

[2] We adopted a soft classification, in which the classes are weighted by their probabilities given by the classifiers; if it happens that the two classes end up with the same score, we opt for the not helpful class.

[3] http://nilc.icmc.usp.br/nilc/index.php/repositorio-de-word-embeddings-do-nilc

[4] This model was fine-tuned and the pre-trained parameters were not frozen during fine-tuning. The reviews were tokenized using the default tokenizer of Bertimbau model. We applied a single layer feed forward network in CLS output vector (768 dimensions) to classify the instances. The main hyperparameters are as follows: *epochs = 2*, *learning rate = 4e-5*, *optimizer = AdamW*, *train batch size = 8*, *max sequence length = 128*. These hyperparameters were empirically chosen.



(a) Information Gain – Apps

(b) Random Forest – Apps

(c) Information Gain – Movies

(d) Random Forest – Movies

Figure 2: Results of feature importance

results are so low for this case. Possible explanations include the more “passionate” and subjective nature of the movie reviews (while apps’ reviews tend to discuss more “technical” aspects). Overall, the ensemble classification could not outperform the previous experiments, while the CNN model outperformed all classifiers.

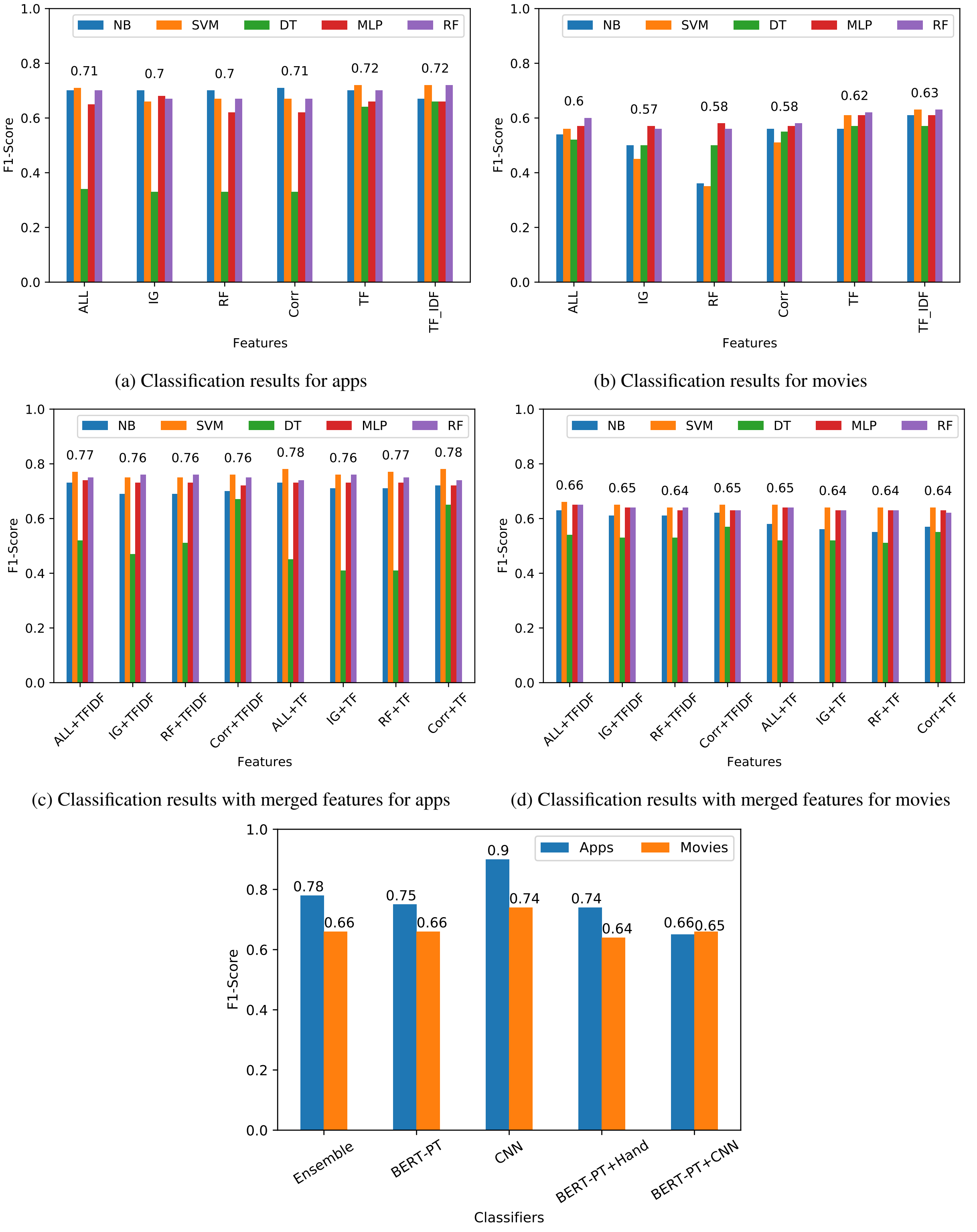
Considering all the experiments, we have some valuable learned lessons. We may see that simple textual features such as TF and TF-IDF may be powerful features for helpfulness prediction. However, merging handcrafted content features with TF-IDF features allows us to achieve better results. Other interesting result is that traditional machine learning techniques may rival more sophisticated strategies as ensemble or BERT-based classifiers. SVM, in special, showed to be an important technique among the classical methods. Anyway, all of them were outperformed by a CNN approach.

Finally, regarding the feature selection processes, the correlation-based one was slightly better than information gain and the Random Forest-based one, but the differences appear to be insignificant. Among the best selected features, although there is some variation depending on the used correlation measure, it is possible to highlight some of them: for apps domain, we highlight average sentence length, star rating and part of speech tags; for movies domain, average sentence length, SMOG readability score, sentiment words and dominant terms.

5 Final Remarks

This paper synthesized a series of experiments on predicting review helpfulness, showing some relevant learned lessons and contributions (in particular, for Brazilian Portuguese, which is considered a low resource language). However, a lot remains to be investigated. We highlight two issues that concern us the most at this time.

Firstly, the different performances for different domains (across different classification methods) keep intriguing us. This is a known behavior in the sentiment analysis area, and we corroborate it by testing new domains in this paper. We wonder whether new methods or features should be tested,



(a) Classification results for apps

(b) Classification results for movies

(c) Classification results with merged features for apps

(d) Classification results with merged features for movies

(e) Results of ensemble classification and deep models with their combinations

Figure 3: Classification Results

maybe focusing on those that are more domain independent, or whether we should “transform” our data, “eliminating” domain specific traits.

The other issue refers to the helpfulness prediction task itself. Although the literature (including us) have exhaustively tried with this task, it is a highly subjective task that (indirectly) incorporate several other tasks, as subjectivity classifi-

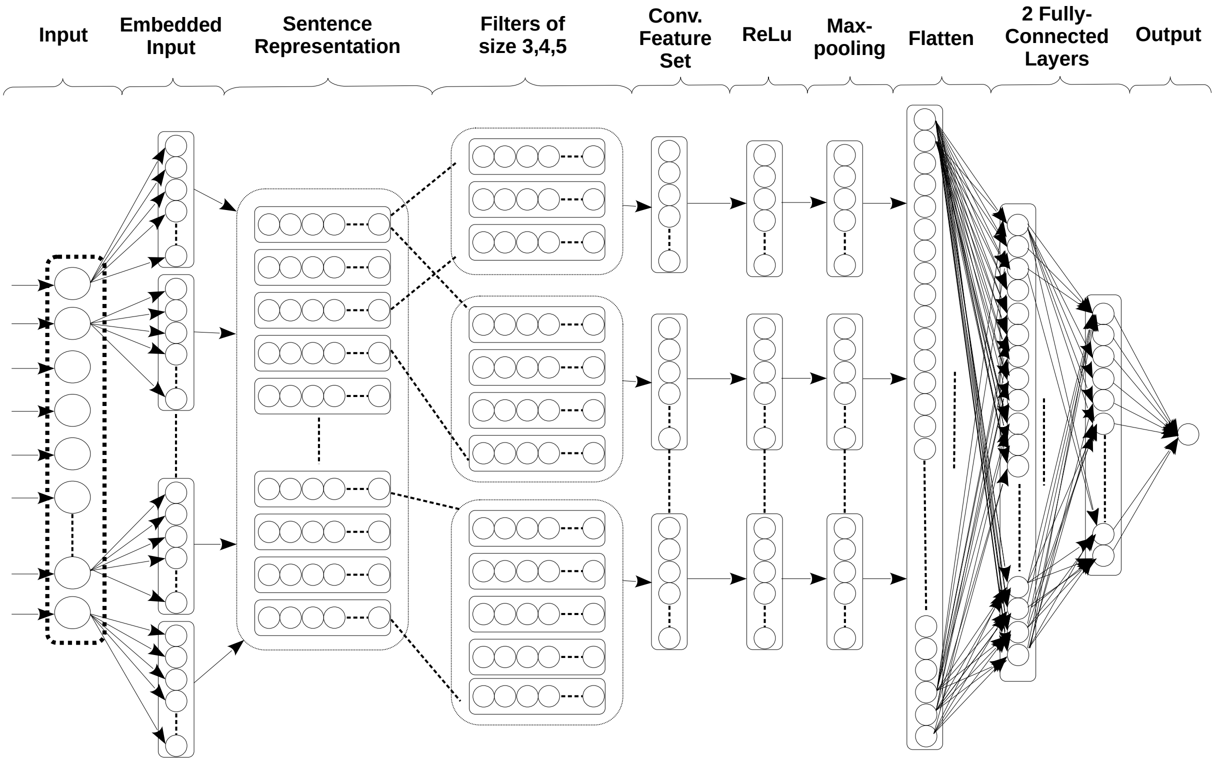


Figure 4: CNN's Architecture. We use 300-dimensional GloVe embeddings as input features. As we can see, we employ three paralels convlayers and set to 100 the size of the output channel for each convlayer. Also, the other parameters are: *epochs = 5*, *optimizer = Adam*, *batch size = 32*. Fully connected layers: input 1 = 300, output 1 = 32 and Dropout = 0.7

cation (more "personal" reviews look to be more interesting), polarity classification (more "radical" opinions call more attention), aspect identification (as reviews that directly cite some aspects look to be more useful), and detection of user information need (ultimately, a review is helpful only if it attends the information need of the user). Future efforts might explore such supporting tasks for helpfulness prediction.

The complete code for our features and models are available online at https://github.com/RogerFig/deep-helpfulness. The interested reader may also find more information at the POeTiSA project web portal (https://sites.google.com/icmc.usp.br/poetisa).

Acknowledgments

The authors are grateful to the Center for Artificial Intelligence (C4AI) of the University of São Paulo, sponsored by IBM and FAPESP (grant #2019/07665-4), and to *Instituto Federal do Piauí* (IFPI).

References

Yasamyian Almutairi, Manal Abdullah, and Dimah Alahmadi. 2019. Review helpfulness prediction: Survey. *Periodicals of Engineering and Natural Sciences*, 7(1):420–432.

Hélder Antunes and Carla Teixeira Lopes. 2019. Analyzing the adequacy of readability indicators to a non-english language. In *Proceedings of the International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 149–155. Springer.

Madeha Arif, Usman Qamar, Farhan Hassan Khan, and Saba Bashir. 2018. A survey of customer review helpfulness prediction techniques. In *Proceedings of SAI Intelligent Systems Conference*, pages 215–226. Springer.

Pedro Balage Filho, Thiago Alexandre Salgueiro Pardo, and Sandra Aluísio. 2013. An evaluation of the brazilian portuguese liwc dictionary for sentiment analysis. In *Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology*, pages 215–219.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William Dubay. 2004. The principles of readability. *CA*, 92627949:631–3309.

Erick Rocha Fonseca and João Luís G Rosa. 2013. Mac-morpho revisited: Towards robust part-of-speech tagging. In *Proceedings of the 9th Brazilian symposium in information and human language technology*, pages 98–107.

Anindya Ghose and Panagiotis G Ipeirotis. 2011. Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE Transactions on Knowledge and Data Engineering*, 23(10):1498–1512.

Nathan Hartmann, Erick Fonseca, Christopher Shulby, Marcos Treviso, Jéssica Silva, and Sandra Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology*, pages 122–131, Uberlândia, Brazil. Sociedade Brasileira de Computação.

Yu Hong, Jun Lu, Jianmin Yao, Qiaoming Zhu, and Guodong Zhou. 2012. What reviews are satisfactory: Novel features for automatic helpfulness voting. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 495–504, New York, NY, USA. ACM.

Albert H Huang, Kuanchin Chen, David C Yen, and Trang P Tran. 2015. A study of factors that contribute to online review helpfulness. *Computers in Human Behavior*, 48:17–27.

Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. 2006. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 423–430, Sydney, Australia. Association for Computational Linguistics.

LF Kozachenko and Nikolai N Leonenko. 1987. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii*, 23(2):9–16.

Jingjing Liu, Yunbo Cao, Chin-Yew Lin, Yalou Huang, and Ming Zhou. 2007. Low-quality product review detection in opinion summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 334–342, Prague, Czech Republic. Association for Computational Linguistics.

Yue Lu, Panayiotis Tsaparas, Alexandros Ntoulas, and Livia Polanyi. 2010. Exploiting social context for review quality prediction. In *Proceedings of the 19th international conference on World wide web*, pages 691–700.

Susan M Mudambi and David Schuff. 2010. Research note: What makes a helpful online review? a study of customer reviews on amazon. com. *MIS quarterly*, pages 185–200.

Marcelo Caetano Martins Muniz. 2004. *A construção de recursos lingüístico-computacionais para o português do Brasil: o projeto Unitex-PB*. Ph.D. thesis, Universidade de São Paulo.

Maria das Graças Volpe Nunes, Fabiano M Costa Vieira, Cláudia Zavaglia, Cássia RC Sossolote, and Josélia Hernandez. 1996. A construção de um léxico para o português do brasil: lições aprendidas e perspectivas. In *Anais do II Encontro para o Processamento de Português Escrito e Falado*, pages 61–70.

Gerardo Ocampo Diaz and Vincent Ng. 2018. Modeling and prediction of online product review helpfulness: A survey. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 698–708, Melbourne, Australia. Association for Computational Linguistics.

James W Pennebaker, Martha E Francis, and Roger J Booth. 2001. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Mário J Silva, Paula Carvalho, and Luís Sarmento. 2012. Building a sentiment lexicon for social judgement mining. In *Proceedings of the International Conference on Computational Processing of the Portuguese Language*, pages 218–228. Springer.

Rogério Figueredo Sousa, Henrico Bertini Brum, and Maria das Graças Volpe Nunes. 2019. A bunch of helpfulness and sentiment corpora in brazilian portuguese. In *Proceedings of the 12th Brazilian Symposium in Information and Human Language Technology*, pages 209–218. Sociedade Brasileira de Computação.

Rogério Figueredo Sousa and Thiago Alexandre Salgueiro Pardo. 2021. The challenges of modeling and predicting online review helpfulness. In *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*, pages 727–738. Sociedade Brasileira de Computação.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2020. BERTimbau: pretrained BERT models for Brazilian Portuguese. In *Proceedings of the 9th*

Brazilian Conference on Intelligent Systems, pages 403–417.

Oren Tsur and Ari Rappoport. 2009. Revrank: A fully unsupervised algorithm for selecting the most helpful book reviews. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 3, pages 154–161.

Xi Wang, Iadh Ounis, and Craig Macdonald. 2020. Negative confidence-aware weakly supervised binary classification for effective review helpfulness classification. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1565–1574.

Shih-Hung Wu and Jun-Wei Wang. 2019. Integrating neural and syntactic features on the helpfulness analysis of the online customer reviews. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 1013–1017. IEEE.

Shuzhe Xu, Salvador E Barbosa, and Don Hong. 2020. Bert feature based model for predicting the helpfulness scores of online customers reviews. In *Future of Information and Communication Conference*, pages 270–281. Springer.

Yi-Ching Zeng, Tsun Ku, Shih-Hung Wu, Liang-Pu Chen, and Gwo-Dong Chen. 2014. Modeling the helpful opinion mining of online consumer reviews as a classification problem. *International Journal of Computational Linguistics & Chinese Language Processing*, 19(2):17–32.

Zhu Zhang and Balaji Varadarajan. 2006. Utility scoring of product reviews. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, pages 51–57, New York, NY, USA. ACM.