Natural Language Processing Approaches for Accrediting Students on Extracurricular Activities

João Pedro F. M. Cavalcante[1], **Mayara C. Marinho**[2], **Vinícius R. P. Borges** [2]

[1]Departamento de Estatística – Universidade de Brasília (UnB)
Brasília – Distrito Federal (DF) – Brazil

[2]Departamento de Ciências da Computação – Universidade de Brasília (UnB)
Brasília –Distrito Federal (DF) – Brazil

jpedro.fmc@gmail.com, mayaracm@unb.br, viniciusrpb@unb.br

Abstract. *The undergraduate programs at Brazilian universities allow students to include extracurricular activities in their academic transcripts. The large amount of proof documents (certificates and declarations) submitted by students that are subsequently analyzed by the academic staff makes the accrediting of extracurricular activities time-consuming and prone to error. This paper describes a methodology to classify academic proof documents according to the pre-defined groups by the Universidade de Brasília regulations for extracurricular activities accreditation. Experimental results showed that TF-IDF with SVM outperformed BERT, CNN and BiLSTM with 0.94 average Macro F1-Score, though their performances' difference were not statistically significant.*

1. Introduction

Extracurricular activities in the academic environment are optional activities that students can engage outside of regular class hours. These activities can be offered by the educational institution or by external organizations and cover a wide range of areas. They can be an important differentiator in students' curricula vitae, demonstrating initiative, proactivity, and interest in areas beyond the classroom. These activities expose students to new areas of knowledge and different perspectives, complementing their academic training. Moreover, they are great opportunities for students to meet new people, make new contacts, and build networking [Lawhorn 2008].

Brazilian universities encourage students to gain experience through extracurricular activities, such as attending talks, conferences, workshops, and internships, as well as participating in research, tutoring, extension projects, and online courses, by offering extra credits. To obtain credits, students must submit proof documents, such as certificates and declarations. Traditionally, the accreditation process is a manual task, typically performed by the undergraduate program coordinator or academic secretariats. This process becomes particularly challenging due to the many certificates and declarations submitted by the students, resulting in a time-consuming, tedious, and error-prone procedure.

In this sense, automating the accreditation of extracurricular activities can benefit the academic administration and coordination so that they can dedicate more to activities that require greater care and cannot be automated. For this purpose, Natural Language Processing (NLP) and Machine Learning (ML) techniques can be employed,

DOI: 10.5753/sbie.2024.242548

given their increasing popularity and widespread adoption in various knowledge domains [Khurana et al. 2023][Dogra et al. 2022] [Meystre and Haug 2006].

The literature has reported some similar approaches based on ML and NLP, such as for the rule extraction from legal documents [Dragoni et al. 2018], review and evaluate the practiced classical techniques, tools, models, and systems for automatic information extraction (IE) from published scientific documents, such as research articles, patents, theses, technical reports, and case studies [Rahman et al. 2022]. The analysis of educational data has been explored as ML applications, with proposals for evaluating learning methodologies, student performance [Xiao et al. 2022] and extracting domain-specific concepts and prerequisites relations learning [Lu et al. 2019].

To the best of our knowledge, the literature has not previously explored NLP-based approaches to analyze academic documents for the accreditation of extracurricular activities in students' transcripts. This motivated us to tackle this as a classification approach using NLP models. The raw text extracted from proof documents (certificates, declarations, scientific papers, and other academic documents) can be pre-processed so that NLP models can learn the patterns of each category of documents considered in extracurricular activities regulations and make proper predictions.

For that purpose, we describe a methodology based on an NLP pipeline that extracts raw texts from declarations, certificates, and academic documents so that they are transformed into structured representations, allowing the use of ML and language models. Moreover, the lack of public corpora containing documents similar to the certificates and declarations required us to create a labeled document corpus to enable the training of the classification models.

Thus, we can describe the contributions of this research below:

- A NLP-based approach for classifying proof documents (certificates, declarations, academic documents, scientific papers, etc) by considering the extracurricular activities regulation of the Computer Science Undergraduate Program at Universidade de Brasília (*University of Brasilia*).
- A labeled corpus constituted by certificates, declarations, and other proof documents categorized into pre-defined groups of extracurricular activities according to the regulation of the Computer Science Undergraduate Program at Universidade de Brasília.

The structure of this paper is described next. Section 2 presents recent works in the literature that address similar tasks. Section 3 describes the dataset construction, the pre-processing step, and state-of-the-art methods for the task at hand. Section 4 presents the experiments, using well-known evaluation strategies for text classification, and discusses the results of the work and its limitations. Finally, Section 5 provides final considerations.

2. Related Works

ML and NLP techniques are widely used for processing semi-structured documents in several knowledge domains [Krishnamurthy et al. 2017]. In the context of accrediting extracurricular activities in educational institutions, particularly when dealing with supporting documents like certificates, academic statements, and diplomas, the literature on this topic is still scarce.

In [Heppner et al. 2019], the authors explored a semi-automatic web-based system for credit transfer evaluation and articulation agreements between higher education institutions. The utilization of an unsupervised NLP algorithm initially yielded a 71% course overlap percentage, comparable to human expert selections. To improve accuracy, a Word2Vec-based algorithm, incorporating domain-specific language and dependency analysis, increased the overlap to 86%. The study addresses key challenges in compiling a domain-specific corpus and utilizing unsupervised algorithms for semantic similarity in education. The results highlight the significant advantages of the NLP-based system, including a 50% reduction in transfer agreement development time, emphasizing the importance of a domain-specific approach for accurate evaluation in North America's educational context.

Widiastuti et al. [Widiastuti and Dewi 2020] developed a system focused on extracting images from various types of documents, such as assignments, decrees, and certificates. The design takes into account specific requirements for the particularities of each document type. The proposed system aims to facilitate the development or implementation process, enabling the retrieval of information from document images.

Hassan et al. [Hassan and Le 2020] proposed a development of an automated framework using NLP and ML techniques for the problem of requirements identification in construction contracts. The task was to classify contractual texts into two categories: requirements and non-requirements. The research involved building a dataset and manually labeling 1,787 statements, consisting of 1,388 requirement and 399 non-requirement statements. The authors utilized a processing pipeline that included text preprocessing, sentence representation, implementation of ML algorithms, and evaluation metrics.

The lack of research on processing digitized academic documents and automating the accreditation of extracurricular activities motivated us to explore an NLP-based approach for this task, described in next section.

3. Methodology

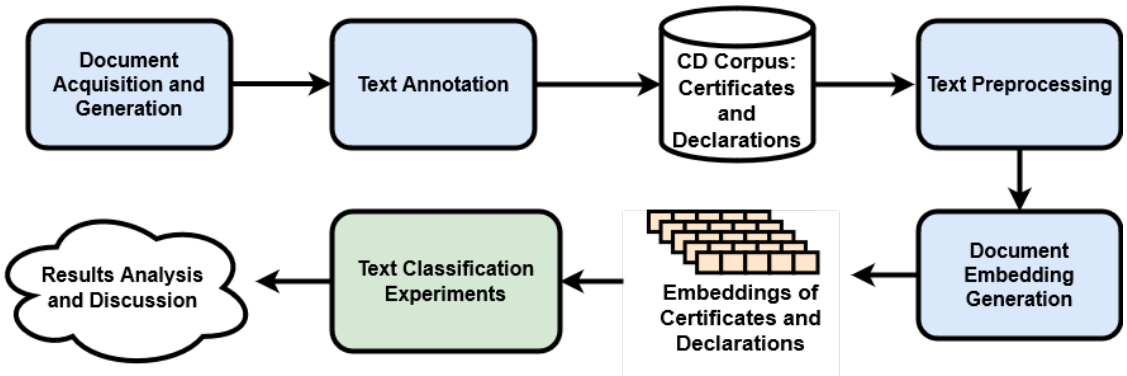

Figure 1. The proposed NLP-based pipeline and its constituting steps.

This research considers the accreditation of extracurricular activities as a text classification task, illustrated in Figure 1. It involves corpus construction and annotation, text preprocessing, and document embedding generation, thus allowing the use of traditional

and state-of-the-art classification models. Their performances were later compared to validate the proposed methodology.

3.1. Document Analysis

The Computer Science curriculum at Universidade de Brasília considers the following extracurricular activities, which the student is responsible for choosing to be accredited in their transcripts during his enrollment to the undergraduate program:

- **Group 1**: publication of scientific papers in conferences, workshops, journals, etc;
- **Group 2**: participation in an Undergraduate Research Program (*Programa de Iniciação Científica*);
- **Group 3**: participation in a Tutorial Education Program (*Programa Educação Tutorial (PET)*) [Fleith et al. 2012] [Martins 2007];
- **Group 4**: participation as a lecturer, organizer, speaker, or exhibitor in Computer Science-related events or short-term courses;
- **Group 5**: participation as a member of an undergraduate tutoring project;
- **Group 6**: participation in conferences, workshops, symposiums, short-term courses, and meetings;
- **Group 7**: participation as a team member of an extension project;
- **Group 8**: participation as a team member of a junior enterprise in computing (*Empresa Júnior de Computação*).

For each submitted document of those groups, the student must present the associated proof documents to be validated by the undergraduate program coordinator or supporting secretariats, here called specialists, as shown in Table 1. In this sense, the specialist verifies the content of each proof document and the dedicated hours to the corresponding activity. The most time-consuming procedure is to analyze the document's content, being the focus of this research.

Table 1. Types of proof documents for each group.

Group	Proof document
1	Copy of the paper
2	Certificate issued by the Undergraduate Research Program or declaration signed by the supervisor
3	Declaration issued by the chancellors or by the professor
4	Certificate issued by the organization
5	Declaration issued by the chancellors or by the responsible Professor
6	Certificate issued by the organization
7	Declaration issued by the chancellors or by the responsible Professor
8	Declaration issued by the junior enterprise Leader or by the responsible Professor

3.2. Document Acquisition and Generation

The supervised NLP models considered for this problem demand labeled corpora. To the best of our knowledge, there are no public corpora available related to certificates and declarations in Portuguese or English languages. This motivated us to build an initial corpus

of 603 academic documents in the English and Portuguese languages for the underlying task.

Our main approach was to randomly collect certificates, declarations, and documents related to Computer Science (CS) from the Internet. The document acquisition was conducted on image search engines to manually collect visual examples related to the study, focusing on online courses because CS students regularly accomplish them for training and skills improvement purposes. Another approach adopted was to manually search the main websites for conferences, symposiums, scientific, and technology events correlated with the area of Computer Science at national and international level[1].

We also generated some artificial certificates and declarations using templates from the Universidade de Brasília. These institutional academic documents are typically issued to verify students' participation in internal events, extension projects, and tutorial programs. The template of an internal academic certificate follows a defined structure that contains the university logo, event's name, date, participant's name, venue, and signatures of the responsible. Since these institutional certificates cannot be publicly collected on the internet, fake names, events, and dates were filled into the templates as a pseudonymization approach to preserve the data distribution.

Although we collected and generated several academic documents on the internet, the frequency distribution of documents per category must be respected as they appear in a real scenario of accreditation of extracurricular activities. For that purpose, we considered a sample of 104 real assessed proof documents by the specialists and we determined the frequencies per category, which are shown in Table 2. Documents not belonging to any predefined category were labeled as "-1", meaning that the given proof documents were not considered valid by the specialists.

Table 2. Number of instances per category (class label) in specialist's sample.

Group	Number of documents	Approximate proportions
-1	0	0%
1	2	2%
2	4	4%
3	1	1%
4	11	10%
5	0	0%
6	73	70%
7	6	6%
8	7	7%
Total	104	100%

The final step was to extract raw text from the documents, using PyMuPDF[2] for PDF file format and Tesseract OCR[3] for image files, as illustrated in Figure 2.

[1] https://ppgcc.github.io/discentesPPGCC/pt-BR/qualis/
[2] https://pypi.org/project/PyMuPDF/
[3] https://github.com/tesseract-ocr/tesseract

Nome do Participante
Participou do XII Congresso Brasileiro de Informática na Educação - CBIE, que ocorreu durante os dias 6 a 10 de novembro de 2023, na cidade de Passo Fundo, Rio Grande do Sul.
C.H. 40 horas
1303.
Part_CBIE23

(a) (b)

Figure 2. Text extraction from certificate: (a) original certificate in PDF file — sensible information were intentionally removed; (b) raw text extracted using PyMuPDF.

3.3. Text Annotation

The annotation process for the certificates and declarations from the collected documents was performed by a specialist with the support of LabelStudio[4] platform. The specialist was required to label each document according to the guidelines regarding Table 1. Documents not belonging to any of the pre-defined labels were labeled as "-1". This new category is important since students may mistakenly submit an invalid certificate or declaration that does not satisfy the rules. For instance, a "Best Paper" certificate is not accredited for extracurricular activity since it must explicitly show the student's name as a speaker or presenter, as well as the hours of workload. Table 3 shows the frequencies by instance in the built corpus.

Table 3. Number of instances per class label in the built corpus.

Group	Number of documents	Approximate proportions
-1	66	11%
1	92	15%
2	73	12%
3	14	2%
4	41	7%
5	35	6%
6	198	33%
7	34	6%
8	50	8%
Total	603	100%

We can note that built corpus contains more samples for the minor class labels in relation to the distribution of the real samples to minimize the class imbalance during the training of the classification models.

[4] https://labelstud.io/

3.4. Text Preprocessing

The goal of text preprocessing is to prepare the input text by retaining the relevant words for further classification and information retrieval tasks. This step benefits these tasks by reducing noise, improving the efficiency of algorithms, enhancing the accuracy of the model, and ensuring that the text is in a standardized format [Jurafsky and Martin 2019].

The NLTK[5] (Natural Language Tool Kit) library [Bird 2006] was used for preprocessing. Punctuation symbols were removed and words were adjusted to lowercase. Next, we eliminated stopwords, which are commonly used terms in text that do not provide relevant meaning or information to NLP models. Lemmatization was performed to reduce words to their base forms and to group inflected forms. The removal of stopwords and the application of lemmatization were performed according to the language of the input raw text.

3.5. Document Embedding Generation

TF-IDF (Term Frequency–Inverse Document Frequency) is a NLP technique used to evaluate the importance of a word in a document relative to a collection of documents [Arroyo-Fernández et al. 2019]. The main idea of TF-IDF is to highlight words that are frequent in a specific document but relatively rare in the entire set of documents. This is achieved through two main parts:

- **TF (Term Frequency)**: measures the frequency of a word in a specific document. The more times a word appears in a document, the higher its TF score.

$$TF(t) = \frac{\text{Number of times term } t \text{ appears in a document}}{\text{Total number of terms in the document}} \tag{1}$$

- **IDF (Inverse Document Frequency)**: measures the rarity of a term in relation to all documents in the corpus. Terms that appear in fewer documents have a higher IDF.

$$IDF(t) = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents with term } t \text{ in it}}\right) \tag{2}$$

Word2vec [Mikolov et al. 2013] is a structured representation of words as dense vectors, known as word embeddings. Word2vec offers an implementation of the Continuous Bag-of-Words and Continuous Skip-Gram architectures. In this technique, a shallow neural network model is used to obtain semantic information of words in a corpus of documents. The Continuous Bag-of-Words Model estimates the current word based on the context, while the Continuous Skip-Gram Model uses the current word to predict the context window of surrounding words.

3.6. Classification Models

The classification task consists of using traditional models and state-of-the-art NLP language models.

Support Vector Machine (SVM) is a supervised ML algorithm commonly used in classification, regression, and outlier detection tasks. SVM seeks to find the hyperplane

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

that best separates data from different classes in the feature space. In this research, we use the SVM classifier with the input text represented as TF-IDF vectors due to its successful employment in literature [Dadgar et al. 2016] [Luthfi and Lhaksamana 2020].

Convolutional Neural Networks (CNNs) are similar to Artificial Neural Networks (ANNs), although CNNs are primarily utilized for recognizing patterns in images [O'Shea and Nash 2015]. CNNs, with their architecture specifically designed for feature extraction, can be combined with pre-trained word vector representations for various classification tasks [Kim 2014].

Long Short-Term Memory (LSTM) is a recurrent neural network (RNN) designed to process sequential data, allowing it to learn and make predictions based on past samples [Siami-Namini et al. 2019]. Its architecture consists of several gates, which are responsible for retaining past information and constructing the prospective model. We also considered the Bidirectional LSTM (BiLSTM) architecture, which contains two LSTM cells, in which the input data is processed from left to right in the first LSTM while the second one processes the input data from right to left.

Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al. 2018] is a language representation model with two main steps: pre-training, which includes Masked Language Modeling (MLM) and Next Sentence Prediction (NSP), and fine-tuning. The Masked Language Modeling involves masking some words in the sentence and training the model to predict the masked words using the context of the remaining words. Next Sentence Prediction (NSP) involves training the model to understand the relationship between two sentences by receiving pairs of sentences and deciding whether the second sentence corresponds to the context of the first one presented. Fine-tuning consists of adjusting the parameters with a small sample of labeled data for each task the model will perform.

The full setting of the classification models are described in the next section, since they require the optimization of hyperparameters.

4. Experimental Results

In this section, we present the experimental results with discussions to validate the proposed methodology by considering the designed approaches for addressing the task. The source code and the CD corpus are available at our repository [6].

The evaluation strategy is the Stratified K-Fold Cross Validation with 5 folds, in which for each fold, the data is split in training (80%) and validation (20%). In each round, the classification model is trained using the training set, and its hyperparameters are optimized using the validation set. After obtaining the best classification model, we compute the Macro *F1-score* on the testing set due to the significant imbalance within the corpus.

We first provide details concerning the optimization of hyperparameters for each classification model. The training of the classification models were performed using the Adam algorithm [Kingma and Ba 2014]. The tuning of hyperparameters in the language

[6] https://gitlab.com/gvic-unb/sbie-2024-classification-of-certificates-and-declarations

models was performed using KerasTuner[7].

4.1. Hyperparameter Optimization

Five classification models were considered, each using the most suitable text representation. We describe the range of values tested for each hyperparameter in the optimization processes.

The first classifier considers TF-IDF and SVM with a Radial Basis Function (RBF) kernel. The document vectorization was restricted to a maximum of 5000 features, and grid search[8] was employed to optimize the SVM hyperparameters for training text representation:

- **TF-IDF+SVM** : Regularization parameter **C**: [10^{-1}, 1 , 10^{2}, 10^{3}] and Kernel coefficient γ: [1 , 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}].

Three classifiers were derived from the aforementioned language models based on recurrent neural networks and transformers. The architectures consider pre-trained Word2vec embeddings with 300 dimensions in the embedding layer. These embeddings were obtained from the NILC repository [9] [Hartmann et al. 2017] and were trained in a large corpora of Portuguese language. These models were trained using a batch size of 32 with a maximum sentence length of 256.

- **Word2vec-CNN**: The model was trained using 400 epochs. The 1-D convolutional layer considered the number of filters units varying in the range $[16, 32, 64, 128, 256, 512]$, followed by an 1-D Global Average Pooling layer. The kernel size varied in the range $[1, 7]$ with 2-step size and the range of the learning rates was [10^{-3}, 10^{-4}, 2×10^{-4}, 5×10^{-4}, 5×10^{-5}]. The hyperparameter tuner was set to 5 maximum trials for 100 epochs. Early stopping was applied to halt training when the validation loss did not decrease for 5 epochs.
- **Word2vec-LSTM**: The model was trained using 400 epochs. The number of units in the LSTM layer varied within the range $[128, 256, 512, 1024]$. The flatten layer followed by a dropout layer with $[0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55]$ dropout values. The same range of dropout values is applied to the recurrent dropout in the LSTM layer. The range of the learning rates was [1×10^{-3}, 1×10^{-4}, 2×10^{-4}, 5×10^{-4}, 5×10^{-5}, 2×10^{-6}, 5×10^{-6}]. The hyperparameter tuner was set to 15 maximum trials for 150 epochs. Early stopping was applied to halt training when the validation loss did not decrease for 5 epochs.
- **Word2vec-BiLSTM**: The model was trained for 400 epochs. The number of units in the BiLSTM layer varied within the range $[128, 256, 512, 1024]$. A flatten layer followed by a dropout layer was tested with dropout values in the range $[0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55]$. The same range of dropout is applied to the recurrent dropout in the BiLSTM layer. The range of tested learning rates was [10^{-3}, 10^{-4}, 2×10^{-4}, 5×10^{-4}, 5×10^{-5}, 2×10^{-6}, 5×10^{-6}]. The hyperparameter tuner was set to 15 maximum trials for 100 epochs. Early stopping was applied to halt training when the validation loss did not decrease for 5 epochs.

[7] https://keras.io/keras_tuner/

[8] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

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

The last model was a pre-trained BERT using the multilingual base model [10]. The BERT-Multilingual model consists of 12 Transformer encoder layers, with each layer containing 768 hidden units and 12 attention heads. The model includes a dropout rate of 0.1, internally applied within the BERT architecture. For this experiment, 10 epochs for training were set using a batch size of 8. A maximum sentence length of 256 and early stopping was applied to halt training when the validation loss did not decrease for 2 epochs:

- **BERT-Multilingual**: Learning rates were [10^{-5}, 2×10^{-5}, 5×10^{-5}]. The hyperparameter tuner was set to 2 maximum trials.

4.2. Results and Discussion

Table 4. Classification results using Stratified K-Fold Cross Validation ($\mathbf{K = 5}$): average and standard deviation of Macro F1-Score per class label for all folds.

Class	Text Classification Models					Support
	TF-IDF and SVM	BERT (Mult)	Word2Vec (CNN)	Word2Vec (BiLSTM)	Word2Vec (LSTM)	
-1	**0.77±0.08**	0.68±0.13	0.64±0.11	0.58±0.10	0.57±0.07	13
1	0.98±0.02	**0.99±0.02**	0.98±0.01	0.97±0.04	0.97±0.03	18
2	**0.98±0.03**	**0.98±0.03**	0.96±0.03	0.97±0.03	0.97±0.03	15
3	**0.95±0.10**	0.91±0.11	0.87±0.13	0.71±0.19	0.56±0.34	3
4	**0.83±0.11**	0.71±0.07	0.66±0.12	0.64±0.14	0.54±0.31	8
5	**1.00±0.00**	0.99±0.03	**1.00±0.00**	**1.00±0.00**	0.99±0.03	7
6	0.92±0.02	**0.93±0.03**	0.91±0.02	0.88±0.02	0.88±0.02	39
7	**0.98±0.03**	0.82±0.31	0.84±0.07	0.95±0.07	0.86±0.17	7
8	**1.00±0.00**	0.95±0.08	0.94±0.04	0.99±0.02	0.96±0.06	10
GA	**0.94±0.08**	0.88±0.11	0.87±0.13	0.86±0.16	0.81±0.18	120

To check for the presence of a significant performance difference among the models' *F1-scores* outcomes in Table 4, the non-parametric Friedman test was conducted [Pohlert 2014], since normality was not assumed and considering a significance level of $\alpha = 5\%$. Additionally, the folds were generated using the same random seed for all classifiers, resulting in multiple comparisons under the same conditions. The following null and alternative hypotheses were formulated:

H_0 : No differences between all the models average macro *F1-scores* per class.
H_a : At least one model average macro *F1-scores* per class differs from the others.

Due to null hypothesis rejection for a resulting *P-Value* of 0.002092, a Nemenyi pairwise comparison [Pohlert 2014], considering a significance level of α = 5% was performed, as shown in Table 5.

[10] https://huggingface.co/google-bert/bert-base-multilingual-uncased

Table 5. Nemenyi post-hoc test results.

Pairwise Model Comparison	P-Value	Test Outcome
TF-IDF + SVM vs Word2vec (CNN)	0.0687	Not Significant
TF-IDF + SVM vs Word2vec (LSTM)	0.0018	Significant
TF-IDF + SVM vs Word2vec (BiLSTM)	0.0687	Not Significant
Word2vec (CNN) vs Word2vec (LSTM)	0.7971	Not Significant
Word2vec (CNN) vs Word2vec (BiLSTM)	1.0000	Not Significant
Word2vec (LSTM) vs Word2vec (BiLSTM)	0.7971	Not Significant
BERT vs TF-IDF + SVM	0.5685	Not Significant
BERT vs Word2vec (CNN)	0.7971	Not Significant
BERT vs Word2vec (LSTM)	0.1664	Not Significant
BERT vs Word2vec (BiLSTM)	0.7971	Not Significant

The results showed in Table 4 indicate that despite a limited number of documents from specific classes ("-1", "3" and "4") the classification models achieved satisfactory results given that the corpus is not large, which can be explained by the difficulty of obtaining certificates, declarations, and statements on the Internet. Notably, the TF-IDF+SVM model achieved the highest average macro *F1-score* of 0.94, which is a positive aspect due to its low computational processing time compared to the other models.

The BERT-Multilingual, Word2Vec-CNN, and Word2Vec-BiLSTM models also obtained a satisfactory performance, though slightly lower than TF-IDF and SVM, with average *F1-scores* of 0.88, 0.87, and 0.86, respectively. Finally, the Word2Vec-LSTM model, with an average macro *F1-score* of 0.81, was the most impacted by the class imbalance in the dataset. This difference in performance is emphasized by their respective *F1-scores* within less represented classes, such as labels 3 and 4. This suggests that processing the text in both directions provided the model to capture more patterns for the predictive task. Word2Vec-LSTM obtained the highest global average deviation, with a standard deviation of 0.18 indicating that the average *F1-score* was probably affected by outliers within the stratified 5-fold cross-validation process.

Training BERT-Multilingual is computationally expensive, and its training in this study was tailored accordingly. Further training and hyperparameter tuning could potentially enhance its performance. The Word2Vec-CNN yielded satisfactory results, showing that this neural network architecture can effectively extract features in texts by capturing relations between neighbor words.

All models presented difficulties when categorizing documents related to class label "-1". This factor may be associated with the high similarity of the text with factors present in other documents as well as there are fewer documents of this class in relation to the other class labels. Increasing the size of the corpus and, consequently, the training set, may lead to improvements in the model, especially for less-represented class labels.

Nemenyi post-hoc makes pairwise comparisons, considering the null hypothesis of equality between the macro averages *F1-scores*. The test results in Table 5 show significant differences between Word2Vec-LSTM compared to the best *F1* scoring model, TF-IDF+SVM. This implies that the performance gap between these models is not due to random chance, but is statistically significant. In other words, the TF-IDF+SVM model

consistently outperforms the Word2Vec-LSTM model in terms of *F1-scores* across the folds tested in the described experiments.

5. Conclusion

This research introduced an innovative contribution by proposing an NLP-based pipeline to address the accreditation of extracurricular activities to the student's transcripts. This process demands students to submit proof documents which are assessed by specialists, who analyze and manually label each one according to the regulation. This task was handled as a classification problem, which required the construction of a corpus of academic documents (certificates, declarations, scientific papers, etc) to allow the training of classification models.

The considered classification models were SVM, language models based on recurrent neural networks (LSTM and BiLSTM), and transformers (BERT). The experimental results demonstrated that SVM alongside the TF-IDF representation of texts is reliable and powerful by achieving a high accuracy while pursuing a low computational cost compared to the other classification models. The proposed solutions can benefit coordinators and educational institutions in the accreditation of extracurricular activities, improving their productivity and speeding up this process for the academic community.

Future works will be guided to explore additional tasks to validate the built corpus, such as career progression for faculty and administrative staff. Furthermore, implementing the proposed solution in an academic information system would make the accreditation of extracurricular activities more efficient for students, secretariats, and institutional personnel. Expanding the dataset is a key task since it can achieve even more robust results. Other approaches for increasing the corpus may include semi-supervised learning strategies proposed in the literature [Xie et al. 2020] [Zhang et al. 2021] [Duarte and Berton 2023].

6. Acknowledgments

The authors of this paper would like to thanks to FAPDF (Fundação de Apoio à Pesquisa do Distrito Federal) for the support provided throughout this work - Finance Code 10422B.

References

Arroyo-Fernández, I., Méndez-Cruz, C.-F., Sierra, G., Torres-Moreno, J.-M., and Sidorov, G. (2019). Unsupervised sentence representations as word information series: Revisiting tf–idf. *Computer Speech & Language*, 56:107–129.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Dadgar, S. M. H., Araghi, M. S., and Farahani, M. M. (2016). A novel text mining approach based on tf-idf and support vector machine for news classification. In *2016 IEEE International Conference on Engineering and Technology (ICETECH)*, pages 112–116. IEEE.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dogra, V., Verma, S., Kavita, Chatterjee, P., Shafi, J., Choi, J., and Ijaz, M. F. (2022). A complete process of text classification system using state-of-the-art nlp models. *Computational Intelligence and Neuroscience*, 2022(1):1883698.

Dragoni, M., Villata, S., Rizzi, W., and Governatori, G. (2018). Combining natural language processing approaches for rule extraction from legal documents. In *AI Approaches to the Complexity of Legal Systems: AICOL International Workshops 2015-2017: AICOL-VI@ JURIX 2015, AICOL-VII@ EKAW 2016, AICOL-VIII@ JURIX 2016, AICOL-IX@ ICAIL 2017, and AICOL-X@ JURIX 2017, Revised Selected Papers 6*, pages 287–300. Springer.

Duarte, J. M. and Berton, L. (2023). A review of semi-supervised learning for text classification. *Artificial intelligence review*, 56(9):9401–9469.

Fleith, D. D., Costa Jr, A. L., and Soriano De Alencar, E. M. (2012). The tutorial education program: An honors program for brazilian undergraduate students.

Hartmann, N., Fonseca, E. R., Shulby, C., Treviso, M. V., Rodrigues, J. S., and Aluísio, S. M. (2017). Portuguese word embeddings: Evaluating on word analogies and natural language tasks. *CoRR*, abs/1708.06025.

Hassan, F. u. and Le, T. (2020). Automated requirements identification from construction contract documents using natural language processing. *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction*, 12(2):04520009.

Heppner, A., Pawar, A., Kivi, D., and Mago, V. (2019). Automating articulation: Applying natural language processing to post-secondary credit transfer. In *IEEE Access*, volume 7, pages 48295–48306.

Jurafsky, D. and Martin, J. H. (2019). *Speech and Language Processing*. Pearson.

Khurana, D., Koli, A., Khatter, K., and Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3):3713–3744.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Krishnamurthy, J., Dasigi, P., and Gardner, M. (2017). Neural semantic parsing with type constraints for semi-structured tables. In Palmer, M., Hwa, R., and Riedel, S., editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1516–1526, Copenhagen, Denmark. Association for Computational Linguistics.

Lawhorn, B. (2008). Extracurricular activities. *Occupational Outlook Quarterly*, 9(1):16–21.

Lu, W., Zhou, Y., Yu, J., and Jia, C. (2019). Concept extraction and prerequisite relation learning from educational data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9678–9685.

Luthfi, M. F. and Lhaksamana, K. M. (2020). Implementation of tf-idf method and support vector machine algorithm for job applicants text classification. *Jurnal Media Informatika Budidarma*, 4(4):1181–1186.

Martins, I. L. (2007). Educação tutorial no ensino presencial: uma análise sobre o pet. *PET–Programa de Educação Tutorial: estratégia para o desenvolvimento da graduação. Brasília: Ministério da Educação*.

Meystre, S. and Haug, P. J. (2006). Natural language processing to extract medical problems from electronic clinical documents: performance evaluation. *Journal of Biomedical Informatics*, 39(6):589–599.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations, ICLR 2013*.

O'Shea, K. and Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

Pohlert, T. (2014). The pairwise multiple comparison of mean ranks package (pmcmr). *R package*, 27(2019):9.

Rahman, A.-u., Musleh, D., Nabil, M., Alubaidan, H., Gollapalli, M., Krishnasamy, G., Almoqbil, D., Khan, M. A. A., Farooqui, M., Ahmed, M. I. B., et al. (2022). Assessment of information extraction techniques, models and systems. *Mathematical Modelling of Engineering Problems*, 9(3).

Siami-Namini, S., Tavakoli, N., and Namin, A. S. (2019). The performance of lstm and bilstm in forecasting time series. In *2019 IEEE International conference on big data (Big Data)*, pages 3285–3292. IEEE.

Widiastuti, N. and Dewi, K. (2020). Document image extraction system design. volume 879, page 012069.

Xiao, W., Ji, P., and Hu, J. (2022). A survey on educational data mining methods used for predicting students' performance. *Engineering Reports*, 4(5):e12482.

Xie, Q., Dai, Z., Hovy, E., Luong, T., and Le, Q. (2020). Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268.

Zhang, B., Wang, Y., Hou, W., Wu, H., Wang, J., Okumura, M., and Shinozaki, T. (2021). Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419.