Journal of Scientometric Research, 2025; 14(1):373-382.
https://www.jscires.org

Research Article

Quartile Prediction and Journal Recommendation Using Deep Learning Models for Artificial Intelligence Articles

Fernando Aguilar-Canto, Cesar Macias*, Alberto Espinosa Juárez, Marco Antonio Cardoso-Moreno, Hiram Calvo

Laboratorio de Ciencias Cognitivas Computacionales, Centro de Investigación en Computación, Instituto Politécnico Nacional, Mexico City, MEXICO.

ABSTRACT

Journal recommendation systems serve as valuable tools for researchers, addressing the complex task of multi-class text classification. With the advent of Transformer architectures, there is newfound potential to enhance existing recommendation systems, particularly in the realm of academic journals. While current technologies are capable of classifying journals based on article content, we still lack an algorithm that can predict the quartile ranking of journals. Such a development would be immensely beneficial for researchers to assess their articles before submission. In our study, we tackle both tasks simultaneously. We trained various state-of-the-art Transformer architectures and machine learning algorithms, ranging from BERT to GPT-2. Surprisingly, we achieved better quantitative results with smaller models, especially DistilBERT, as well as classical classifiers. However, when it came to quartile prediction, success was limited to testing within the same journals. Generalization across different journals proved elusive. This observation strongly suggests that quartile prediction currently relies indirectly on journal classification, highlighting the limitations of existing technology, the collected dataset, or the impossibility of solving the task.

Keywords: Quartile Prediction, Journal Recommendation System, Transformers.

Correspondence:
Cesar Macias
Laboratorio de Ciencias Cognitivas Computacionales, Centro de Investigación en Computación, Instituto Politécnico Nacional-07720, Mexico City, MEXICO.
Email: cmaciass2021@cic.ipn.mx.
ORCID: 0009-0005-1708-5359

Received: 25-07-2024;
Revised: 14-09-2024;
Accepted: 12-11-2024.

INTRODUCTION

The development of Deep Learning models has profoundly impacted different subareas of Natural Language Processing (NLP), particularly since the introduction of the Transformer architecture.[1-3] In addition, Recommendation Systems have also been significantly influenced by recent advances in NLP.[4,5]

In particular, the dual task of Journal Recommendation and Quartile Prediction for journals, as part of Natural Language Processing, might also be solved by leveraging Transformers. A Journal Recommendation system can guide users in correctly inferring where to submit a completed research paper, while Quartile Prediction assesses the quality of the paper. Quartiles serve as bibliographic metrics of journal quality, based on citations. Although this hypothesis is debatable, it assumes that better papers receive more citations than others. Even if this hypothesis is not universally true, predicting the quartile of a research paper indirectly predicts its future citations, which a is key feature in research.

DOI: 10.5530/jscires.20251460

In this article, we aim to predict both the quartile and journal of a given research paper in the field of Artificial Intelligence. We used three crucial features: the title, the abstract and the keywords. While the second subtask (Journal Recommendation) has been extensively explored with various models (see Section 2), the Quartile Prediction task remains understudied, at least to our knowledge. The relationship between the two tasks lies in the fact that a good journal predictor is also a competent quartile predictor, or at least performs comparably to the quartile classifier. However, can this generalization hold for unseen data?

This paper is structured as follows. Section 2 is devoted to the literature research. Section 3 presents the Methodology. Results are shown in Section 4, while the Discussion appears in Section 5.

Related work

To our knowledge, while the development of journal recommendation systems is an active and rich research area, the formulation of the more specific topic of quartile prediction systems remains unexplored. Nevertheless, our literature research indicates that in the context of journal recommendation, most work has been conducted using classical Natural Language Processing (NLP) techniques, with Deep Learning methods being scarcely implemented.

Beel *et al.*[6] conducted a review until 2016, revealing that over 200 articles had been published on journal recommendation. Most of these studies relied on traditional Machine Learning techniques, such as TF·IDF. However, the emergence of the Transformer architecture shifted the landscape. The proposal of the Transformer architecture in 2017 by Vaswani *et al.*[1] rapidly became the state-of-the-art for several NLP tasks. Consequently, journal recommendation systems should consider incorporating Deep Learning techniques, especially Transformers.

Feng *et al.*[7] implemented a Convolutional Neural Network (ConvNet) to classify 1130 biomedical journals, achieving superior results compared to most commercial journal finders. Notably, they also explored the incorporation of Long Short-Term Memories (LSTMs), albeit with slightly slower performance. It is worth noting that the utilization of journal identifiers, such as the LocatorPlusID, may not be advisable due to potential issues related to data leakage.

Nguyen *et al.*[8] also compared the performance of an LSTM architecture and ConvNets, incorporating text embeddings. They proposed an ensemble approach, combining ConvNet with the Scientific Submission Recommendation System for Computer Science (S2RSCS) method, achieving top results in their data. Their technique, named S2CFT, utilized inputs from the title, abstract and keywords. The S2RSCS comprises Logistic Regression and a shallow neural network.[9] The optimal results were obtained by leveraging a combination of abstracts and keywords, underscoring the relevance of title information. Subsequent refinements have been detailed in their follow-up publication.[10]

Recent developments have considered the usage of Transformers. For instance, Michail *et al.*[11] achieved superior results using the DistilBERT architecture compared to both the BERT architecture and Doc2vec. Liu *et al.*[12] also implemented BERT as a classifier and the combination of BERT with an AutoEncoder architecture yielded better results than classical models. Additionally, Hassan *et al.*[13] compared BERT, SciBERT and the USE transformers, with the last architecture showing the best results. Hybrid approaches have also been explored. For example, Zhao *et al.*[14] implemented a BERT architecture and a Bidirectional Gated Recurrent Unit (BiGRU), achieving better results than classical methods.

In spite of the comparison with the classical approaches, Barolli *et al.*,[15] which focused its research on COVID-19-related papers, defends the use of techniques such as TF·IDF and graph-based approaches since their representation are similar to the Transformers (BERT and MiniLM).

METHODOLOGY

Deep Learning Models

Considering the literature review, we fine-tuned the following Transformer architectures:

- BERT-base.[2]
- DistilBERT-base.[16]
- SciBERT-base.[17]
- RoBERTa-base.[18]
- XLM-RoBERTa-base.[19]
- GPT-2 medium.[3]

We used Adam optimizer with a learning rate of 3×10^{-5} and Sparse Categorical Cross Entropy as a loss function. In this case, we selected a low learning rate because we were implementing fine-tuning. The rest of the hyperparameters are default in Hugging Face. The training process considered two initial epochs for training and the training continues until decay on the F1-score macro in validation. The rest of the hyperparameters were not changed in the context because we prioritized evaluating different models rather than different configurations of the same models.

Machine Learning Models

For the purpose of comparison, we implemented some classical machine learning experiments. In these experiments, we employed Support Vector Machines (SVM) with a linear kernel. The maximum number of iterations was set to 1×10^{9}, with a regularization parameter of 0.9. The data underwent preprocessing, including the removal of non-ASCII characters, alphanumeric words and HTML entities and the conversion of all letters to lowercase.

Text encoding was performed using the TF·IDF technique, employing N-gram ranges from 1 to 5 for both title and abstract SVMs. For the SVM trained with keywords, N-gram ranges from 1 to 3 were utilized. The maximum document frequency was set at 0.7, 0.8 and 1.0 for SVMs trained with titles, abstracts and keywords, respectively. The parameters for the TF·IDF vectorizer and the SVM model were selected after several experiments with different parameters were computed, with this parameter the SVM model had the best performance on the validation set.

Model Ensemble

For the model ensemble, we opted for a majority voting system. In the event that all three models (title, keywords, abstract) voted for the same class, the global class was assigned. If two of these models voted for the same class, the majority class was assigned. If each model voted for a different class, then preference was given to the model that had performed best during validation.

Data acquisition

The acquisition processing was performed by using web scrapping using Elsevier's Abstract Retrieval API. Additional refinements of the dataset were performed manually. We considered all journals with quartile in the Journal Citation Reports 2022 list with the tag COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE-SCIE.

However, as a result of the web scrapping process, only a few papers are correctly labeled with the abstract and keywords. This process led to three different configurations of the dataset, depending on the specific subtask.

Tasks

Subtask 1A: Quartile prediction with titles

In the first experimental setup, we aimed to predict the quartile by using the title. As a result, we got 54749 papers for the training set, 6844 for the validation set and 6844 for the testing set. In total, 17526 were devoted to the Q1 class, 16721 for the Q2 class, 16921 for Q3 and 17269 for Q4, showing a low imbalance. This dataset was studied with most of the models. In this configuration, the split process was completely random. Each journal has around 466 papers. We considered 147 journals in total. Subtask 1B:

Quartile prediction

We considered a reduced dataset with the full information of the articles, which are the title, abstract and keywords, using the same scheme of.[8] In total, the reduced dataset used seven journals. To test the generalization capability of the quartile prediction scheme, we used 5 were devoted to training, one for validation and one for testing. The classes present balance as shown in Figure 1.

Subtask 2: Journal recommendation

In this case, the reduced dataset was split differently, using a random scheme. In total, we got 9540 articles in the training set, 1193 in the validation set and 1193 for testing. In this case, the task aimed to predict the journal, given the information of the article, creating a recommender system. It is also worth mentioning that the dataset is slightly unbalanced, as shown in Figure 2.

RESULTS

Subtask 1A: Quartile prediction with titles

Subtask 1A was employed as a preliminary exploration for subsequent experiments. In this context, all considered deep learning models were utilized. Each model underwent training for two Epochs (Ep). However, for F1-macro (F1-m) values surpassing 0.49, a retraining process was implemented until the validation F1-macro demonstrated a decrease. The corresponding results are presented in Table 1.

In the table, we observe that cased models were generally slightly better, or at least as good as their uncased counterparts. Contrary to intuition, BERT-cased, DistilBERT-cased and SciBERT-scivocab-cased achieved superior comparative results, even outperforming GPT-2, XLM-RoBERTa and RoBERTa. We conducted evaluations on the testing set with the best model (DistilBERT-base-cased), resulting in an F1-score macro of 0.48397, accuracy (Acc) of 0.48758 and a weighted F1-score (F1-w) of 0.48422. For the prediction of two classes (Q1-Q2 versus Q3-Q4), we obtained an F1-macro of 0.69022, an accuracy of 0.74883 and a weighted F1-score of 0.75582. The confusion matrix for the testing set is provided in Figure 3.

Subtask 1B: Quartile prediction with full information

In the subsequent experiments, we exclusively considered the best models from Sub-task 1A (BERT, DistilBERT, SciBERT) while introducing the use of Machine Learning algorithms. For the Quartile prediction task with full information, it's important to note that we kept journals separate in the training, validation and testing sets, focusing on studying the generalization capabilities of the predictor. Unfortunately, due to VRAM limitations on the A100, we were only able to evaluate abstracts with the smaller models (DistilBERTs and Machine Learning algorithms). In this scenario,

we measured the F1-score$_2$ performance, addressing the binary classification problem by merging classes Q1 with Q2 and Q3 with Q4. Table 2 summarizes the results of the models using the title as input. It is noteworthy that DistilBERT-base-uncased achieved the best performance in F1-macro, consistently delivering good results in the binary classification problem. However, all the studied models performed worse than random in the multi-class problem, a challenge persisting throughout the entire evaluation.

In Table 3, when using keywords as inputs, DistilBERT-base-cased demonstrated better results in the F1-macro for the multi-class scenario, while DistilBERT-base-uncased exhibited superior results in the F1-macro for the binary classification. In Table 4, the reverse situation arises when utilizing abstracts as inputs: DistilBERT-base- cased presents better results in the binary problem, whereas DistilBERT-base-uncased showed superior

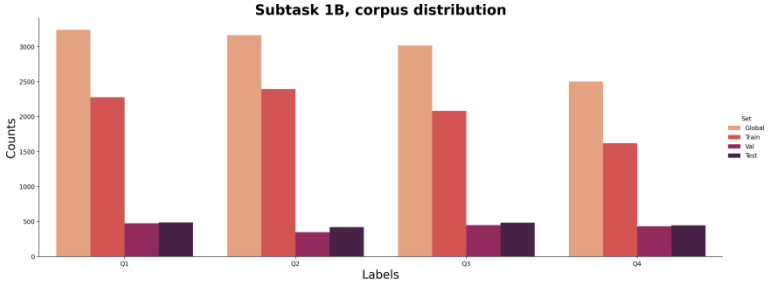

Figure 1: Distribution of the reduced dataset with the quartils as labels.

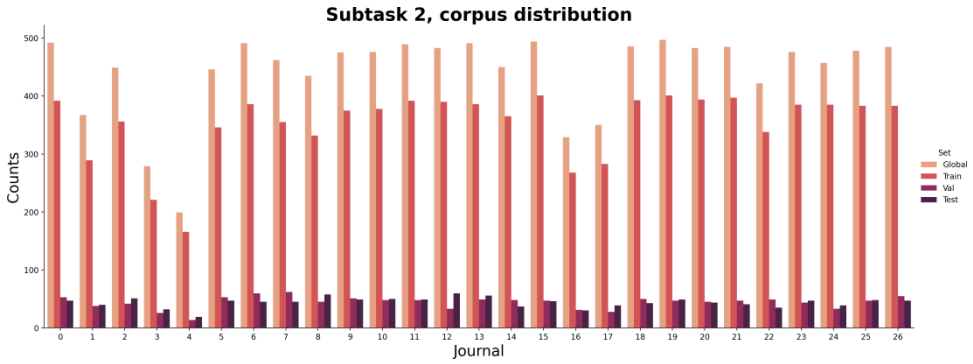

Figure 2: Distribution of the reduced dataset with the journals as labels.

Table 1: Summary of the results in the Subtask 1A in validation.

Model	Ep	F1-m	F1-w	Acc
BERT-uncased	2	0.47587	0.47648	0.47604
BERT-cased	2	0.49828	0.49912	0.50395
BERT-cased	3	0.49749	0.49846	0.50321
DistilBERT-uncased	2	0.46672	0.46733	0.47209
DistilBERT-cased	2	0.49472	0.49527	0.49416
DistilBERT-cased	3	0.49497	0.49575	0.49825
DistilBERT-cased	4	0.4992	0.5002	0.50161
DistilBERT-cased	5	0.48942	0.49036	0.49416
SciBERT-scivocab-uncased	2	0.47587	0.47585	0.48057
SciBERT-scivocab-cased	2	0.49332	0.49242	0.49211
SciBERT-scivocab-cased	3	0.49251	0.49312	0.4943
RoBERTa	2	0.46652	0.46784	0.48159
RoBERTa	3	0.49269	0.49332	0.49459
RoBERTa	4	0.49035	0.49116	0.49138
XLM-RoBERTa	2	0.10065	0.25205	0.25205
DeBERTa	2	0.47268	0.47382	0.47867
GPT-2 medium	2	0.42935	0.42958	0.44302

results in the multi-class problem. In general, we observed a trade-off between an increase in binary metrics and a decrease in multi-class metrics.

Ensemble

Results of the ensemble are presented in Table 5. In this case, it is evident that only the SVM method achieved better results than random. However, it exhibits slightly lower performance in the binary problem. All ensembles performed worse than the best base models.

Test results

Table 6 displays the application of the best models on the validation set, based on two indicators: the best model in the multi-class problem (SVM trained with abstracts) and the best model in the binary problem (DistilBERT-base-cased trained with abstracts). The quantitative results reveal a dramatically different scenario compared to the validation set, despite the identical data generation process. In this case, performance is notably worse than random in the multi-class problem but demonstrates results similar to random in the binary problem. This suggests that the best models struggle to generalize effectively.

Subtask 2: Journal recommendation

In the case of the Journal recommendation, we utilized similar models to those employed in Subtask 1B. We conducted evaluations using three different inputs: title,

keywords and abstract. For the title input, the best-performing model was BERT- cased with an F1-macro of 0.54883, a Top-5 accuracy (Acc_5) of 0.90277 and a Top-10 accuracy (Acc_{10}) of 0.96982, exhibiting slightly better performance than DistilBERT, which was the top-performer in Subtask 1A (see Table 7). However, when using key- words, DistilBERT-cased

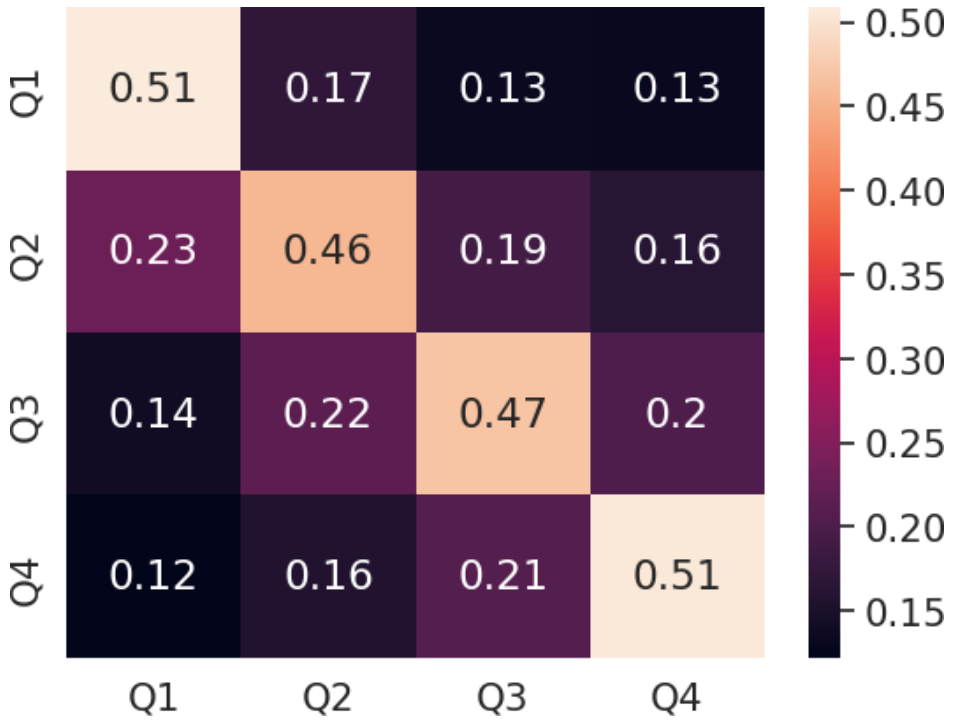

Figure 3: Confusion matrix of the best model (DistilBERT-base-cased) with the testing set.

Table 2: Model Performance Metrics using title.

Model	Ep	F1-m	F1-w	Acc	$F1\text{-}m_2$	$F1\text{-}w_2$	Acc_2
DistilBERT-cased	2	0.20173	0.19842	0.21744	0.58429	0.58296	0.58638
DistilBERT-cased	3	0.20708	0.19756	0.2049	0.5412	0.54219	0.54223
BERT-cased	2	0.18392	0.17958	0.19128	0.5614	0.56076	0.56185
BERT-cased	3	0.21067	0.19544	0.22234	0.56217	0.56364	0.56458
BERT-cased	4	0.20341	0.19796	0.21253	0.5984	0.59747	0.59946
BERT-cased	5	0.19187	0.18488	0.19619	0.55193	0.55161	0.55204
DistilBERT-uncased	2	0.25793	0.25209	0.27139	0.61362	0.6123	0.6158
DistilBERT-uncased	3	0.22863	0.22457	0.23869	0.6182	0.61902	0.61907
DistilBERT-uncased	4	0.23819	0.23428	0.24687	0.61444	0.61567	0.61635
SVM-kernel-linear		0.30577	0.30853	0.32145	0.66139	0.66082	0.66219

Table 3: Model Performance Metrics using keywords.

Model	Ep	F1-m	F1-w	Acc	$F1\text{-}m_2$	$F1\text{-}w_2$	Acc_2
DistilBERT-cased	2	0.23695	0.23536	0.23936	0.60737	0.60692	0.60759
DistilBERT-cased	3	0.23969	0.24255	0.26582	0.64808	0.64722	0.64902
DistilBERT-cased	4	0.22038	0.22398	0.25662	0.62226	0.62022	0.62716
DistilBERT-uncased	2	0.22235	0.21608	0.23705	0.67032	0.66968	0.67089
DistilBERT-uncased	3	0.21295	0.20779	0.22727	0.63724	0.63515	0.64269
BERT-cased	2	0.24034	0.24675	0.29804	0.5844	0.58039	0.60184
BERT-cased	3	0.22854	0.22548	0.23533	0.57826	0.57481	0.59091
SVM-kernel-linear		0.26011	0.26284	0.27586	0.62168	0.62049	0.62478

Table 4: Model Performance Metrics using abstracts.

Model	Ep	F1-m	F1-w	Acc	$F1\text{-}m_2$	$F1\text{-}w_2$	Acc_2
DistilBERT-cased	2	0.26015	0.25072	0.30609	0.77273	0.77265	0.77275
DistilBERT-cased	3	0.27683	0.2818	0.31072	0.71234	0.71334	0.71478
DistilBERT-uncased	2	0.30055	0.30712	0.34087	0.7153	0.71546	0.71536
DistilBERT-uncased	3	0.31504	0.31212	0.34145	0.74377	0.74375	0.74377
DistilBERT-uncased	4	0.29251	0.28594	0.30551	0.65267	0.65524	0.66609
SVM-kernel-linear		0.34296	0.34787	0.38399	0.72298	0.72254	0.72355

Table 5: Ensemble performance metrics on validation set.

Model	F1-m	F1-w	Acc	$F1\text{-}m_2$	$F1\text{-}w_2$	Acc_2
Deep Learning	0.24389	0.2742	0.23575	0.72521	0.72525	0.72522
SVM	0.32301	0.32699	0.35535	0.70509	0.70452	0.70602

Table 6: Final testing metrics.

Model	Ep	F1-m	F1-w	Acc	$F1\text{-}m_2$	$F1\text{-}w_2$	Acc_2
DistilBERT-cased	2	0.15685	0.15674	0.17177	0.48584	0.48598	0.48739
SVM (abstracts)		0.18468	0.18817	0.20924	0.49168	0.49160	0.49185

Table 7: Model Performance Metrics using title.

Model	Ep	F1-m	F1-w	Acc	Acc_5	Acc_{10}
DistilBERT-cased	2	0.47366	0.49305	0.51802	0.87008	0.97402
DistilBERT-cased	3	0.53272	0.54964	0.55574	0.89941	0.97569
DistilBERT-cased	4	0.5279	0.54852	0.55239	0.8969	0.96731
DistilBERT-uncased	2	0.4215	0.43704	0.46521	0.79631	0.92372
DistilBERT-uncased	3	0.4215	0.43704	0.46521	0.79631	0.92372
SciBERT-scivocab-cased	2	0.45742	0.47309	0.49036	0.83152	0.94971
SciBERT-scivocab-cased	3	0.46856	0.48288	0.49539	0.83319	0.943
SciBERT-scivocab-cased	4	0.46477	0.47937	0.4912	0.83152	0.93797
BERT-cased	2	0.52685	0.54984	0.55574	0.89187	0.96815
BERT-cased	3	0.54775	0.56626	0.57083	0.89438	0.97821
BERT-cased	4	0.54883	0.56286	0.5767	0.90277	0.96982
BERT-cased	5	0.52606	0.54326	0.54065	0.90277	0.96647
BERT-uncased	2	0.48925	0.49717	0.52221	0.82481	0.93462
BERT-uncased	3	0.53812	0.54222	0.55407	0.83236	0.94635
BERT-uncased	4	0.54328	0.54486	0.55742	0.84828	0.94971
BERT-uncased	5	0.53182	0.53933	0.5549	0.829	0.92791
SVM-kernel-linear		0.44551	0.45843	0.47946	0.79044	0.91534

outperformed other models (see Table 8). In evaluations with abstracts, DistilBERT-uncased demonstrated superior results and emerged as the overall best model (Table 9).

ENSEMBLE

In this specific case, the ensemble proved beneficial for the SVM, although it did not enhance the performance of the Deep Learning algorithms (see Table 10).

Test results

In the testing set (Table 11), we saw a similar scenario than in the validation set. In this case, DistilBERT, in general, presents healthy metrics and just slightly worse than in the validation set, indicating that the model is capable of correctly classifying unseen papers to the corresponding journal. This result differs from the results of Subtask 1B. The corresponding confusion matrix is presented in Figure 4.

Table 8: Model Performance Metrics using keywords.

Model	Ep	F1-m	F1-w	Acc	Acc_5	Acc_{10}
DistilBERT-cased	2	0.53399	0.5363	0.55658	0.84074	0.94803
DistilBERT-cased	3	0.55185	0.55801	0.56329	0.83236	0.95138
DistilBERT-cased	4	0.57642	0.57975	0.58676	0.8508	0.95054
DistilBERT-cased	5	0.55754	0.56329	0.56245	0.829	0.94384
SciBERT-scivocab-cased	2	0.53582	0.53689	0.55155	0.85331	0.95725
SciBERT-scivocab-cased	3	0.52683	0.52973	0.55071	0.84661	0.95725
BERT-base-cased	2	0.55859	0.561	0.57837	0.84577	0.94803
BERT-cased	3	0.56146	0.56845	0.57921	0.81978	0.94132
BERT-cased	4	0.54599	0.54959	0.5658	0.83738	0.94216
BERT-uncased	2	0.48666	0.48663	0.5197	0.81894	0.93713
BERT-uncased	3	0.51924	0.52296	0.54149	0.83738	0.95138
BERT-uncased	4	0.52612	0.52815	0.53982	0.82397	0.94635
BERT-uncased	5	0.53543	0.54174	0.54568	0.82984	0.94384
BERT-uncased	6	0.53437	0.53603	0.53898	0.82397	0.93294
DistilBERT-uncased	2	0.49941	0.50289	0.52389	0.83068	0.93797
DistilBERT-uncased	3	0.52903	0.52886	0.54736	0.83906	0.94216
DistilBERT-uncased	4	0.53085	0.53415	0.54652	0.83319	0.94468
DistilBERT-uncased	5	0.53967	0.54498	0.55071	0.84158	0.94216
DistilBERT-uncased	6	0.53098	0.53598	0.53982	0.82146	0.94384
SVM-kernel-linear		0.53503	0.53906	0.56245	0.83319	0.94216

Table 9: Model Performance Metrics using the abstract.

Model	Ep	F1-m	F1-w	Acc	Acc_5	Acc_{10}
DistilBERT-uncased	2	0.54478	0.54937	0.56832	0.85163	0.95977
DistilBERT-uncased	3	0.55177	0.55413	0.57251	0.85834	0.95893
DistilBERT-uncased	4	0.57718	0.58165	0.59765	0.86505	0.96982
DistilBERT-uncased	5	0.55379	0.55747	0.55993	0.86505	0.95809
DistilBERT-cased	2	0.55792	0.56261	0.58173	0.8751	0.96731
DistilBERT-cased	3	0.56408	0.56744	0.59095	0.87846	0.9715
DistilBERT-cased	4	0.55663	0.56496	0.58173	0.88097	0.96815
SVM-kernel-linear		0.55826	0.56516	0.58592	0.8684	0.96144

Table 10: Ensemble performance metrics on validation set.

Model	F1-m	F1-w	Acc
Deep Learning	0.57500	0.58049	0.59681
SVM	0.56046	0.56795	0.59179

In the testing set (see Table 11), we observed a similar scenario to that of the validation set. In this case, DistilBERT generally exhibits robust metrics, slightly

inferior to those in the validation set, suggesting the model's capability to accurately classify unseen papers into the corresponding journals. This result notably differs from the outcomes of Subtask 1B. The corresponding confusion matrix is presented in Figure 4.

DISCUSSION

In general, the application of different Machine Learning algorithms and recent Deep Learning models showed two relevant conclusions about the studied subtask:

1. The journal classification process is plausible, even with a high number of classes.

2. The quartile classification process remains open.

Table 11: Final testing metrics.

Model	Ep	F1-m	F1-w	Acc	Acc5	Acc10
DistilBERT-base-cased	4	0.5548	0.54894	0.56245	0.86421	0.95809

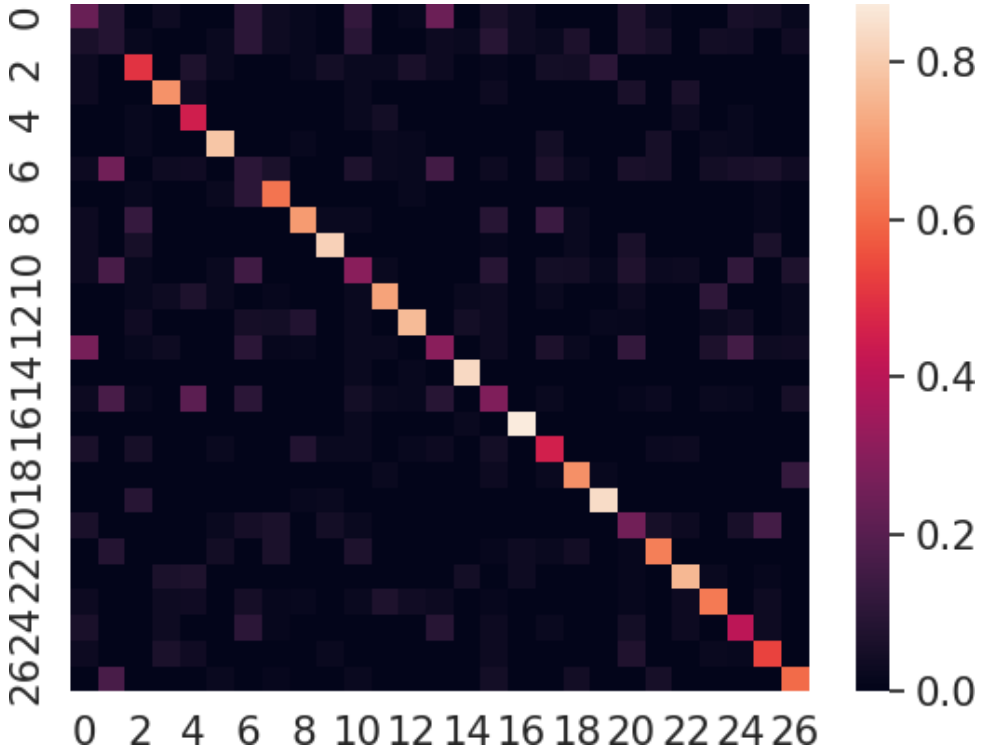

Figure 4: Confusion matrix of the best model applied in the testing set (DistilBERT-base-cased).

Various approaches have been attempted for journal prediction, but quartile classification has not been extensively explored in the literature. This may be due to the apparent challenge of predicting quartiles based on current technologies. Why did all models fail in a different task with the same data and fewer classes? It is intriguing to note that, in some cases, the trained models even performed worse than random ones, providing potential insights into the challenges of generalization.

As Subtask 1A demonstrates, it is indeed possible to predict the quartile of journals that appeared in the training set. In this regard, the confusion matrix of the best model for this subtask (see Figure 3) shows that similar quartiles are predicted similarly, apparently showing that the classifiers fail to distinguish similar quartiles. Nevertheless, considering the results of Subtask 1B, perhaps this confusion was produced by the presence of similar journals in similar quartiles, but this idea is debatable.

Generalizing to unseen journals proves to be more challenging. None of the studied classifiers produced good performance in the complete subtask with the four quartiles, although the distinction between Q1-Q2 and Q3-Q4, was possible in the validation set. However, the best classical Machine Learning classifier and the best Deep Learning algorithm failed to generalize to the testing set, which did not include the journals of the training set. In general, since most methods performed similarly, the main problem may rely on the insufficiency of the information to accurately predict the quartile without depending on predicting the journal. By applying sentence embeddings with the all-MiniLM-L6-v2 model[20] and Principal Component Analysis, we visualize the cluster of data points labeled with quartiles (Figure 5 in the validation set and Figure 6 in the testing set) and with journals (Figure 7). It is evident that there is a lack of structure in both quartile distributions, but some clusters emerge when labeled by the journal. In Figure 7 we can see that some classes are clusterized in different regions, although we can comprehend that there are many outliers. For example, a journal devoted to Natural Language Processing (for instance, *Natural Language Engineering*) would contain most journals with titles semantically similar to the main topic, but this does not exclude the possibility that one NLP article cannot be found in a very different journal, due to the possibility of finding special issues, the presence of interdisciplinarity, or the existence of very general journals like *Artificial Intelligence*. This might explain why the metrics in Subtask 2 are low in comparison with other NLP tasks.

In the case of quartile prediction, the problem is even bigger. If we have a perfect journal recommendation system, we do have a perfect quartile prediction algorithm, but it might not be able to generalize for unseen journals. Figures 5 and 6 depicts

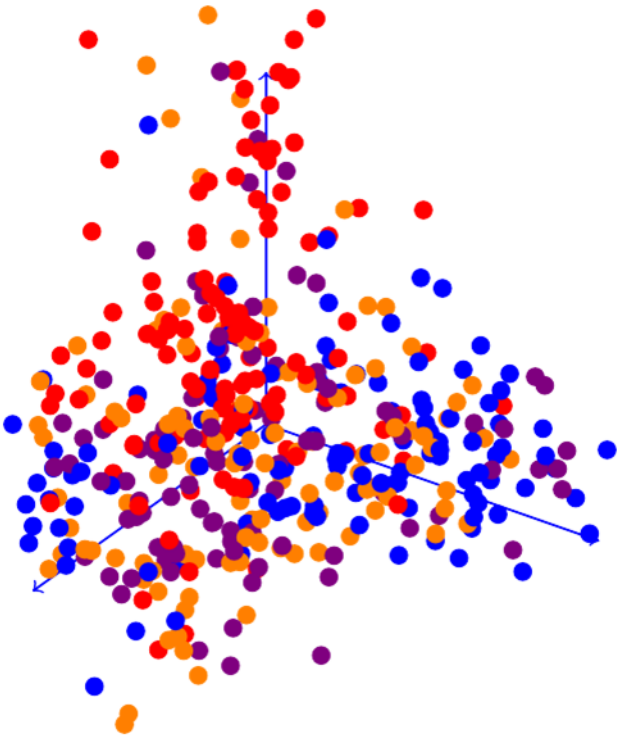

Figure 5: Scatter plot of different journals labeled with the journal quartile using the validation set. Orange dots represent the Q1 class, red dots represent the Q2 class, violet dots represent the Q3 class, while blue dots stand for the Q4 class.

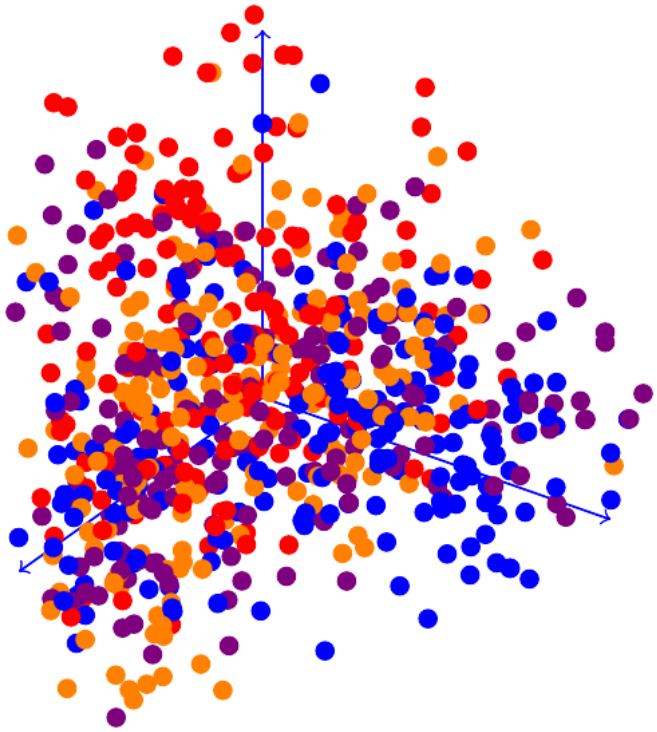

Figure 6: Scatter plot of different journals labeled with the journal quartile using the testing set. Orange dots represent the Q1 class, red dots represent the Q2 class, violet dots represent the Q3 class, while blue dots stand for the Q4 class.

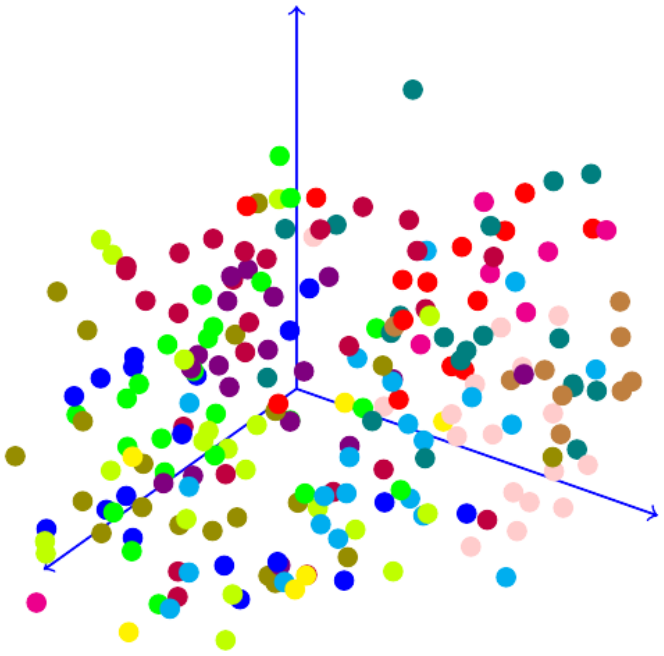

Figure 7: Scatter plot of different journals labeled with 14 different journals.

the situation. Although there are some areas with one color, dots from different classes can be found in the same regions. In both figures, we see a similar distribution of the classes, but the quantitive results suggest otherwise. In general terms, the task of predicting the quartile might need more information than the one provided for this problem.

Another important explanation for this phenomenon can be deduced from the metrics. As observed, some models performed better for the binary classification problem while others excelled in the multi-class problem. Moreover, clear results lower than random or trivial algorithms surfaced. For this reason, it can be concluded that the training process biased the classifiers and, in all cases, quartile prediction is an indirect form of journal prediction.

CONCLUSION

In this article, we explored two closely related text classification problems: quartile prediction and journal classification for a given paper. Despite the apparent connection, these two tasks are more intertwined than initially perceived. As demonstrated in the literature, the subtask of journal estimation has been reasonably well-executed. Comparing various approaches, we implemented recent Large Language Models, including GPT-2. Surprisingly, more modest models outperformed larger ones and classical approaches also exhibited relatively good performance when compared with Transformer architectures.

In the task of quartile prediction, the results are controversial. The initial experiment (Subtask 1A) showed that it is feasible to train

Machine and Deep Learning models for this specific subtask. In this instance, we combined journals in the training, validation and testing sets, achieving relatively good results, at least better than random. However, upon separating journals (Subtask 1B), we observed that quartile prediction becomes challenging. This suggests that the quartile prediction process indirectly predicts the journal to classify the article's quartile. However, this process

struggles to generalize, or at least, it is not achievable with the studied methods. Despite employing several state-of-the-art models, the results appear inconclusive.

It seems that current classifiers face challenges in predicting quartiles effectively, potentially due to insufficient data or indistinguishable classes. Nonetheless, we assert that this topic warrants further research, offering the potential to pave the way for improvements in the automated evaluation of articles.

FUNDING

The authors wish to thank the support of the Instituto Politécnico Nacional (COFAA, SIP-IPN, Grant SIP 20240610) and the Mexican Government (CONAHCyT, SNII).

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

AUTHOR CONTRIBUTION

Fernando Aguilar-Canto: Conceptualization, Methodology, Writing-original draft, Formal analysis, Software, Investigation. Cesar Macias: Conceptualization, Methodology, Writing-original draft, Formal analysis, Software, Investigation. Alberto Espinosa-Ju´arez: Data acquisition. Marco Antonio Cardoso-Moreno: Writing-review. Hiram Calvo: Supervision, Project administration.

ABBREVIATIONS

COFAA: Comisión para el Fomento de Actividades Académicas; **SIP-IPN:** Secretaria de Investigación y Posgrado del Instituto Politécnico Nacional; **CONAHCyT:** Consejo Nacional de Humanidades, Ciencia y Tecnología; **SNII:** Sistema Nacional de Investigadoras e Investigadores.

REFERENCES

1. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN. Kaiser, L-, Polosukhin, I: attention is all you need. Adv Neural Inf Process Syst. 2017;30.
2. Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805 2018.
3. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I, *et al.* Language models are unsupervised multitask learners. OpenAI blog 1(8). Vol. 9; 2019.
4. Pohan HI, Warnars HL, Soewito B, Gaol FL. Recommender system using transformer model: A systematic literature review. In: 1st International Conference on Information System & Information Technology (ICISIT). IEEE PUBLICATIONS; 2022. p. 376-81. doi: 10.1109/ICISIT54091.2022.9873070.
5. Gheewala S, Xu S, Yeom S, Maqsood S. Exploiting deep transformer models in textual review-based recommender systems. Expert Syst Appl. 2024;235:121120. doi: 10.1016/j.eswa.2023.121120.
6. Beel J, Gipp B, Langer S, Breitinger C. Research-Paper recommender systems: a literature survey. Int J Digit Libr. 2016;17(4):305-38. doi: 10.1007/s00799-015-0156-0.
7. Feng X, Zhang H, Ren Y, Shang P, Zhu Y, Liang Y, *et al.* The deep learning-based recommender system "pubmender" for choosing a biomedical publication venue: development and validation study. J Med Internet Res. 2019;21(5):e12957. doi: 10.2196/12957, PMID 31127715.
8. Nguyen D, Huynh S, Huynh P, Dinh CV, Nguyen. B.T.: S2cft: a new approach for paper submission recommendation. In: SOFSEM 2021: theory and practice of computer science: 47th International Conference on Current Trends in Theory and Practice of Computer Science. Bolzano, Bozen: SOFSEM; 2021.
9. Springer Huynh ST, Huynh PT, Nguyen DH, Cuong DV, Nguyen, B.T.: S2rscs: An efficient scientific submission recommendation system for computer science. Italy, January 25-29, 2021. Proceedings. 2021;47:563-73. In: Trends in artificial intelligence theory and applications. Artificial intelligence practices. Proceedings: 33rd International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2020, Kitakyushu, Japan, Sep 22-25, 2020. Vol. 33. Springer; 2020. p. 186-98.
10. Nguyen DH, Huynh ST, Dinh CV, Huynh PT, Nguyen BT. Psrmte: paper submission recommendation using mixtures of transformer. Expert Systems with Applications. 2022;202:117096.
11. Michail S, Ledet JW, Alkan TY, İnce MN, Günay M. A journal recommender for article submission using transformers. Scientometrics. 2023;128(2):1321-36. doi: 10.1007/s11192-022-04609-x.
12. Liu C, Wang X, Liu H, Zou X, Cen S, Dai G. Learning to recommend journals for submission based on embedding models. Neurocomputing. 2022;508:242-53. doi: 10.1016/j.neucom.2022.08.043.
13. Hassan HA, Sansonetti G, Gasparetti F, Micarelli A, Beel J. Bert, elmo, use and infersent sentence encoders: the panacea for research-paper recommendation? RecSys (Late-Breaking Results). 2019:6-10.
14. Zhao X, Kang H, Feng T, Meng C, Nie Z. A hybrid model based on lfm and bigru toward research paper recommendation. IEEE Access. 2020;8:188628-40. doi: 10.1109/ACCESS.2020.3031281.
15. Barolli L, Di Cicco F, Fonisto M. An investigation of Covid-19 papers for a content-based recommendation system. Adv Oncol P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 16th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2021), pp. 156- 164 (2022). Springer.
16. Sanh V, Debut L, Chaumond J, Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108 2019.
17. Beltagy I, Lo K, Cohan A. SciBERT: A pretrained language model for scientific text. arXiv preprint arXiv:1903.10676 2019.
18. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, *et al.* RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692 2019.
19. Conneau A, Khandelwal K, Goyal N, Chaudhary V, Wenzek G, Guzmán F, *et al.* Unsupervised Cross-lingual Representation Learning at Scale. arXiv preprint arXiv:1911.02116 2019.
20. Wang W, Wei F, Dong L, Bao H, Yang N, Zhou M. MiniLM: deep Self-Attention Distillation for Task-Agnostic Compression of PreTrained Transformers. Adv Neural Inf Process Syst. 2020;33:5776-88.

Cite this article: Aguilar-Canto F, Macias C, Espinosa-Juárez A, Cardoso-Moreno MA, Calvo H. Quartile Prediction and Journal Recommendation Using Deep Learning Models for Artificial Intelligence Articles. J Scientometric Res. 2025;14(1):373-82.