University of Twente

Natural Language Processing for journal publication selection

How Natural Language Processing can be used to aid in selecting the right journal for a specific publication.

Jason Hsu (s2708442)
Jump Srinualnad (s2690837)
Tintin Wongthanaporn (s2712911)
Thijs Beumer (s2761831)

Technical Computer Science

N. Bouali

10 - 11 - 2023

Contents

1 Goal and Motivation

The process of publishing scientific findings in journals is often a time-consuming task. This selection is not just another step in the process, but also an important factor for reaching the intended audience and thereby maximising the overall impact of the research within the academic community.

To ease this process, our objective was to develop a Machine Learning model and a Journal Finder web application. This application is designed to give the top 5 most relevant journals for a specific article by considering the subject area(s) and the abstract for that article. The model used by the application would be trained on a large dataset consisting of 40,001 open-access articles originating from the Elsevier Data Repository[1]. The usage of this model in the journal selection process could improve the efficiency with which important scientific knowledge can reach the broader academic environment.

2 Methods

Data Preparation

The data preparation process involved handling a dataset comprising approximately 40,001 articles in JSON format. The objective was to extract and structure key features, such as document identifiers (docID), abstracts, body texts, keywords, subject areas, and journal identifiers (DOIs), into a proper data frame, and then format and prepare the dataset for training properly. In this phase, we will ensure the dataset's readiness for subsequent scientific analysis.

First of all, it is essential to note that certain entries within the dataset were incomplete, lacking one or more of these essential features, rendering them unsuitable for subsequent analysis. Consequently, we used the listwise deletion, resulting in a refined dataset consisting of 35,370 usable articles.

Subsequently, we transformed the dataset into CSV (Comma-Separated Values) format. This decision was based on the advantages associated with CSV files over JSON files, primarily related to the efficiency of data retrieval and processing.

To further prepare the dataset, the body text and journal identifiers (DOIs) were subjected to specific extraction processes.

For the body text extraction, the Python function **"ast.literal_eval(data)"** was employed. This function safely evaluates a string, which represents a Python literal expression, and transforms it into the corresponding Python data structure. Finally, all individual sentences are extracted from the body text and ordered by the sentence offset before it is concatenated to construct comprehensive paragraphs for each article.

In the case of journal identifier (DOI) extraction, a selective approach was adopted. The process involved the removal of specific components from the DOI, namely, the prefix, the publication year, and the article identifier, and the remaining component is the journal's name (we will call it **"doi"** in the remaining report for simplicity). For example, the DOI **"10.1016/j.aap.2013.07.029"** was transformed into **"aap"**.

Last but not least, a certain threshold was chosen to ensure that journals with an insufficient number of associated documents, specifically those containing fewer than 22 articles are excluded from the data because this is the average number of documents for each journal. This step is crucial as it focuses on retaining journals with sufficient representation. Subsequently, journals failing to meet the established threshold are filtered out, resulting in a refined dataset that concentrates on journals with a more substantial corpus of 22 or more associated articles. This selection process is an essential data processing step, ensuring that the subsequent analyses and modelling are based on a more representative subset of journals. In total, our dataset contains 378 distinct journals, each of which is categorised as a subset of 26 diverse subject areas.

Modelling

After completing the data preparation phase, we were prepared to train our models. Initially, we planned to incorporate the abstract, body text, subject area, and keywords for training purposes. However, after experimentation and careful deliberation, we decided to train models only with the abstract and the subject area. Comprehensive reasons for excluding body text and keywords can be found in Section 6. In this section, we will introduce our models and explain how we combine outcomes from multiple models to derive the final prediction.

Abstract models (X: abstract, y: doi)

To train abstract models, we first partitioned articles in the dataset based on the designated subject area; it resulted in 26 subsets of articles, and they are not mutually exclusive because articles can be related to multiple subject areas. Next, we developed dedicated models for each subject area and used 'abstract' to train the model. Below are the steps we employed to develop each abstract model:

1. We converted abstracts to lower-case and applied clean text to remove numbers and special characters.
2. We one-hot encoded the label (y: doi).
3. We tokenized abstracts with the SciBERT tokenizer and utilised the SciBERT model to generate contextual embeddings from the [CLS] token.
4. We defined a neural network model that contains an input layer that takes the embeddings and an output layer that uses the softmax activation function. Categorical cross-entropy loss function was used for this multiclass classification problem.

We utilised the SciBERT model ("allenai/scibert_scivocab_uncased") during our training on abstract models because it's pre-trained on a massive corpus of scientific literature and research papers, which is well-suited for our project, and it's able to generate embeddings that represent features of abstracts. Additionally, we selected the embeddings from the [CLS] token, a special token included at the beginning of the input sequence, since it can capture the contextual information of the entire input.

Subject area model (X: subject area, y: doi)

Here we only use the subject area to predict the journals. Given that the subject areas for each article are represented in a format such as "['PHYS', 'BOIC']," we can employ a bag-of-words approach coupled with a multiclass neural network for the training to predict the journals. We follow the below steps when implementing this model:

1. We vectorized subject areas with ***"CountVectorizer"***.
2. We one-hot encoded the label (y: doi).
3. We defined a softmax regression where the output layer is a dense layer with units equal to the number of unique classes from the label. We used the softmax activation function and categorical cross-entropy loss function for this multiclass classification problem.

Combine models

In our attempt to rank the top 5 most relevant journals, we developed 27 models, comprising 1 subject area model and 26 abstract models. We employed the "accuracy-based weight" as our averaging criterion to determine the final prediction. With this criterion, the weight for each model will be dynamic because they will change based on the subject areas selected. The weight is calculated by dividing the accuracy of each model by the sum of the accuracies of all models used in the prediction.

The formula for calculating the model weight is: $\frac{model\ accuracy}{sum\ of\ all\ model\ accuracy}$.

The accuracies employed for weight calculations are all derived from the test set as those accuracies shown in Section 3. For instance, when determining journals suitable for a paper with subject areas "BIOC" and "ENVI," the weight for the BIOC model will be computed as follows: $\frac{BIOC\ model's\ accuracy}{BIOC\ model's\ accuracy + ENVI\ model's\ accuracy + Subject\ area\ model's\ accuracy}$. Given the diverse range of accuracies observed across various models ***(Fig 3.2)***, a simplistic averaging approach proves insufficient for consolidating predictions. To address this challenge, we employ a dynamic weighting mechanism where higher accuracies contribute more significantly to the final predictions. With the dynamic weight and the predictions derived through the softmax activation function, we can obtain better-combined journals' rankings. The results and evaluation of our combined model can be found in Section 3.

In conclusion, the integration of both the subject area and abstract models results in a comprehensive journal recommendation system. The abstract models ensure that recommendations align with the contextual nuances of each article's content, while the subject area model offers a practical solution when content-specific information is limited, providing domain relevancy. By utilising both models, we can provide an effective approach to journal prediction.

3 Results and Evaluation

Abstract models' results:

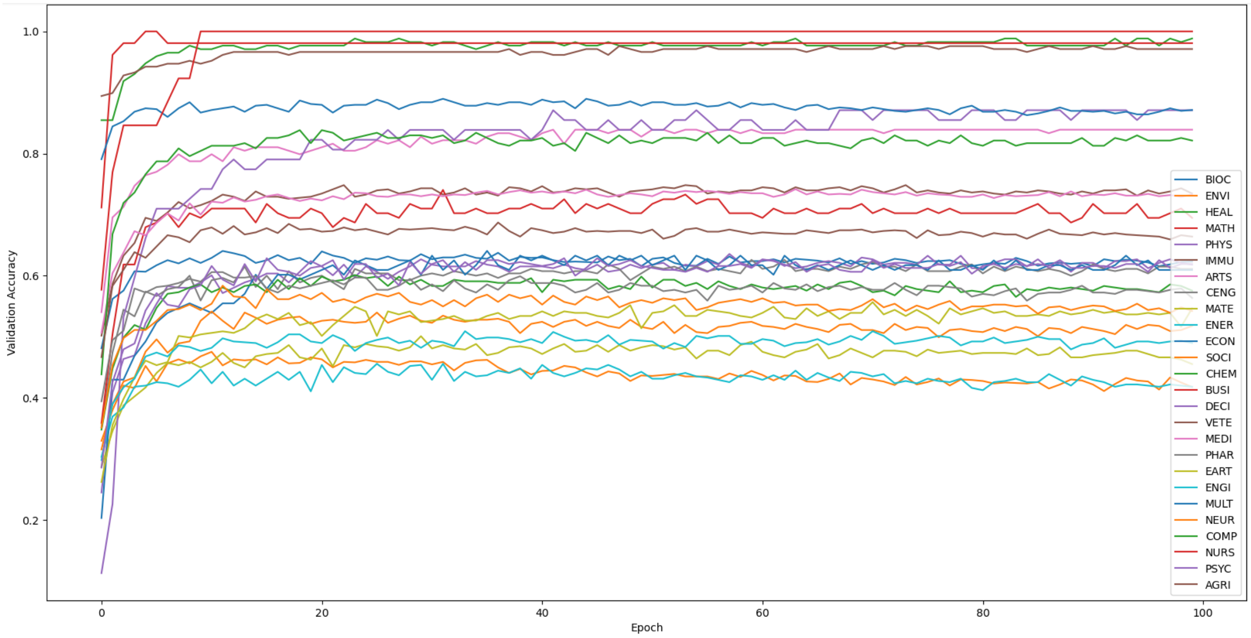


Fig. 3.1 The learning progression of each abstract model

Fig. 3.1 reveals variations in the learning patterns across different subject areas. The variability in accuracy can be attributed to the diverse nature of subject areas. Some subjects encompass a larger volume of journals to predict, which can lead to increased computational complexity and model training challenges. ***Fig. 3.1*** could be difficult to interpret the accuracy, below is the graph (***Fig. 3.2)*** that clearly shows the final accuracy of each abstract model.

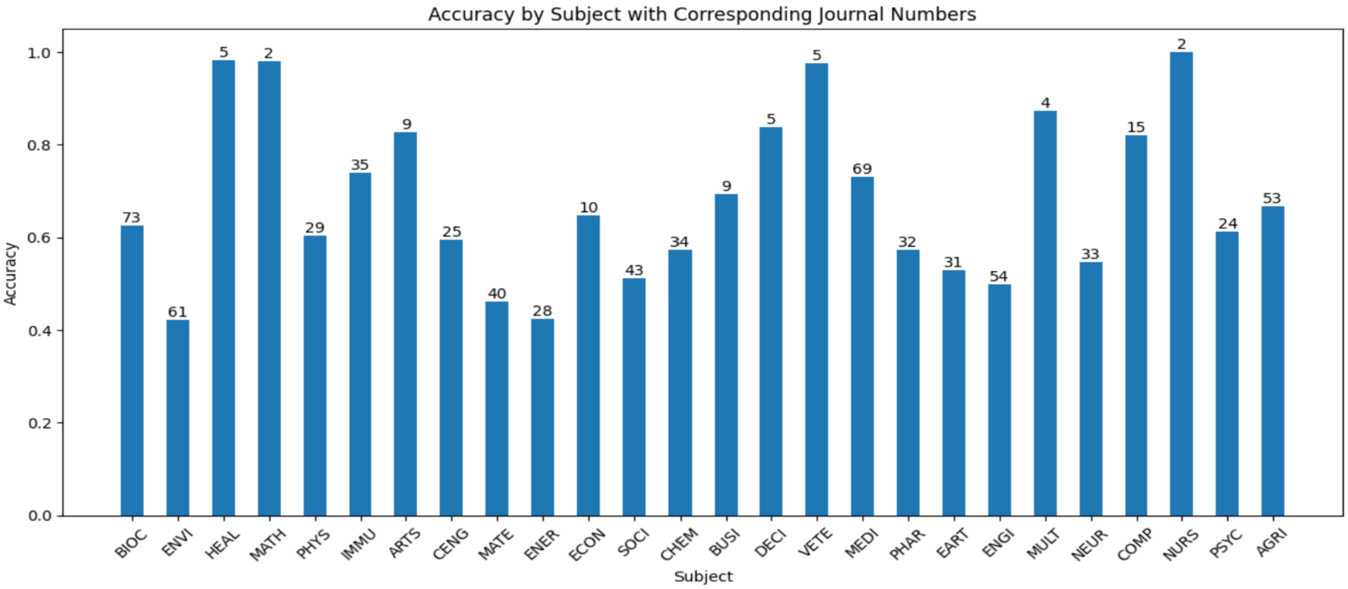


Fig. 3.2 The final accuracy of each abstract model

Fig. 3.2 displays the concluding accuracy scores of all individual models when evaluated on the testing dataset. The numerical values atop each bar indicate the number of journals in the respective subject area. ***Fig. 3.2*** reaffirms the inverse relationship between the volume of journals for prediction and model accuracy. Notably, subject areas with fewer journals, such as "MATH" and "NURS", each comprising only two journals, exhibit higher levels of accuracy, underscoring a notable performance advantage. Models with a greater number of journals to predict such as "MEDI" or "BIOC" also exhibit commendable accuracy.

Overall, ***Fig. 3.2*** suggests that the complexity of predicting journals in subject areas with a larger volume of journals adversely impacts model accuracy, whereas subject areas with a smaller number of journals tend to achieve higher accuracy levels.

Subject area model's results:

Test set size: 15%
Technique: softmax regression
Accuracy on train set: 0.59426
Accuracy on test set: 0.60100

In ***Fig. 3.3***, an accuracy rate of 60% on the test set demonstrates the model's significant capability in effectively contributing to the final journal rankings. In contrast, assuming a uniform distribution of classes, random guessing would result in a mere 0.26% accuracy (1 out of 378). Therefore, the model outperforms random guessing by a substantial factor of approximately 226.92 times, highlighting its valuable predictive performance.

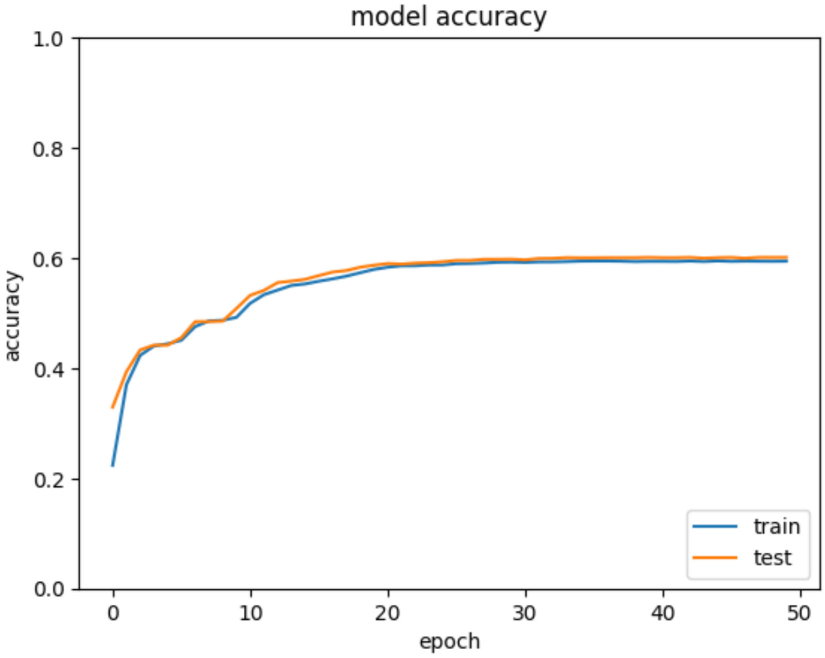


Fig. 3.3 The learning progression of the Subject area model

Combined model's (dynamic weight) results:

Test size: 3000
Test Accuracies:

True accuracy (true journal is ranked top 1)	76.8%
Top 3 accuracy (true journal is ranked within the top 3)	93.4%
Top 5 accuracy (true journal is ranked within the top 5)	96.9%

We can see from the Test Accuracies table above that the Top 5 accuracy is almost 97%, which is the reason why we chose to show the top 5 relevant journals to the user.

What we also discovered is that the accuracy of the combined model was relatively low when an article belonged to fewer subject areas. However, this drawback is mitigated after implementing dynamic weight prediction, which also contributes to the overall robustness of our models.

In the next section, we will introduce our Journal Finder web application and explain how users can interpret the results it provides. Given the nice predictive performance of our models, as shown in this section, we believe our Journal Finder will serve as an effective platform for authors seeking the most fitting publication venues for their research.

4 Journal Finder Web Application

The web application allows for easy usage of the model. Whenever one would like to find the best matching journal, the abstract and subject areas can be provided in the initial window of the web application as seen in ***Fig. 4.1***.

After providing this information and clicking on the “Find Relevant Journals” button, the application will automatically start the process of determining the most suitable journals. When this process is completed, the user will be taken to the results screen which an example is shown in ***Fig. 4.2***.

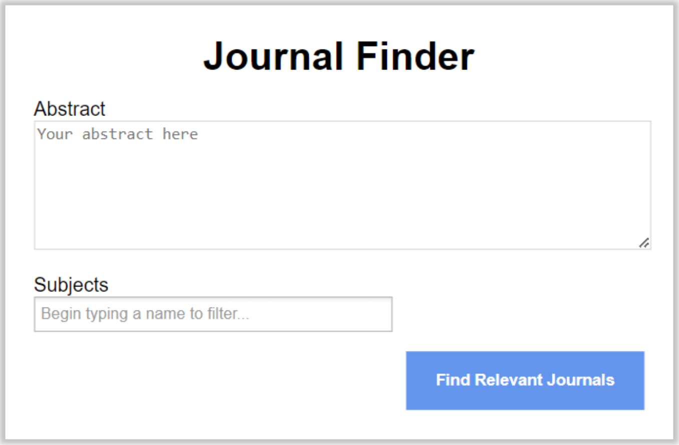
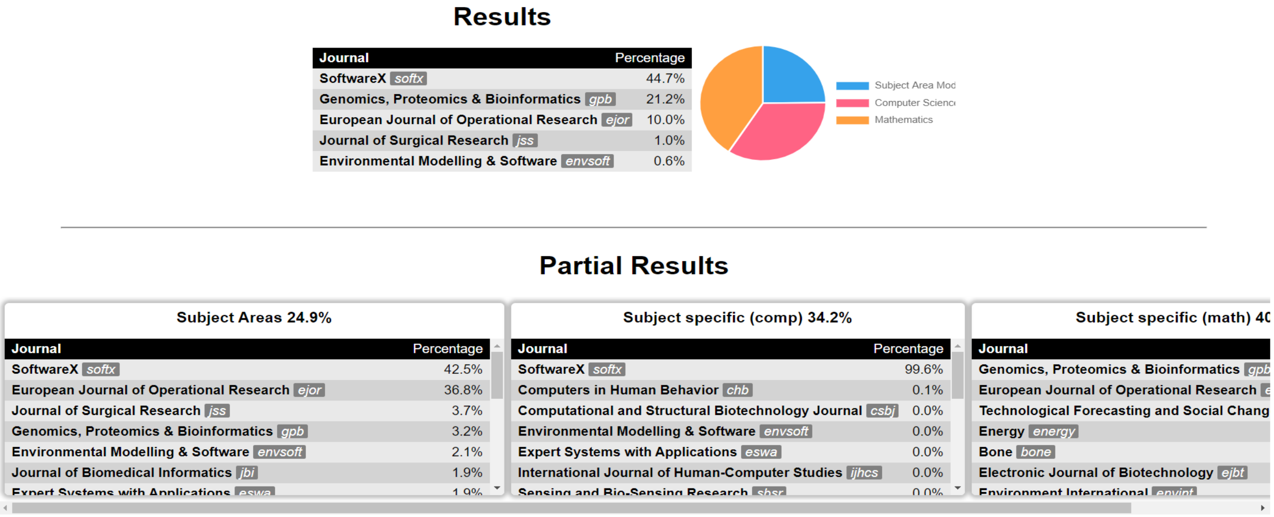


Fig. 4.1 Search page



Results

Journal	Percentage
SoftwareX softx	44.7%
Genomics, Proteomics & Bioinformatics gpb	21.2%
European Journal of Operational Research ejor	10.0%
Journal of Surgical Research jss	1.0%
Environmental Modelling & Software envsoft	0.6%

Partial Results

Subject Areas 24.9%

Journal	Percentage
SoftwareX softx	42.5%
European Journal of Operational Research ejor	36.8%
Journal of Surgical Research jss	3.7%
Genomics, Proteomics & Bioinformatics gpb	3.2%
Environmental Modelling & Software envsoft	2.1%
Journal of Biomedical Informatics jbi	1.9%
Expert Systems with Applications eswa	1.9%

Subject specific (comp) 34.2%

Journal	Percentage
SoftwareX softx	99.6%
Computers in Human Behavior chb	0.1%
Computational and Structural Biotechnology Journal csbj	0.0%
Environmental Modelling & Software envsoft	0.0%
Expert Systems with Applications eswa	0.0%
International Journal of Human-Computer Studies ijhcs	0.0%
Sensing and Bio-Sensing Research sbsr	0.0%

Subject specific (math) 40

Journal
Genomics, Proteomics & Bioinformatics gpb
European Journal of Operational Research e
Technological Forecasting and Social Chang
Energy energy
Bone bone
Electronic Journal of Biotechnology ejbt
Environment International envint

Fig. 4.2 Result page

In the top left corner of ***Fig. 4.2***, the best matching journals are displayed. The percentage shown in this table is the model's determination of how well fitting the abstract and subject areas are to that journal. Next to this results tab, there is a pie chart showing how much influence every separate model has had on these total results. Lastly, in the partial results section, the best-fitting journals determined by every model separately are shown.

5 Conclusion

Through our Journal Finder and all the models that we have trained, we believe that we have achieved our goal of assisting authors in identifying a fitting journal for their research articles and making the process of publishing scientific findings in journals less time-consuming task. We ensure the results are as transparent as possible by allowing the author to view details of the results of all models used for the predicting article and their respective weights. While the Journal Finder can provide a list of the top 5 most relevant journals, it's important to note that authors may still need to manually verify and select the most appropriate journal, taking into account factors like publication fees and personal preferences. Nonetheless, the results obtained from our Journal Finder are expected to significantly reduce the time and effort authors initially invest in this selection process.

Furthermore, in contrast to existing solutions relying solely on abstracts for journal predictions, by integrating both subject area and abstract information, our model enhances prediction accuracy and specificity. This combination harnesses the domain relevance provided by the subject areas, offering authors a more nuanced understanding of their research content. Thus, this approach addresses the limitations of abstract-only prediction models, providing authors with a more sophisticated tool for precise journal recommendations.

6 Reflection

During this project, we utilised several pieces of knowledge that we gained from Module 9, especially from our coursework in data quality and machine learning classes. In this section, we will identify the techniques we learned in class and elucidate their practical application within the project. Furthermore, we will explain the reason for the exclusion of keywords and body text when we trained our models.

Application of academic knowledge

In the data preparation phase, we employed the data quality technique we acquired in class to address data instances with missing values. Specifically, we applied listwise deletion (we assumed data were missing completely at random) to remove all the data instances with missing values before we extracted the features we needed to train our models.

Moreover, the knowledge we learned from the natural language processing courses and the laboratories empowered us to select appropriate models to train during this project and proficiently implement them in the Python programming environment. It is worth noting that despite not having received specific instruction on the SciBERT model during our coursework, our foundational knowledge acquired in class, coupled with supplementary online resources, facilitated our understanding and integration of this model into our project.

Exclusion of keywords and body text

Initially, we attempted to use keywords as the independent variable (X) to predict the respective journal categories (y). This approach was chosen with the expectation of identifying journals based on the presence of specific keywords. However, the results were less promising than anticipated. The model's accuracy on the test set hovered at approximately 28%. This level of accuracy, while significant, does not meet the desired benchmark compared to other models. It became evident that relying solely on keywords, may not capture the context that defines journal classifications.

Regarding the body text, we attempted to use it as the independent variable (X) in our modelling approach as well. While this approach offers the advantage of providing a comprehensive and detailed source of information, we encountered significant practical challenges. The body text of each article consists of over 20,000 words. To effectively make use of this wealth of information, we discussed employing deep learning techniques such as Recurrent Neural Networks (RNN) or the approach we adopted to deal with the abstract. However, it became apparent that training models on such extensive textual data would be excessively time-consuming and resource-intensive. Despite the potential advantages in terms of predictive accuracy, the computational demands and time constraints posed substantial obstacles. Thus, we opted not to use body text due to the impracticality of training complex models on such voluminous text data. Instead, we sought a compromise between computational feasibility and predictive power, ultimately guiding our decision-making process to focus on more manageable features for journal prediction.

7 Ethical perspective and framework

In shaping our Journal Finder, we have considered the ethical perspective. As a user-centric application, we provide the top 5 most relevant journals, accompanied by the associated percentage of predictions. This ethical approach ensures transparency in our system's suggestions, empowering users with a diverse set of choices. By offering clear prediction information and multiple recommendations, we uphold ethical standards, prioritising user autonomy in the decision-making process. Our commitment is to provide a tool that not only recommends but also respects users' research needs and preferences, aligning with ethical principles in the development and deployment of our Journal Finder.

References

[1] Kershaw, Daniel; Koeling, Rob (2020), “Elsevier OA CC-BY Corpus”, Elsevier Data Repository, V3, doi: 10.17632/zm33cdndxs.3