An organizational approach for discipline prediction in research projects

Hoang-Son Pham, Amr Ali-Eldin, Hanne Poelmans
ECOOM-UHasselt, Data Science Institute, Hasselt University, Belgium
{hoangson.pham, amr.alieldin, hanne.poelmans}@uhasselt.be

I. Introduction

The prediction of research disciplines has gained increasing attention in recent years due to its potential implementations in a variety of fields, such as academic advising, career counseling, and academic research funding allocation. Research information systems storing projects (meta) data play a crucial role in managing and evaluating research (meta) data across different disciplines and fields of study. In this context, research projects are manually assigned one or more research disciplines to facilitate this process. This is usually done by research administrators due to the limited time the principal researchers themselves might have. In addition to being rather subjective and time-consuming, this can lead to inconsistencies in discipline assignments and hence impact the quality of data used for monitoring and reporting. To address these limitations various approaches have been proposed, in the literature, to predict disciplines associated with research documents, e.g., publications, and projects. The frequently used methods in bibliometrics are bibliographic coupling, co-citation, and direct citation [1]. These approaches used citation network analysis techniques to determine the disciplines related to a publication. More recently, machine learning techniques have been applied to classify research documents [2]–[4]. In these approaches, the publications' abstracts were used as features to predict related disciplines. Machine learning techniques have been demonstrated to perform better than traditional approaches in bibliometrics. Although these approaches are useful, they may not be applicable to research information systems that lack citation data or have low-quality abstracts.

In this paper, we propose a novel approach to predict the disciplines of research projects in a research information system. The proposed approach uses machine learning algorithms and extracted disciplines from researchers and their related information such as organizations, projects, co-authors on projects, publications, and co-authors on publications. By analyzing the disciplines from these resources, the proposed model can predict each project's most appropriate research disciplines, providing a more objective and consistent approach to discipline assignment. This approach is helpful when there are no citation data or high-quality abstracts available. In the following sections, we describe the development and evaluation of our model, including the data sources and methods used to train the machine learning algorithms, as well as the performance metrics used to evaluate the accuracy and effectiveness of the proposed approach.

II. Data and method

In this work, we used data available on Flanders Research Information Space (FRIS)[1]. This is a current research information system (CRIS) governed by the Department Economy, Sciences and Innovation (EWI) of the Flemish government. The FRIS research portal discloses information on (partially) publicly funded research (e.g. researchers, research institutions, projects, and publications) from over 40 data providers in Flanders. The FRIS portal is used for reports, analysis, and statistics in the context of policy-making and for monitoring trends in research and innovation. Each object, e.g., a project, a publication, or a researcher, is assigned one or more research disciplines based on the Flemish Research Discipline Standard (FRDS) [5].

In the FRIS portal, for a given research project, we can find a list of researchers and their related information, such as the affiliations they work for, the projects and publications they have worked on, and their co-authors. We assume that the expertise of researchers involved in a project can be used to determine the project's disciplines. Therefore, in this work, we attempt to use deep machine learning to predict the disciplines of projects based on the disciplines of researchers and other resources related to them.

To do that, we create a training dataset, in which predictor variables are the disciplines related to researchers, whereas, output variables are the disciplines of the projects. To generate the predictor variables, we follow these steps. For each project, we gather information about the researchers involved. We collect N disciplines for each researcher from 6 sources: their profiles, organizations, projects they have worked on, their co-authors on those projects, their publications, and their co-authors on those publications. Therefore, for each researcher, we have a matrix with N rows and 6 columns representing their disciplines. Assuming a project has n researchers, we create a matrix representing the project by summing the matrices of all the researchers involved in the project. This results in a matrix with N rows and 6 columns, which represents the project's disciplines. After completing this process for all projects, we obtain a 3-dimensional matrix with the first dimension being the number of projects. Each project is represented by a matrix with N rows and 6 columns, which

[1] https://researchportal.be/en

represents the project's disciplines as a combination of the researchers' disciplines.

Output variables (or labels of the projects) are disciplines assigned to the projects. Note that each project is assigned one or more disciplines from a set of N disciplines. In this work, we used disciplines at the second level of FRDS, which includes 42 discipline codes. More granular levels (the third and/or fourth levels) were mapped onto their corresponding, hierarchical higher second-level disciplines. For example, suppose a researcher r has a set of disciplines: $(01010101, 01010103, 01020201)$, then the disciplines of r will be reduced to the second level codes as $(0101, 0101, 0102)$.

In theory, any multi-label classification model can be applied to build a classifier. In this study, however, we propose to use recurrent neural networks, i.e., Long Short-Term Memory (LSTM) neural networks [6], to build a classification model. The LSTM algorithm has proven to be a powerful tool for modeling sequential data and has been widely adopted in many applications in industry and academia.

To evaluate the performance of the model, we used Precision, Recall, and F1-scores metrics. However, since the number of labels is much larger than the number of labels in a given sample, then the one-hot encoded label vector for each sample will have many zeros, which can make the above metrics less informative. For example, if we have 42 possible labels, but the average sample has only two or three labels, then the one-hot encoded label vector for each sample will have 39-40 zeros and only 2-3 ones. In this case, a model that predicts all zeros for every sample will be highly accurate, even though it is not helpful for the actual task.

To address this issue, we propose to use evaluation metrics that consider the number of labels per sample, such as the Hamming loss or the Jaccard index. The Hamming loss is the fraction of labels that are incorrectly predicted for all samples, while the Jaccard index measures the similarity between the predicted labels and the true labels for a given sample. A low score of the Hamming Loss means that the model is making very few errors in predicting the class labels for the dataset. Similarly, for the Jaccard Index, a score of one indicates a perfect match between the predicted and actual labels, while a score of zero indicates no similarity. Overall, a low Hamming Loss and a high Jaccard Index are positive indicators of the performance of the classification model.

III. Preliminary Results

To evaluate the proposed model, we implemented it on 3954 research projects available on the FRIS portal. The dataset was split into training and testing datasets with proportions of 80% and 20%, respectively. We evaluated the LSTM with different numbers of hidden units, such as 32, 64, 128, and 256. The number of epochs was set to 100 and the batch size to 32. As a result, the LSTM model with 128 hidden units achieved the best performance. Table I shows the performance of the LSTM model on the dataset. As can be seen, the scores of Precision and F1-scores were quite low, which means that the model was not performing well in terms of predicting both positive and negative instances correctly. For instance, with the Micro average method, the Precision, Recall, and F1-scores were 0.29, 0.81, and 0.43, respectively. The Precision of 0.29 indicates that out of all the positive predictions made by the model, only 29% were actually correct. A Recall of 0.81 indicates that the model was able to correctly identify 81% of the actual positive instances. The result of a high recall and low precision indicates that the model is able to identify most of the positive cases (true positives), but it also predicts a high number of false positives. The F1-scores of 0.43, which is the harmonic mean of Precision and Recall, suggests that the model is not performing well on this task.

TABLE I
THE PERFORMANCE OF THE CLASSIFICATION MODEL.

	Precision	Recall	F1-score	Support
Micro avg	0.29	0.81	0.43	1318
Macro avg	0.25	0.70	0.36	1318
Weighted avg	0.32	0.81	0.45	1318
Samples avg	0.31	0.87	0.44	1318

The Hamming Loss and Jaccard Index were 0.09 and 0.30, respectively. The Hamming Loss of 0.09 suggests that on average, 9% of the labels assigned by the model to the instances were incorrect. The Jaccard Index of 0.30 suggests that there was a relatively low overlap between the predicted and true labels. The reason for the low performance of the model is possible that the model made more correct predictions overall, but the predictions were not accurate in terms of individual labels. This could happen if the classes are imbalanced, and the model is biased toward the majority class. It could also occur if the features used to train the model were not representative enough of the true underlying patterns in the data.

To assess the performance of the model on individual classes (discipline codes), we calculated the Precision, Recall, and F1-scores for each class. The results are presented in Table II. As observed, the number of projects (support) containing discipline codes varies significantly, indicating that the test data suffers from an imbalance issue. As a result, the model did not perform well on this dataset. Moreover, we can see that the Precision, Recall, and F1-scores associated with some discipline codes were low or even zero. This outcome indicates that many discipline codes were not predicted accurately by the model.

To address the issue of imbalanced data in the model, we excluded projects that contained low-frequency discipline codes. Specifically, we excluded projects that had discipline codes occurring less than 100 times. This resulted in a total of 3552 projects being available for analysis. The model's performance on this dataset is presented in Table III. As can be seen, all Precision, Recall, F1-scores were better than the previous results. The Hamming Loss and Jaccard Index scores were 8% and 43%, respectively, indicating a slight improvement in the model's performance. By excluding the low-frequency discipline codes, we reduced the impact of

TABLE II
PERFORMANCE OF THE MODEL ON INDIVIDUAL LABELS.

Discipline codes	Precision	Recall	F1-score	Support
0101	0.28	0.51	0.36	37
0102	0.32	0.65	0.43	65
0103	0.31	0.7	0.43	44
0104	0.22	0.97	0.36	58
0105	0.23	0.83	0.36	18
0106	0.46	0.91	0.61	104
0107	0.33	0.8	0.47	30
0201	0.34	0.79	0.47	34
0202	0.4	0.93	0.56	71
0203	0.26	0.93	0.41	45
0204	0.28	0.66	0.39	41
0205	0.27	0.78	0.4	45
0206	0.24	0.64	0.35	22
0207	0.31	0.72	0.43	25
0208	0.1	0.22	0.13	18
0299	0	0	0	14
0301	0.5	0.93	0.65	122
0302	0.29	0.91	0.44	58
0303	0.19	0.73	0.3	33
0304	0.33	0.82	0.47	28
0305	0.16	0.96	0.28	24
0306	0.2	0.8	0.32	45
0399	0	0	0	0
0401	0.42	0.81	0.56	37
0402	0.08	0.5	0.14	4
0499	0	0	0	2
0501	0.29	0.86	0.44	35
0502	0.38	0.9	0.53	40
0503	0.19	0.7	0.3	23
0504	0.18	0.86	0.3	28
0505	0.35	0.93	0.51	27
0506	0.21	0.88	0.33	16
0507	0.31	0.57	0.4	7
0508	0.2	0.85	0.32	13
0599	0.1	0.33	0.15	3
0601	0.29	0.93	0.44	27
0602	0.3	0.88	0.44	25
0603	0.21	0.88	0.33	17
0604	0.43	0.91	0.59	32

TABLE III
THE PERFORMANCE OF THE CLASSIFICATION MODEL.

	Precision	Recall	F1-score	Support
Micro avg	0.39	0.85	0.53	1127
Macro avg	0.39	0.83	0.52	1127
Weighted avg	0.44	0.85	0.56	1127
Samples avg	0.45	0.89	0.56	1127

imbalanced data on the model and were able to improve its performance.

IV. CONCLUSION

In this work, we proposed an approach to predict disciplines in research projects based on an organizational approach. The proposed approach is useful when there are no citation data or high-quality abstracts available. Particularly, in this proposed approach, the projects are represented by researchers' disciplines collected from various resources such as profiles, organizations, projects, co-authors on projects, publications, and co-authors on publications. To predict disciplines related to projects, we applied a deep machine learning algorithm. We implemented the proposed approach on research projects available on the FRIS portal. The preliminary results showed that the model was not performing very well for this particular task. There are several things we can try to improve the performance of the model:

- Collect more data to improve the representation of each class and balance the data if possible.
- Experiment with different hyperparameters such as the number of LSTM layers, the number of nodes in each layer, and the learning rate.
- Using a different type of neural network architecture, such as a convolutional neural network (CNN) or a transformer network.
- Consider preprocessing the data differently, such as using different normalization techniques or feature engineering.
- Explore the possibility of incorporating external data sources or domain knowledge to improve the performance of the model.
- Additionally, it might be helpful to analyze the specific misclassifications the model is making to gain insights into why the model is not performing well. This can help guide future improvements to the model.

REFERENCES

[1] K. Boyack and R. Klavans, "Co-citation analysis, bibliographic coupling, and direct citation: Which citation approach represents the research front most accurately?" *Journal of the American Society for Information Science and Technology*, vol. 61, pp. 2389–2404, 12 2010.

[2] J. Eykens, R. Guns, and T. Engels, "Article level classification of publications in sociology: An experimental assessment of supervised machine learning approaches," 09 2019.

[3] M. Rivest, E. Vignola-Gagné, and É. Archambault, "Article-level classification of scientific publications: A comparison of deep learning, direct citation and bibliographic coupling," *PLOS ONE*, vol. 16, no. 5, p. e0251493, may 2021.

[4] T. Weber, D. Kranzlmüller, M. Fromm, and N. T. de Sousa, "Using supervised learning to classify metadata of research data by field of study," *Quantitative Science Studies*, pp. 1–26, may 2020.

[5] S. Vancauwenbergh and H. Poelmans, "The flemish research discipline classification standard: A practical approach," *KNOWLEDGE ORGANIZATION*, vol. 46, no. 5, pp. 354–363, 2019.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, nov 1997.