TUM Data Innovation Lab

Munich Data Science Institute (MDSI)

Technical University of Munich

&

TUM Chair of Software Engineering for Business Information Systems (sebis)

Final report of project:

NLP and Knowledge Graphs for Research Cluster Prediction and Analysis

Authors	Cavit Cakir, Christian Brand, Raoyuan Zhao, Valentina Izrailevitch, Yaling Shen
Mentor	M.Sc. Tim Schopf
Project Lead	Dr. Ricardo Acevedo Cabra (MDSI)
Supervisor	Prof. Dr. Massimo Fornasier (MDSI)

Feb 2023

Abstract

With a fast growing research community and even faster growing domains of area of research it is important to create Knowledge Graphs allowing for orientation in the domains of various fields of study (FoS). Creating of such Knowledge Graphs requires identifying relevant information such as areas of research in particular institutions, classifying the information to different subjects and creating relevant nodes and corresponding relations in the knowledge graph. In our project we focus on classifying the scientific publications (using the titles and abstracts of the papers) based on concepts of the Field of Study Ontology. We use the FoS Ontology provided by the OpenAlex research. It includes 19 global fields of study and has 6 levels of hierarchy, from more abstract at level 0 to the most specific at level 5. For the supervised approach we use Transformer-based models such as BERT and SciBERT as well as SVM trained on different datasets of labeled publications. The results shows superiority of the SciBERT model (BERT pretrained with scientific texts) over two over models in most of the global (level 0) categories. We continue with the unsupervised approach based on CSO Classifier and expand it to 19 scientific domains. In addition to changing the ontology of the classifier we implement two different embeddings approaches - word2vec model based on the OpenAlex dataset and SPECTER embeddings of the same dataset. The result of both approaches were similar. We present a hybrid model which combines the successful parts of both approaches - it uses the supervised approach for global classification followed with the unsupervised classification on more specific levels. Additional we present an unsupervised hierarchical model which classify the publications based on their similarity to the ontology concepts using embeddings of Transformer models. On the final phase of our project we proceed to the modeling of the research clusters using both text-based Topic Modeling as well as Graph clustering.

Contents

1 Introduction

1.1 Problem definition and goals of the project

When considering the current academic world, publications are an integral part of how research is communicated and shared. But as the academic world has grown and become a global effort, the number of publications has risen accordingly, making literature research an ever more time-consuming task. Our project with the TUM chair for Software Engineering for Business Information Systems (sebis) wants to tackle this problem, by working on automatically labeling publications with topics from the OpenAlex topic ontology, based on their title and abstract.
Our project has 2 main focuses: the primary goal was to implement, adapt and compare existing methods of topic labeling for publications from all fields of study. Our secondary goal was to analyze the publication topics of an institution and their researchers , and what insights a graph cluster representation might offer, e.g., concerning their research areas in comparison to text-based topic modeling.

2 Outline

This section contains a brief overview of the following chapters outlining the various models we examined.
After sourcing, preprocessing and downsampling our data (see chapters 3 and 4), we implemented and adapted existing models for an unsupervised and supervised approach (see chapters 5 and 6). While the unsupervised approach had the goal of predicting all potential topic labels, the supervised approach focuses on predicting top level concepts (e.g., medicine, art, computer science, etc.).
Beyond just improving and expanding existing models, we created a hybrid model combining the top-level supervised prediction with the unsupervised approach (see chapter 7).
Further we implemented a new model idea, which utilizes topic embeddings and hierarchies, correspondingly named embedding-based hierarchical selection model (see chapter 6).
In Chapter 8, a graph clustering approach is compared to a topic modeling approach on a publication graph of the TUM.

3 Data Sets

3.1 OpenAlex and Ontology Dataset

OpenAlexs dataset [8] was the foundation for the main dataset we use in our project. In particular we used the subset of it with scientific works. The dataset was downloaded through the OpenAlex API from the OpenAlex website, which has at the moment 240M works and adds approximately 50,000 newly indexed works daily [7]. The data was downloaded in snapshots, the latest was from the 16th of September, 2022. The dataset includes 7,154,102 works entity. Each work entity provides information about the title of the work,

the text of the abstract, date of publication, labels, which corresponds to the concepts from the OpenAlex ontology. Due to the limitations in the computational power and to speed up the exploration we downsample the original dataset to approximately 10% of its size, preserving the distribution of the concepts among the papers.
The OpenAlex Ontology is a hierarchical representation of concepts, including relationship (hyponym - hypernym) between them. Concepts are abstract topics, related to the content of the papers. The current version of the OpenAlex Ontology includes 5 levels of concepts, starting with 19 most global at the level 0 (such as "medicine" or "computer science") and ending with more precise and rare concepts at level 5 (such as "cultural analytics" or "software analytics").

3.2 Silver and Gold Dataset from SciRepEval

The silver and gold datasets are used for the supervised level zero multi-label classification task and are part of SciRepEval, a novel collection of datasets and a new evaluation benchmark for document-level embeddings in the scientific domain released by Amanpreet Singh et al. [10]. There are 25 tasks including classification, regression, ranking, and search format. The silver dataset is a new large-scale Field-of-Study (FOS) set with labels based on publication venue, where each paper within a venue is assumed to belong to a narrow set of fields and the FoS labels are manually assigned to the publication venues. The gold dataset, whose FoS labels are manually labeled into at most three, is used only for testing.
The original silver and gold datasets have 23 FoS labels while the OpenAlex corpus has only 19 corresponding categories at level 0. To perform a more precise comparison of the classification results, we remove the samples that have been assigned with labels not included in the 19 labels in the OpenAlex corpus out of the silver and gold datasets, and re-assign their label ids correspondingly to the OpenAlex id system. After preprocessing, there are in total of 484,605 publications in the silver train set and 60,991 in the test set. The number of samples included in the gold dataset has decreased from 472 to 392.

3.3 TUM Publication Graph Data

TUM publication graph data is used for generating the publication graph of TUM, which can help to extract the popular topics and identify connection between them by means of clustering. There are 83,580 publications included. The original publication data consists of DOI, article title, abstract, author information, institution, and year of publication.
Each publication is additionally tagged with topic labels originating from the OpenAlex ontology, thus providing the concept information. The tagging is performed with the help of the OpenAlex-concept-tagging V2 model, which provides the results of the classification based on the input consisting of the title and abstract of the article.
Each publication in the original data is then treated as a node, as well as each label in the ontology. There are two kinds of relationships in the graph data: the correspondence between the publication and the label, and the ontology relationship within the labels. These relationships are directional, so the whole graph is a directed graph. There are 157,832 nodes and 2,791,452 edges in the graph data.

4 Downsampling

Originally downloaded OpenAlex data set is too large, and given the computational power of the existing GPU, it would take too long to train model on the whole data set. For that reason we decided to reduce the size of the data set to ten percent of the original size. To keep the bias of the classification model minimal and maintain the data representing of the original dataset but still balanced we preserve the data distribution (identical to the original one). The downsampling is therefore necessary.
In order to achieve these two goals, the downsampling algorithm is designed in the following way:
Firstly, we take the best-fit label in the concepts as the only label for each sample, then the data is grouped according to the level of the labels and the types of labels in each group are counted, thus calculating the mean of amounts of labels in each level. The gap between original samples for each label and the expected are obtained by subtracting one-tenth of the mean from the number of samples for each label. To make the data more balanced, only samples with a number of labels above one-tenth of the mean and with a gap greater than 1 are discarded, and the ratio of discards per label is calculated as:

$$discard\_pro = \frac{\frac{gap[i][j]}{gap\_sum[i]} * 0.1 * discard\_amount[i]}{amount[i][j]}$$

where i means which level and j indicates the specific label in current level. discard_amount[i] is the amount of samples whose label belongs to level i that should be discarded. amount[i][j] is the amount of samples for current label. gap_sum[i] is the sum of all positive gaps in the level i. Then, random sampling is taken in one minus this proportion to complete the downsampling.
After applying this algorithm, the amount of samples in original dataset change from 71,541,027 to 7,154,541 while the number of label stay the same, i.e. 63,245, which means none of labels were lost in the process of the downsampling.

Level	Amount(original)	Amount(downsampled)	Variance(original)	Variance(downsampled)
0	15721137	1572114	2033243709368.83	779115722.34
1	2589180	258919	242754007.68	123197.94
2	33951811	3395241	18041321.63	18754.88
3	141545646	1415533	3754366.51	2687.82
4	4108870	411047	899557.42	567.64
5	1015383	101687	466091.91	268.66

Table 1: Amount and variance of the topic labels before and after downsampling

The visual comparison between original dataset and downsampled dataset as a whole is shown in Figure 1, comparison of the two dataset across the labels of level 0 could be seen at Figure 2, other levels' comparisons are attached in appendix 11. During the downsampling process, publication date was not considered as reference for discarding, but as can be seen in 3, publication date also becomes more balanced.

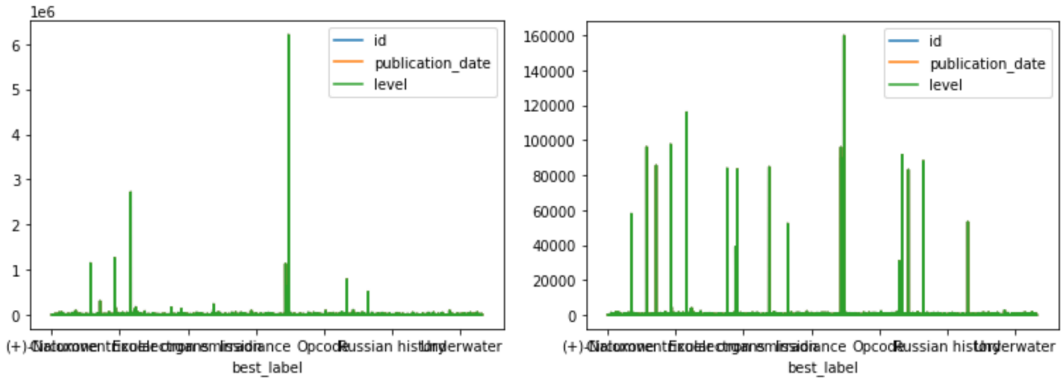


Figure 1: Label Distribution Comparison Between Original Dataset And Downsampled Dataset

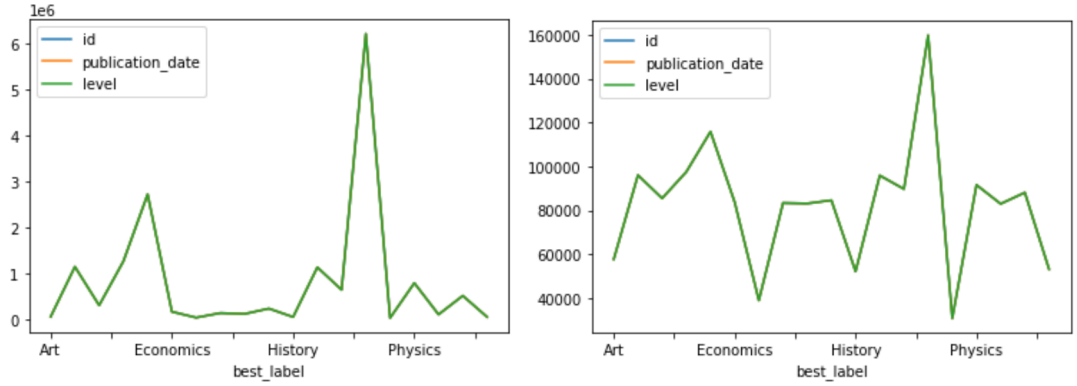


Figure 2: Level 0 Label Distribution Comparison between Original Dataset and Downsampled Dataset

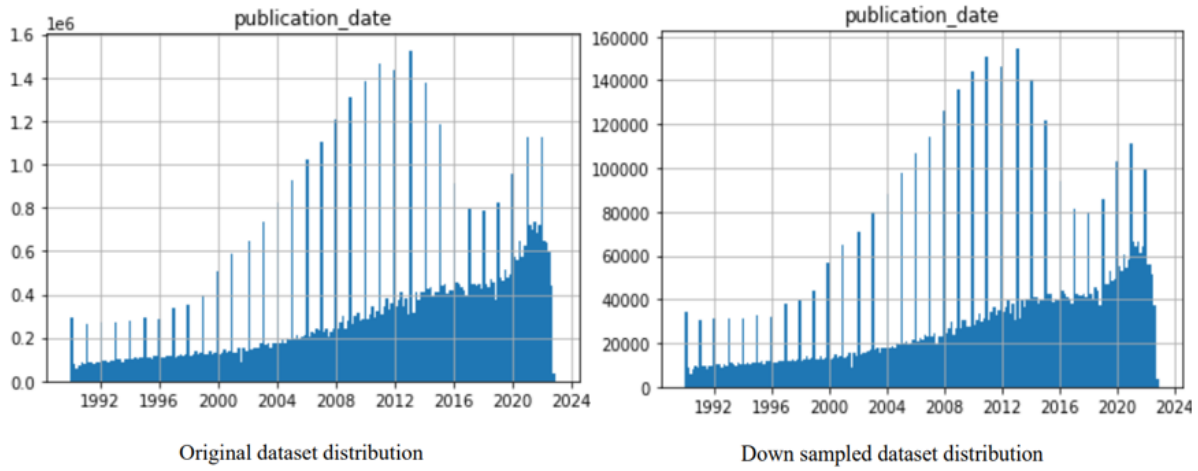


Figure 3: Publication Date Distribution Comparison between Original Data set and Downsampled Data set

There is some data in the data set with label scores below 0.3, which indicates that they have no credibility and are likely to be incorrect data, so the downsampled data needs to be cleaned up. The proportion of such data in the downsampled data set is 0.1967% and can be discarded. The length of the cleaned data set is 7,116,099.
In addition, as the data set after the downsampling is subsequently sliced into the training set, validation set and test set needed to train the model, it is necessary to ensure that the variety of samples in each set is the same. To elaborate more, it is impossible to get the trained model to give classification results for this type of article if a labeled sample is only present in the test set and there is no data with this label in the training set.
Data with label of higher level (e.g., 4 or 5) may only exist in one or two samples, and after the statistics have been calculated, it is found that there are 35,114 samples whose label appear less than ten times (ten is chosen as the filtering threshold because the approximate ratio of training set, validation set and test set is eight to one to one when splitting the set), accounting for 0.4934% of the entire downsampled dataset so they can be discarded directly. The length of the final downsampled dataset is 7,080,985. After splitting, the train set length is 5,664,672, validation set length is 708,247 and the test set length is 708,066.

5 Supervised Approach for Level Zero Concepts

We consider the level-zero ontology label prediction problem as a multi-label classification task, where a single publication can be assigned to one or several level-zero Field-of-Study (FoS) concepts. In this section, a machine learning model, SVM, and two pre-trained deep learning language models, BERT, and SciBERT are implemented to train the classifier. To compare the model performance as well as the ability of generalization, we train each model on two different data sets and test them on three data sets, including a manually labeled one.

5.1 Machine Learning Model: SVM

Since the linear support vector machine model is applied by [10] for the classification downstream task, we also train a multi-label SVM model as our baseline. We simply implemented `scikit-learn` built-in linear SVM function with `MultiOutputClassifier` class and a random state equal to 42 for all the experiments.

5.2 Transformer Models: BERT and SciBERT

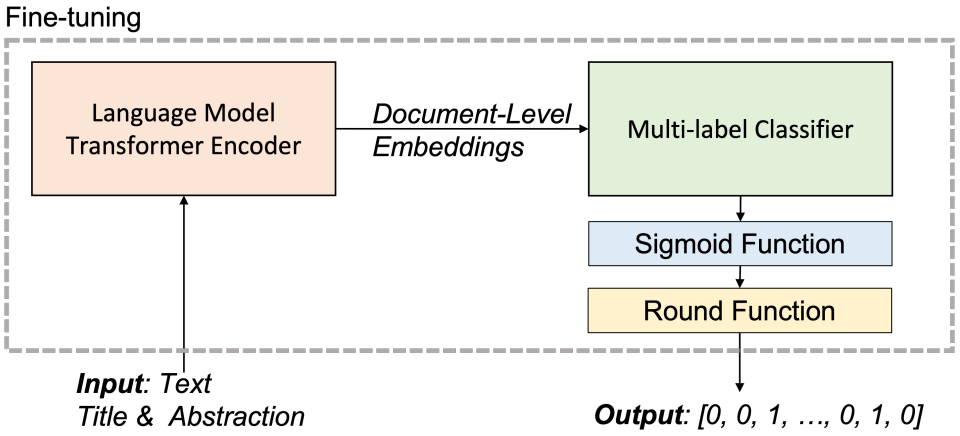


Figure 4: Supervised Multi-label Classification Model Architecture

Deep learning model architecture is shown in Figure 4. It consists of two parts, a Transformer encoder, and a multi-label classifier. A Transformer encoder is implemented to get the document-level embeddings for the input text, including the title and abstract of the publications. Based on the embeddings, the classifier will generate the probabilities of the publication being assigned to each label category. We then use a sigmoid function as well as a round function to get binary results.

We first use the famous pre-trained language representation model **B**idirectional **E**ncoder **R**epresentations from **T**ransformer (BERT) as our baseline transformer model [5]. Since we are dealing with scientific publications, we also implement SciBERT, a language model that is pre-trained on a large multi-domain corpus of scientific publications based on BERT for our multi-label classification task[2].

As for the experimental setup, we simply adopt the built-in functions huggingface transformers package offered in `AutoModelForSequenceClassification.from_pretrained()`, with the problem type of 'multi_label_classification'. We use 'bert-base-uncased' to load the BERT model and 'allenai/scibert_scivocab_uncased' for SciBERT. For all experiments, we train 3 epochs with the max length for encoder padding being 256, batch size of 32, and learning rate 1e-5.

5.3 Experimental Results

The experimental results on the silver dataset and the OpenAlex corpus are shown in Table 2 and 3, respectively. We can see that usually SciBERT model can achieve the best performance, but it is not guaranteed for every category.

	Precision			Recall			F1			#
	SVM	BERT	SciBERT	SVM	BERT	SciBERT	SVM	BERT	SciBERT	
Computer science	82.77	89.71	**93.71**	61.15	**87.31**	85.26	70.34	88.40	**89.28**	4525
Medicine	84.07	95.14	**95.53**	79.36	**96.47**	95.18	81.64	94.90	**95.36**	8312
Chemistry	79.51	**93.87**	93.23	62.10	89.60	**93.02**	69.73	**91.69**	91.13	5200
Biology	78.43	93.48	**95.46**	74.43	93.45	**93.59**	76.53	93.46	**94.52**	6916
Materials science	79.13	88.72	**89.32**	63.76	87.12	**88.46**	70.62	87.91	**88.89**	2997
Psychology	87.73	95.24	**95.67**	78.96	94.71	**94.87**	83.12	94.98	**95.27**	3722
Business	83.87	**92.16**	89.13	63.06	83.93	**86.28**	72.00	**87.85**	87.68	1064
Mathematics	85.23	**92.87**	91.48	72.19	88.73	**92.58**	78.17	90.76	**92.03**	4420
Political science	79.99	**91.70**	86.75	54.52	85.16	**86.77**	64.84	**88.31**	86.76	3153
Environmental science	84.62	94.81	**95.48**	76.21	93.89	**94.04**	80.19	94.35	**94.75**	6713
Physics	86.29	93.18	**95.87**	73.87	**93.22**	92.12	79.60	93.20	**93.96**	8912
Engineering	74.87	**88.53**	82.33	35.51	78.66	**86.31**	48.17	**93.31**	84.27	4523
Sociology	**89.88**	87.24	89.18	42.60	72.36	**67.42**	57.80	**79.11**	76.79	709
Geography	82.78	91.63	**92.89**	78.54	**95.20**	94.47	80.60	93.38	**93.67**	1230
Economics	79.33	83.89	**87.05**	58.03	**87.81**	83.52	67.03	**85.81**	85.25	2864
Geology	81.39	**93.34**	92.89	67.94	**93.15**	92.96	74.06	**93.25**	92.93	1519
History	82.71	**94.63**	93.81	82.02	**94.70**	94.00	82.36	**94.67**	93.91	5401
Art	85.81	**94.66**	92.93	60.47	**86.51**	79.53	70.94	**90.40**	85.71	430
Philosophy	**86.14**	83.73	77.97	37.87	80.63	**84.33**	52.61	**82.15**	81.02	919
Micro Avg.	82.72	92.53	**92.60**	68.67	90.56	**91.22**	75.05	91.54	**91.91**	73529
Macro Avg.	82.87	**91.50**	91.09	64.36	88.46	**88.67**	71.60	**89.89**	89.75	73529
Weighted Avg.	82.50	92.51	**92.66**	68.67	90.56	**91.22**	74.40	91.49	**91.90**	73529
Samples Avg.	70.58	90.54	**90.97**	69.85	90.55	**91.06**	69.10	90.26	**90.77**	73529

Table 2: The models are trained on the silver train set and tested on the silver test set. '#' saves the number of actual occurrences of each label in the silver test set. The results in bold type give the highest score for each label category and each indicator.

Table 4 shows the overall accuracy and F1 score for each model. The three above mentioned models are trained on the OpenAlex corpus and silver data set, respectively. And the six models are then tested on three different test data sets, including the OpenAlex test set, the silver evaluation set, and the gold data set, to compare the generalization ability of each model. It turns out that the model performance decreases drastically when transferred to another test data set. However, SciBERT models still achieve the overall best performance. What is interesting enough to draw attention is that the SVM models have comparable performance when tested across different data sets. The detailed cross-data set classification scores for each ontology label are shown in Table 15 and 16 in Appendix.

We also test all of our six trained models (one machine learning model and two deep learning transformer models trained on two different train sets, respectively) on the gold dataset because the samples in it are manually labeled, and the experimental results are recorded in Table 14 in Appendix. As also indicated in Table 4, models trained on the OpenAlex corpus outperform those trained on the Silver dataset, which may attribute to the difference in the number of data samples they contain as the OpenAlex corpus has far more data than the Silver.

	Precision			Recall			F1			#
	SVM	BERT	SciBERT	SVM	BERT	SciBERT	SVM	BERT	SciBERT	
Computer science	84.67	88.85	**90.72**	73.50	**86.38**	84.66	78.69	**87.60**	87.59	124998
Medicine	87.00	91.54	**91.75**	84.62	91.25	**92.52**	85.79	91.44	**92.14**	114774
Chemistry	82.40	81.59	**88.58**	76.18	**91.32**	85.30	79.17	86.18	**86.91**	84276
Biology	83.76	**89.73**	88.85	81.52	87.73	**91.00**	82.63	88.71	**89.91**	82332
Materials science	83.89	**89.39**	87.27	76.25	86.18	**90.15**	79.89	87.75	**88.69**	62010
Psychology	82.00	**87.07**	86.91	71.02	83.05	**83.78**	76.12	85.01	**85.32**	60189
Business	80.54	85.73	**87.79**	67.58	**85.55**	82.97	73.49	**85.64**	85.31	50482
Mathematics	86.21	89.92	**90.55**	65.66	81.11	**81.56**	74.54	85.31	**85.82**	48721
Political science	77.59	83.13	**83.59**	58.79	**79.95**	77.83	67.03	**81.51**	80.61	38785
Environmental science	81.26	80.88	**86.58**	63.00	**86.07**	82.87	70.98	83.40	**84.68**	37839
Physics	85.69	90.91	**91.00**	70.30	83.71	**84.67**	77.24	87.16	**87.72**	37638
Engineering	74.12	**82.77**	80.40	46.80	64.43	**66.97**	57.40	72.46	**73.07**	34567
Sociology	75.40	**81.57**	80.43	55.19	73.39	**73.95**	63.73	**77.26**	77.05	34547
Geography	76.11	75.71	**81.95**	44.68	**77.47**	69.15	56.31	**76.58**	75.01	32528
Economics	85.01	**89.98**	87.97	77.96	86.32	**88.57**	81.33	88.11	**88.27**	28726
Geology	89.43	**91.87**	89.65	73.65	83.34	**85.83**	80.78	87.40	**87.70**	24407
History	72.67	77.86	**78.28**	46.14	**71.39**	67.53	56.45	**74.48**	72.51	20508
Art	75.36	78.91	**79.71**	49.63	**75.46**	69.19	59.85	**77.15**	74.08	16283
Philosophy	80.09	75.45	**80.82**	49.69	**77.55**	69.52	61.32	**76.49**	74.75	10482
Micro Avg.	82.96	86.48	**87.79**	70.17	**84.33**	83.70	76.03	85.39	**85.70**	944092
Macro Avg.	81.24	84.89	**85.94**	64.85	**81.67**	80.42	71.72	**83.14**	83.01	944092
Weighted Avg.	82.58	86.55	**87.67**	70.17	**84.33**	83.70	75.55	85.33	**85.57**	944092
Samples Avg.	78.86	**88.98**	89.52	75.72	**87.67**	87.27	75.07	86.43	**86.54**	944092

Table 3: The models are trained on the OpenAlex train set and tested on the OpenAlex test set. '#' corresponds to the number of actual occurrences of each label in the OpenAlex test set. The results in bold type give the highest score for each label category and each average indicator.

Test Set	**Train Set**	**OpenAlex**			**Silver**			#
	Method	**Acc.**	**F1**		**Acc.**	**F1**		
			Mirco Avg.	Macro. Avg		Micro Avg.	Macro Avg.	
OpenAlex	SVM	57.79	76.03	71.72	34.53	**51.53**	40.40	944092
	BERT	70.31	85.39	**83.14**	35.24	51.16	**42.98**	
	SciBERT	**71.08**	**85.70**	83.01	**38.81**	49.79	42.73	
Silver	SVM	36.82	53.58	46.11	59.86	75.05	71.60	73529
	BERT	**37.65**	52.63	45.46	87.96	91.54	**89.89**	
	SciBERT	33.33	**54.46**	**47.08**	**88.68**	**91.91**	89.75	
Gold	SVM	41.84	60.50	57.37	34.18	**51.12**	**46.77**	463
	BERT	45.92	62.18	59.23	36.22	50.39	46.55	
	SciBERT	**46.17**	**63.78**	**61.28**	**36.73**	49.17	45.92	

Table 4: The models are trained on the OpenAlex train set and the silver train set, respectively. The experimental results are tested on the OpenAlex test set, the silver test set, and the gold dataset. '#' is the total number of actual occurrences of all the labels in each test set. The results in bold type give the highest score for each indicator among different models trained and tested on the same train and test set.

6 Unsupervised Approach

6.1 Introduction to FoS Classifier

Field-of-Study (FoS) Classifier is an expansion of the CSO Classifier[9] to a larger range of scientific areas. CSO Classifier focuses on the computer science papers and uses unsupervised approach to classification based on similarity distance between the words from the title and abstract of an article and the topics from the Computer Science Ontology. We combined this approach with the FoS Ontology of the OpenAlex and thus broaden the usage of the Classifier to 18 more scientific areas. In addition to changing the Ontology structure, we update the embeddings used in the model for classification process - we use a word2wec approach as well as the embeddings from the SPECTER language model. As well as CSO Classifier, FoS Classifier provides an option of a faster classification based on precomputed similarity between the vocabulary of the OpenAlex Data set and the concepts of the FoS Ontology. The vocabulary was created from the train part of the OpenAlex downsampled dataset.

6.1.1 Word2Vec embedding

We applied the word2vec approach [1] to a collection of texts from OpenAlex Corpus in order to generate word-embeddings. Word2vec is a powerful method for representing words as vectors in a high-dimensional space, and it has been shown to be effective for a variety of NLP tasks. The word2vec approach uses a neural network architecture to learn the vector representations of words from large amounts of text data. During training, the network learns the relationships between words based on their co-occurrence patterns in the text data.
The text data was pre-processed to ensure that the word embeddings generated were accurate and meaningful. The pre-processing steps included lowercasing words, filtering non-English words, removing all punctuation, and non-alphabetic characters. We also removed empty spaces that occurred more than once, and before or after a sentence, replaced spaces with underscores in n-grams matching the FoS topic labels, and for frequent bigrams and trigrams. For example, 'computer science' became 'computer_science'. The n-gram parameters we used were min_count=5 and threshold=15, while the word2vec parameters were size=128, window=10, and min_count=10.

6.1.2 SPECTER embedding

SPECTER [4] is one of the BERT-based models trained on a scientific papers with special emphasis on closer relatedness of the articles through the citations. SPECTER has outperformed other models on classification scientific papers tasks on SciDocs benchmark. Based on these result we expect a certain improvement over a usual word2vec model in classification with FoS Classifier.
We embedded all the entries from the pre-computed vocabulary with SPECTER. As the following step we computed the cosine similarity between each entry and all other entries in the vocabulary, thus creating for each entry list of top 10 similar entries. In the next step we compute the Levenshtein distance between each entry in such list and every concept in the ontology that starts with the same string of four letters as the currently

processed entry. The similarity threshold was set at 0.94. All entries with this score or above are considered as relevant to the concept and the combination entry - concept - similarity score would be saved for the faster classification track.

6.2 Unsupervised Hierarchical Model

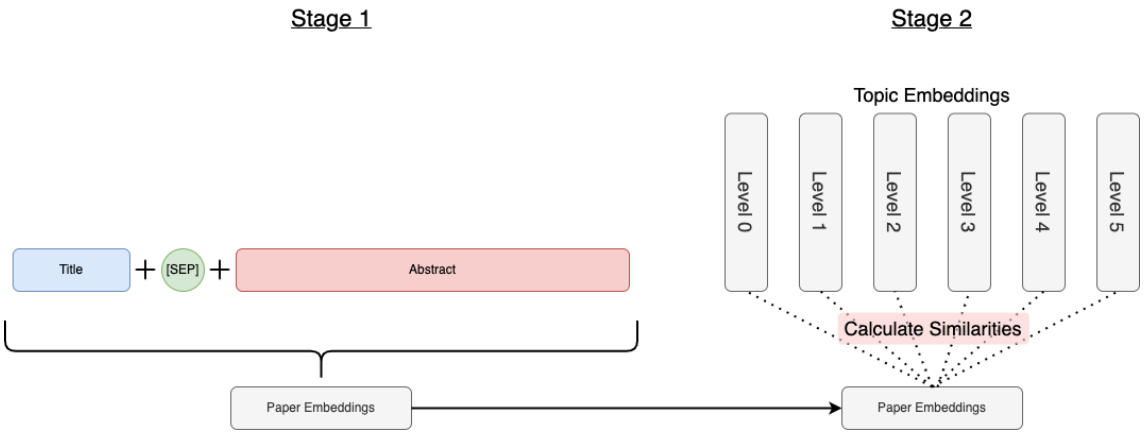


Figure 5: Unsupervised Hierarchical Model Architecture

This approach consists of two stages aimed at categorizing research papers using unsupervised methods and Transformers-based embedding models.
The first stage involves obtaining embeddings for both papers and topics using SPECTER and SBERT embedding models. These embeddings capture the features, information, and meaning of each paper and topic, and will be further described in a subsequent subsection. The process of obtaining topic embeddings is straightforward: each topic is fed individually to the embedding models, and the CLS token from the last hidden layer of the embedder models is saved. For paper embeddings, the title and abstract are combined, separated by a [SEP] token, and then fed to the model, with the CLS token from the last hidden layer being saved.
The second stage compares the similarity between the paper and topic embeddings using Cosine Similarity. The results of this comparison are then utilized in different ways. In the first approach, each paper's embedding is compared to each level's topics, and the top k most similar topics are selected and labeled at each level. In the second approach, knowledge from an ontology is incorporated, creating a graph that connects topics to subtopics. The process starts with level 0, where the top k most similar topics are selected and labeled, and their subtopics are collected. The similarity between these collected subtopics is then calculated, and the process is repeated for all levels.
In conclusion, this approach provides a comprehensive and efficient method for categorizing research papers, making use of state-of-the-art Transformers-based language models in an unsupervised manner. The combination of unsupervised methods and Transformers-based models results in a powerful solution that can effectively capture the meaning and information contained in research papers and categorize them accurately.

6.3 Comparison of models

To understand the effectiveness of our models, we evaluated and compared them to the OpenAlex topics on test data, considering Recall, Precision and F1 Score.
To begin, we want to identify the best model instance for our different unsupervised models. For example, the FoS Classifier can predict topics based on their semantic context, their syntactic context or the union of both. Comparatively for the Hierarchical model we can consider different k numbers of most similar topics.
In table 5, we can see the best performing models for FoS Word2Vec, FoS SPECTER and the Unsupervised Hierarchical Model (See an overview of all models in appendix 17). These all do not have very high overall metrics, so in the following we would like to further investigate the differences between the models and explore ways to potentially improve these results.

	Precision	**Recall**	**F1-Score**
FoS - SPECTER	29.83%	31.31%	30.55%
FoS - Word2Vec	32.83%	29.70%	31.19%
Hierarchical Selection	23.24%	11.31%	15.22%

Table 5: Table shows the best performing model instances for each unsupervised model type

6.3.1 FoS Evaluation

One analysis that offers more insights is splitting the evaluation of our prediction by topic level. For example, we can see in table 6, that the FoS SPECTER model is better at predicting levels 2 and 3, while it does not predict level 0 and level 1 labels well. The low scores for level 0 and 1 can partly be explained by the model not adding general concepts as frequently as necessary, only about every 1 out of 10 times.

Level	**With Tag (Pred.)**	**With Tag (Target)**	**Recall**	**Precision**	**F1 Score**
0	11.48%	98.90%	3.95%	5.25%	4.51%
1	30.17%	93.96%	5.00%	9.20%	6.48%
2	96.78%	93.05%	45.75%	30.47%	36.58%
3	79.05%	67.39%	41.53%	28.18%	33.58%
4	35.07%	25.08%	29.15%	25.85%	27.40%
5	14.14%	7.76%	15.08%	15.43%	15.25%

Table 6: Table shows the prediction metrics on each level for the best performing FoS Classifier with SPECTER embedding compared to the OpenAlex corpus on test data

If we inspect other FoS models (e.g., Word2Vec embedding (18), we observe similar performance on different labels. This difficulty predicting more general concepts therefore seems to be embedding-unrelated but rather specific to the FoS model itself.

6.3.2 Hierarchical Model Evaluation

The hierarchical model has a different approach to the FoS Classifier and this also can be observed in the level metric breakdown(7).

Level	With Tag (Pred.)	With Tag (Target)	Recall	Precision	F1 Score
0	100.00%	99.04%	86.89%	71.07%	78.19%
1	99.49%	94.32%	48.82%	8.56%	14.56%
2	82.35%	93.46%	4.06%	5.78%	4.77%
3	37.81%	67.89%	2.77%	3.75%	3.18%
4	14.32%	25.32%	2.26%	2.57%	2.40%
5	0.00%	7.83%	-	-	-

Table 7: Table shows the prediction metrics on each level for the best performing hierarchical selection model compared to the OpenAlex corpus on test data

We can observe, that this model is better at predicting the more general concepts (level 0 and 1), but instead is not able to predict the more specific concepts (level 2 or more). This also is understandable, as for the more general concepts the hierarchy only allows few options to select from, while for specific concepts the potential topics explode in number. More iteration and fine-tuning on this concept would be necessary to bridge this difficulty and increase predictions in higher levels.

7 Hybrid Model

As seen in the previous chapters, only utilizing unsupervised techniques to predict topics at all levels does not yet reach very reliable levels, especially for level 0 and level 1 label predictions.
As this issue appears across multiple instances and embeddings of the FoS classifer, adapting the model could be a potential solution. In this chapter we propose a hybrid model, which combines the unsupervised FoS classifier with the supervised models, that perform well for predicting level 0 labels.
Therefore we combine the prediction of the best supervised approach (SciBERT) with the unsupervised FoS Classifier (Word2Vec), while utilizing the tree structure of our ontology.

7.1 Model Concept

The hybrid model concept follows two prediction steps. The first step predicts level 0 topics, using the supervised SciBERT model. Then for each predicted level 0 topic, an FoS Classifier (reduced to child topics of the level 0 concept) predicts the topics of the other levels. This results in 19 field-specialized ontologies and FoS Classifiers, which are used depending on the output of the SciBERT model prediction.
The underlying idea for this approach, is to ensure a good prediction for level 0, while then utilizing the ontology structure to guide the field-specific FoS classifiers to better results.

7.2 Evaluation and Comparison

When evaluating the hybrid model, see table 8, we notice a slight increase in our metrics, but the results still do not reach great levels (See table 19 for evaluation of different hybrid models).

	Precision	**Recall**	**F1-Score**
Hybrid Model	36.67%	43.42%	39.76%
FoS - SPECTER	29.83%	31.31%	30.55%
FoS - Word2Vec	32.83%	29.70%	31.19%
Hierarchical Selection	23.24%	11.31%	15.22%

Table 8: Table shows the best performing model instances for each unsupervised model type and the hybrid model

Further when analyzing the level breakdown (see table 9), we see almost no increase in levels other than level 0, compared to the regular FoS classifier.

Level	**With Tag (Pred.)**	**With Tag (Target)**	**Recall**	**Precision**	**F1 Score**
0	100.00%	99.04%	84.96%	83.49%	84.22%
1	30.28%	94.32%	5.00%	9.22%	6.48%
2	96.98%	93.46%	45.79%	30.49%	36.61%
3	79.48%	67.89%	41.60%	28.22%	33.63%
4	35.30%	25.32%	29.23%	25.92%	27.47%
5	14.25%	7.83%	15.11%	15.46%	15.28%

Table 9: Table shows the prediction metrics on each level for the best performing Hybrid Model with Word2Vec embedding compared to the OpenAlex corpus on test data

Therefore the FoS classification is not disturbed by the supervised prediction of level 0, but is not increased significantly either. But comparatively the hybrid model is the current best performing model for predicting topics on all levels. With further fine-tuning and concept iteration (e.g. also predicting level 1 through a supervised model) this approach could prove successful.

8 Graph Clustering vs. Topic Modeling

There are two main methods to obtain topic groups from publication data. One is graph clustering, the other is topic modeling.

8.1 Graph Clustering

For graph clustering, this kind task is called Community Detection. Neo4j was used to run this task. There are five main algorithems to do that: Louvain, Label Propagation, Weakly Connected Components, Triangle Count and Local Clustering Coefficient

8.1.1 Louvain

Louvain algorithm is a modularity-based community discovery algorithm. The basic idea is that the nodes in the network try to traverse the community labels of all their neighbours and select the community label that maximises the modularity increment. After maximising the modularity, each community is seen as a new node and the process is repeated until the modularity no longer increases [3].

8.1.2 Label Propagation

The LPA algorithm starts with each node initialised using a unique community id label. These tags are propagated through the network. At each iteration of propagation, each node updates its label to the label to which the maximum number of its neighbours belong. When each node has a majority of the labels of its neighbours, the LPA reaches convergence and the algorithm is complete [11].

8.1.3 Weakly Connected Components

The Weakly Connected Component (WCC) algorithm finds the set of connected nodes. Two nodes are connected if there is a path between them. The set of all nodes that are connected to each other forms a component.

Algorithm	Community/Component Amount	Max Community	Minimum	Mean
Louvain	814	524	1	193.9
LPA	1,268	79,552	1	124.47
WCC	525	157,308	1	300.63

Table 10: Summary of Three Community Detection Algorithm

8.1.4 Triangle Count and Local Clustering Coefficient

The triangle count algorithm and local clustering coefficient can only work on undirected graph, triangle count algorithm counts the number of triangles for each node in the graph which can be used to detect the degree of cohesion of a community, while it can also be used in local clustering coefficient algorithm. As the result, there are 1,055,163 global triangles. A more informative result is the label node with the most triangles, which can be used as a basis for roughly determining the most popular research topics for TUM. The Table 11 shows the top twenty most triangulated nodes.

Node	Triangle Amount	Node	Triangle Amount
Pathology	14,570	Mathematics	37,361
Statistics	14,920	Organic Chemistry	39,649
Composite Material	14,928	Internal Medicine	40,449
Mathematical Analysis	15,140	Chemistry	41,553
Ecology	15,236	Computer Science	42,827
Optics	20,597	Biochemistry	44,326
Artificial Intelligence	26,962	Medicine	48,497
Genetics	27,079	Physics	50,856
Gene	28,941	Biology	53,937
Engineering	29,090	Quantum Mechanics	58,228

Table 11: Top 20 Most Trianglated Nodes and Corresponding Amount

The clustering coefficient (lcc) is a property of the nodes in a network. It indicates the degree of connectivity of the nodes' neighbourhoods. If the neighbourhood is fully connected, the clustering coefficient is one. A value close to zero indicates that there is hardly any connectivity in the neighbourhood. So the lcc's result have a similar effect to the triangular counts'. The results of this algorithm include numerous nodes with a coefficient of 1. Table 12 shows the names of 40 randomly selected nodes with a coefficient of 1.

Complex Conjugate	Nlp	Web Applications	Triangle Amount
Vapor Phase	Machine-Learning	Haptics	Autonomous Robots
Evolution Equation	Bayesian Method	Financial Markets	Carbonatite
Deflection Angle	Ontologies	Risk Process	Biomedical Imaging
Mathematical Problem	Research Policy	Hexapod	Robot Controls

Table 12: Part of Nodes Whose Local Cluster Coefficient is 1

8.2 Topic Modeling

Comparing to graph clustering, topic modelling uses only the original publication data, which requires only the title and abstarct of the publication, without requiring its classification label.

8.2.1 BertTopic

BertTopic is a collection of transformer and an improved version of TF-IDF to calculate scores for topic modelling tools [6]. Provide concatenated titles and abstracts as input, delievers 1,194 topics as the result. The Figure 6 shows the results of a hierarchical analysis of the fifty most frequent topics.

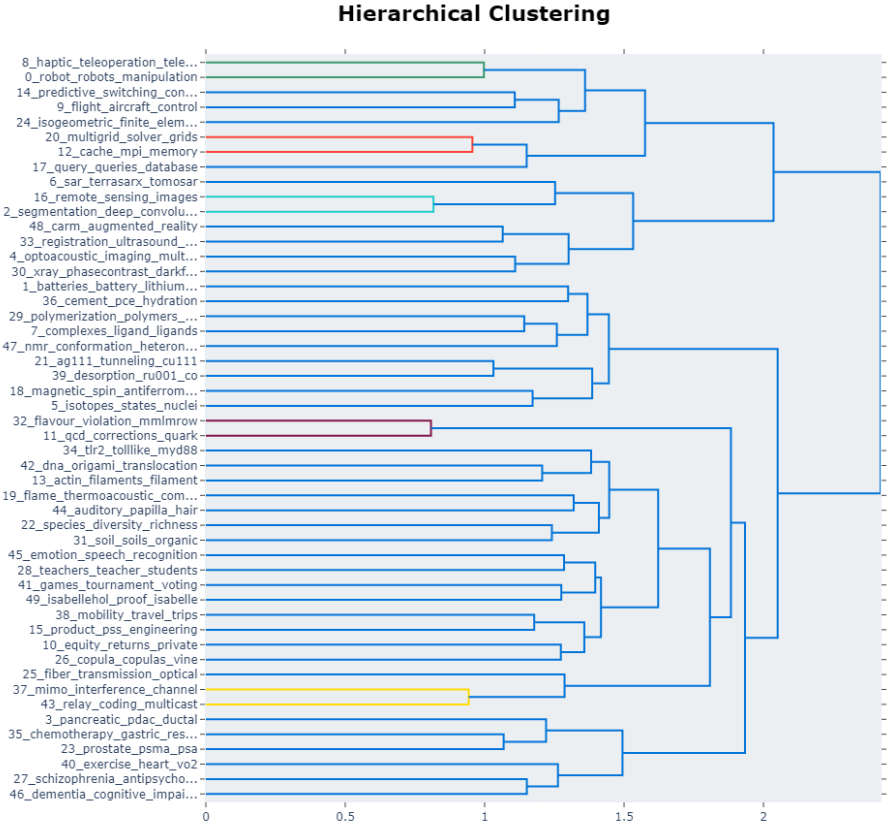


Figure 6: Hierarchical Clustering Result of TUM Publication Data

8.2.2 Top2Vec

Top2Vec is an algorithm for modelling the topic of a given document [1]. In this algorithm, the 'universal-sentence-encoder' is chosen as the embedding model, and the output of the model generates a total of five topics. And the topic shown in 7 can be inferred that it is statistics or machine learning realted topic.

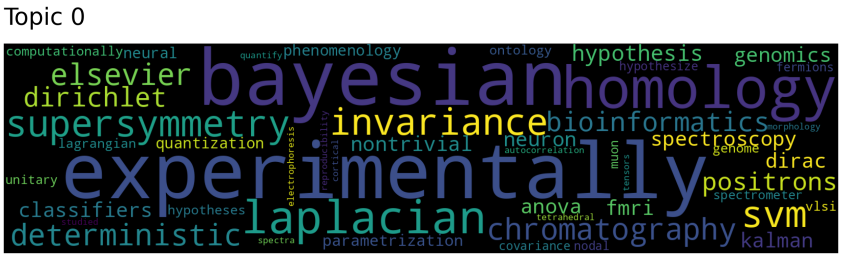


Figure 7: Word Cloud Graph for Topic 0

8.2.3 LDA

The LPA (Parallelized Latent Dirichlet Allocation) is a model for topic analysis of documents integrated in gensim, which uses a Doc2bag model for embedding rather than a

contextual embedding model. The number of topics in this algorithm can be specified, and we have chosen a number twenty that is similar to the number of level zero tags nineteen in our previous study. The final analysis yielded the five most important words for these twenty topics can be seen in Table 13 in Appendix.

8.3 Comparison

Among the five methods of graph clustering, Louvain, LPA and WCC can not only divide the community but also specify the community to which each publication node belongs, while triangle count and local clustering coefficients are more like extracting the keywords of the topic and do not specify the community to which the publication node belongs. We therefore chose the first three methods to calculate the c-TF-IDF scores of keywords for comparison with the topic modelling.

From Table 10, we can see that there are some differences in the communities detected by these three methods, with the LPA algorithm having the largest number of communities and the WCC algorithm having the smallest number of communities. The WCC algorithm has too many nodes in the largest community, which means that a topic most likely contains most of the article nodes, and the LPA algorithm has too many communities, while the maximum community nodes are also as high as 79,552, indicating that the distribution of article nodes has a large variance, which is not desirable. Compared to these two algorithms, Louvain has a smaller number of communities and the maximum number of communities is similar to the cluster size mean, as we chose it for comparison with the topic modelling method. For the topic modelling method, LDA can artificially specify the number of topics, whereas Top2Vec only generates five topics. In contrast TopicBert's topic division results are more comparable to graph clustering, so we choose TopicBert as a representative to compare with graph clustering.

We calculate word scores for the Louvain algorithm in a manner that mimics the way how topicbert calculates word scores. The titles of all publication nodes under different communities are stored in different lists, and then the words for each community are split out. The 100 most frequent words were first identified and then the scores were calculated using c-TF-IDF for each of the 100 words.

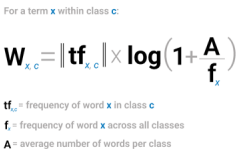


Figure 8: Formula for Calculating C-TF-IDF

The five words with the highest scores are then defined as the most important five words for the topic.

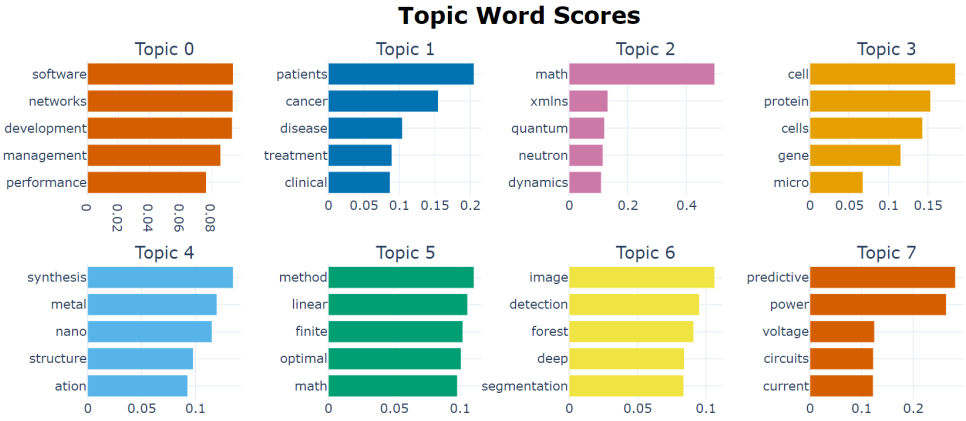


Figure 9: Results of the Louvain Algorithm for Word Scores

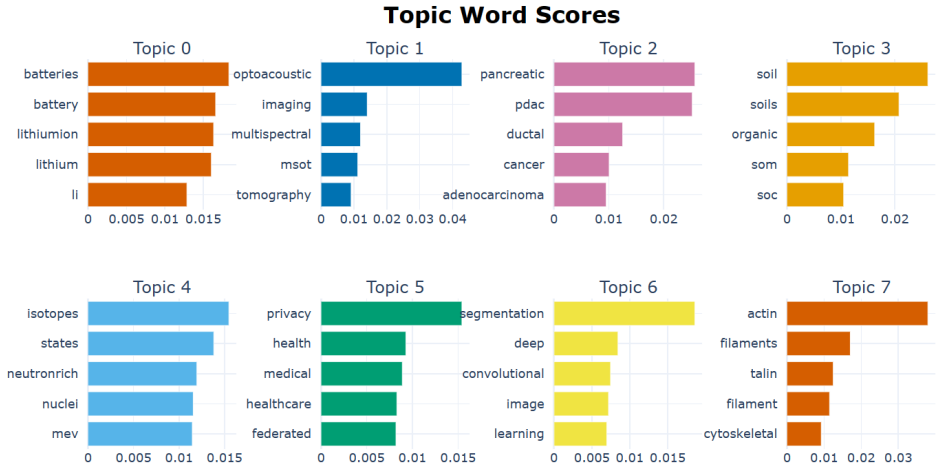


Figure 10: Results of the TopicBert Algorithm for Word Scores

The comparison in Figure 9 and 10 shows that the two methods do not give the same results, but they also have something in common. In TopicBert, Topic 1 is battery related, while in Louvain Topic 7 is only circuit related, there is some connection, but one is the most popular topic and the other is ranked 7th. Louvain's Topic 2 is microphysics related and can correspond to Topic 4 in TopicBert. Louvain's Topic 1 is medical-related and corresponds to Topic 2 and Topic 5 of TopicBert, with cancer being a common term for both. And Topic 6 is identical to both, both being deep learning and computer vision related topics. These show that both approaches are feasible and their results can be used as a reliable reference.

9 Summary

When starting our analysis we first needed to prepare, preprocess and downsample our four used data sets. Especially for the OpenAlex data it was necessary to downsample significantly, while preserving and balancing the distribution of the different topic labels.

Our project then was able to concentrate on the two most important areas for the eventual development of the knowledge graphs for the research related world.
Firstly, we explored different approaches to address the process of the classifying publications according to Field-of-Study taxonomy. We used different language models in the supervised as well as unsupervised approaches. After evaluating the results of two approaches we proposed a hybrid model that combines the strengths of supervised and unsupervised approaches. Additionally we proposed a hierarchical model which is based on the usage of the transformer-based models embeddings to classify the publications.
Secondly, we compared the results of the research cluster modeling using two different perspectives - text-based topic modeling and graph clustering of the TUM publications, thus exploring possible community formations and potential topic clusterings.

10 Conclusion and Future Research

In conclusion, we were able to identify promising directions for our primary goal of extracting topic labels, both for general (more global) concepts (supervised approach) and more specific concepts (unsupervised approach and hybrid model). We also were able to explore the potentials of our secondary goal in graph clustering and text-based topic modeling.
There are still many future research opportunities to improve and expand on our current results.For example, further exploration of the reasons for the different success ratio in predicting topics of different levels is required, in particular in unsupervised approaches. Usage of a different algorithm for the embeddings calculation or similarity score calculation might help resolve the current unbalanced prediction for the topics of different levels, while expanding on combinations of models could offer another solution. Further one could research if differences in data quality play a role, and if the OpenAlex Corpus is a reasonable benchmark.
Finally, by combining both focuses of our project, the TUM publication data could be classified with the help of one of our models and the resulting data set could be used for the further graph clustering exploration, getting one step closer to a more transparent improved academic world.

References

[1] Dimo Angelov. "Top2Vec: Distributed Representations of Topics". In: (2020). arXiv: 2008.09470 [cs.CL].

[2] Iz Beltagy, Kyle Lo, and Arman Cohan. "SciBERT: A pretrained language model for scientific text". In: *arXiv preprint arXiv:1903.10676* (2019).

[3] Vincent D Blondel et al. "Fast unfolding of communities in large networks". In: *Journal of statistical mechanics: theory and experiment* 2008.10 (2008), P10008.

[4] Arman Cohan et al. "Specter: Document-level representation learning using citation-informed transformers". In: *arXiv preprint arXiv:2004.07180* (2020).

[5] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[6] Maarten Grootendorst. "BERTopic: Neural topic modeling with a class-based TF-IDF procedure". In: *arXiv preprint arXiv:2203.05794* (2022).

[7] OpenAlex. *An open and comprehensive catalog of scholarly papers, authors, institutions, and more*. URL: https://openalex.org/ (visited on 02/09/2023).

[8] Jason Priem, Heather Piwowar, and Richard Orr. "OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts". In: *arXiv preprint arXiv:2205.01833* (2022).

[9] Angelo A Salatino et al. "The CSO classifier: Ontology-driven detection of research topics in scholarly articles". In: *Digital Libraries for Open Knowledge: 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019, Proceedings 23*. Springer. 2019, pp. 296–311.

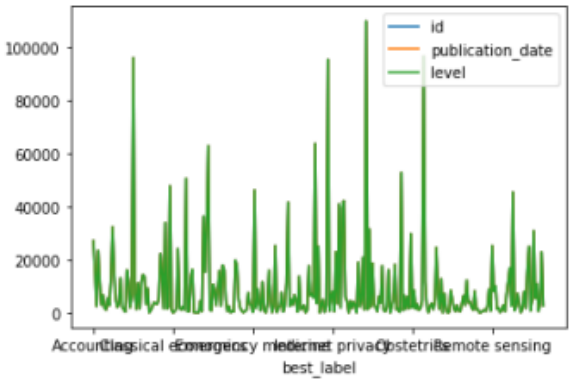
[10] Amanpreet Singh et al. ": A Multi-Format Benchmark for Scientific Document Representations". In: *arXiv preprint arXiv:2211.13308* (2022).

[11] Xiaojin Zhuf and Zoubin Ghahramanif½. "Learning from labeled and unlabeled data with label propagation". In: (2002).

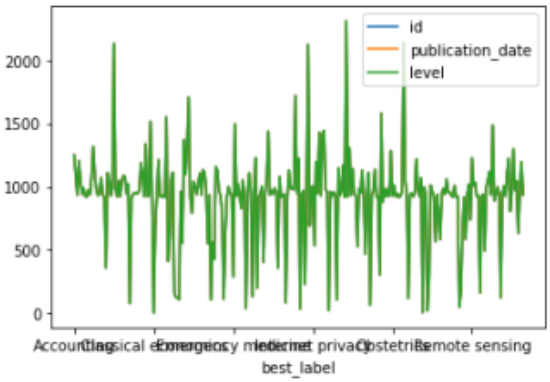
Appendix

10.1 Downsampling

- Level 1 Comparison:

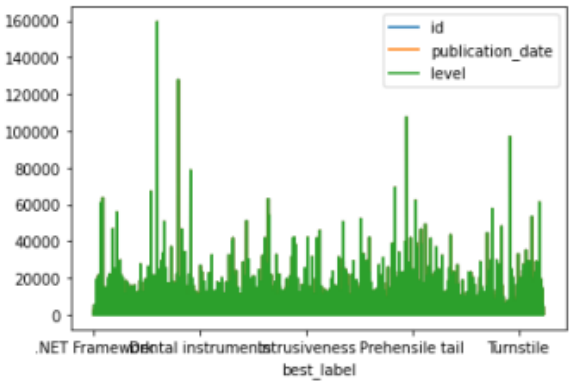


Original dataset

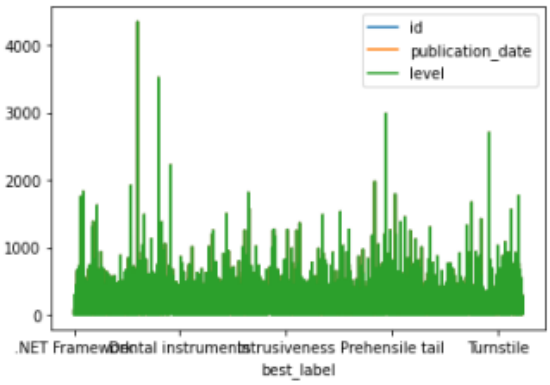


Down sampled dataset

- Level 2 Comparison:

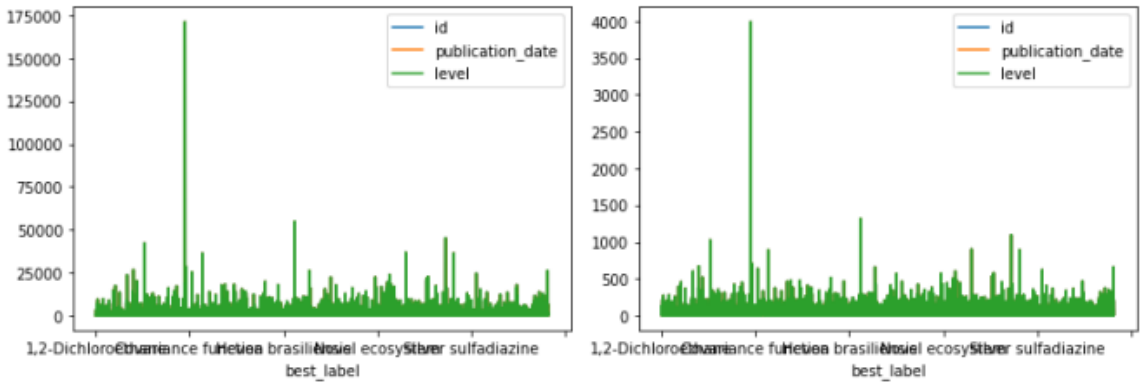


Original dataset



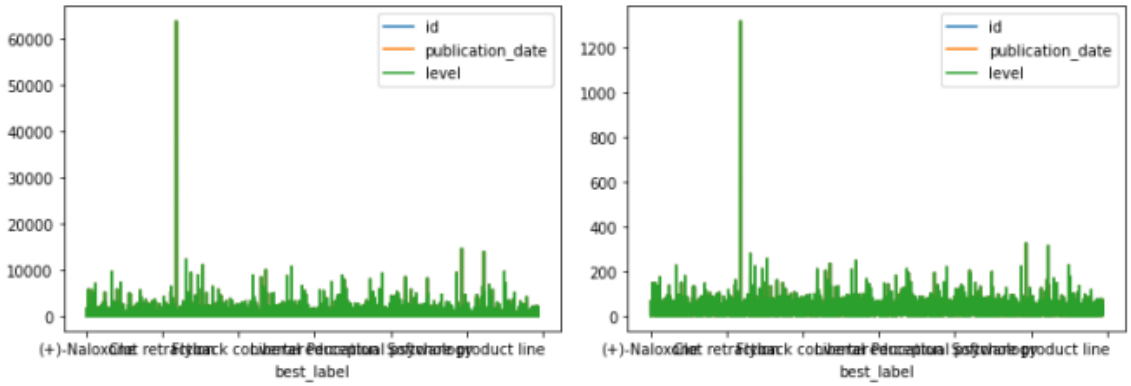
Down sampled dataset

- Level 3 Comparison:



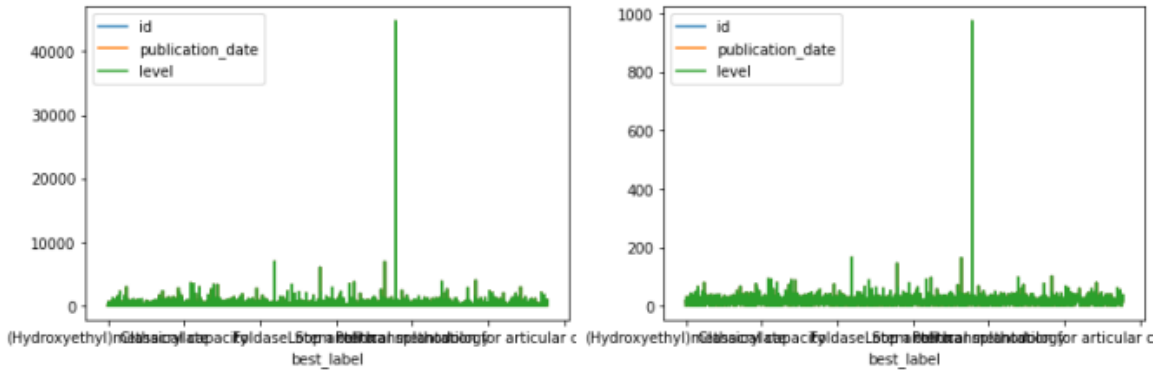
Original dataset | Down sampled dataset

- Level 4 Comparison:



Original dataset | Down sampled dataset

- Level 5 Comparison:



Original dataset | Down sampled dataset

Figure 11: Comparison among levels between original dataset and downsampled dataset from 1 to 5

Topic	Word_0	Word_1	Word_2	Word_3	Word_4
0	cells	cell	activation	genes	membrane
1	neural	networks	network	deep	cognitive
2	diversity	molecular	intervention	reaction	structure
3	protein	gene	proteins	dna	sequencing
4	species	soil	water	different	plant
5	urban	robot	variants	realworld	manufacturing
6	acid	activity	production	using	high
7	study	health	patients	clinical	participants
8	cell	cells	cancer	expression	tumor
9	temperature	surface	properties	materials	material
10	climate	forest	tree	trees	emissions
11	industry	business	concrete	perspective	achieving
12	power	optical	frequency	laser	signal
13	model	dynamics	flow	simulations	results
14	energy	resources	mass	matter	solar
15	data	approach	system	systems	learning
16	outcomes	mmlmrow	al	baseline	et
17	method	model	problem	models	accuracy
18	effects	food	effect	policy	pain
19	patients	risk	study	associated	disease

Table 13: LDA:Five most important word for each topics

10.2 Experimental Results for Supervised Model

Performance	**Precision**						**Recall**						**F1**						#
Method	SVM		BERT		SciBERT		SVM		BERT		SciBERT		SVM		BERT		SciBERT		
Test Set	Alex	Silver	Alex	Silver	Alex	Silver	Alex	Silver	Alex	Silver	Alex	Silver	Alex	Silver	Alex	Silver	Alex	Silver	
Computer science	48.28	**73.68**	50.98	52.17	57.45	54.17	**93.33**	46.67	86.67	40.00	90.00	43.44	63.64	57.14	64.20	45.28	**70.13**	48.15	30
Medicine	84.62	74.19	**96.30**	68.89	92.86	63.04	53.66	56.10	63.41	**75.61**	63.41	70.73	65.67	63.89	**76.47**	72.09	75.36	66.67	41
Chemistry	47.92	60.00	44.68	**61.54**	50.00	58.33	**85.19**	66.67	77.78	59.26	81.48	51.85	61.33	**63.16**	56.76	60.38	61.97	54.90	27
Biology	86.36	86.21	**86.96**	80.65	84.00	78.12	40.43	**53.19**	42.55	**53.19**	44.68	**53.19**	55.07	**65.79**	57.14	64.10	58.33	63.29	47
Materials science	73.33	**91.67**	75.86	75.00	71.88	80.00	78.57	39.29	78.57	42.86	**82.14**	42.86	75.86	55.00	77.19	54.55	**76.67**	55.81	28
Psychology	56.25	61.90	60.61	58.62	58.82	**62.50**	75.00	54.17	**83.33**	70.83	**83.33**	62.50	64.29	57.78	**70.18**	64.16	68.97	62.50	24
Business	48.00	**85.71**	46.43	66.67	41.67	71.43	63.16	31.58	**68.42**	21.05	52.63	26.32	54.55	46.15	55.32	32.00	46.51	38.46	19
Mathematics	63.64	**81.25**	70.37	81.25	67.86	53.85	60.87	56.52	**82.61**	56.52	**82.61**	60.87	62.22	66.67	**76.00**	66.67	74.51	57.14	23
Political science	25.00	**33.33**	28.00	20.00	27.27	31.82	38.46	23.08	**53.85**	23.08	46.15	**53.85**	30.30	27.27	36.84	21.43	34.29	**40.00**	13
Environmental science	62.50	35.71	**70.00**	25.00	60.87	23.53	**78.95**	**78.95**	73.68	73.68	73.68	63.16	69.77	49.18	**71.79**	37.33	66.67	34.29	19
Physics	**76.92**	64.71	68.75	42.42	76.47	48.00	47.62	52.38	52.38	**66.67**	61.90	57.14	58.82	57.89	**59.46**	51.85	68.42	52.17	21
Engineering	**80.00**	60.00	50.00	42.11	77.78	35.48	25.00	37.50	18.75	50.00	43.75	**68.75**	38.10	46.15	27.27	45.71	**56.00**	46.81	16
Sociology	45.45	75.00	50.00	50.00	70.00	**80.00**	34.48	10.34	44.83	10.34	**48.28**	13.79	39.22	18.18	47.27	17.14	**57.14**	23.53	29
Geography	20.00	6.25	27.27	11.76	**40.00**	10.53	27.27	9.09	**54.55**	18.18	**54.55**	18.18	23.08	7.41	36.36	14.29	**46.15**	13.33	11
Economics	**57.14**	45.45	46.15	30.00	40.00	38.46	50.00	31.25	**37.50**	**37.50**	**37.50**	31.25	**53.33**	37.04	41.38	33.33	38.71	34.48	16
Geology	87.88	84.62	88.24	91.67	84.85	**93.33**	87.88	33.33	**90.91**	33.33	84.85	42.42	**87.88**	47.83	89.55	48.89	84.85	58.33	33
History	**62.50**	47.37	58.33	47.50	59.09	43.59	62.50	75.00	58.33	**79.17**	54.17	70.83	**62.50**	58.06	58.33	59.37	56.52	53.97	24
Art	72.73	71.43	60.71	**83.33**	55.56	62.50	69.57	21.74	**73.91**	43.48	65.22	21.74	**71.11**	33.33	66.67	57.14	60.00	32.26	23
Philosophy	**72.73**	57.14	52.17	50.00	63.16	42.86	42.11	21.05	**63.16**	31.58	**63.16**	31.58	53.33	30.77	57.14	38.71	**63.16**	36.36	19
Micro Avg.	60.30	60.47	59.76	52.07	**62.40**	50.54	60.69	44.28	64.79	48.81	**65.23**	47.95	60.50	51.12	62.18	50.39	**63.78**	49.17	463
Macro Avg.	61.64	**62.93**	59.57	54.66	62.08	54.29	58.63	41.99	63.73	46.65	**63.87**	46.54	57.37	46.77	59.23	46.55	**61.28**	45.92	463
Weighted Avg.	65.58	**68.11**	64.74	60.20	66.46	59.74	60.69	44.28	64.79	48.81	**65.23**	47.95	60.13	50.19	62.33	50.21	**63.87**	49.23	463
Samples Avg.	58.97	45.28	63.10	49.96	**64.29**	49.36	63.82	46.17	67.98	50.72	**68.54**	49.87	59.12	44.51	63.38	48.80	**64.24**	48.28	463

Table 14: Experimental results tested on the gold dataset. '#' corresponds to the number of actual occurrences of each label in the preprocessed gold dataset. The results in bold type give the highest score for each indicator among the six models.

	Precision			Recall			F1			Count
	SVM	BERT	SciBERT	SVM	BERT	SciBERT	SVM	BERT	SciBERT	
Computer science	**94.16**	80.92	80.81	**32.73**	30.65	26.65	**48.58**	44.46	40.08	124998
Medicine	**79.87**	72.55	73.27	80.84	80.69	80.29	**80.35**	76.40	76.62	114774
Chemistry	**85.48**	75.56	69.30	40.97	52.57	**54.05**	55.40	**62.00**	60.73	84276
Biology	60.06	62.75	61.09	**67.55**	66.20	60.47	63.59	**64.43**	60.78	82332
Materials science	**89.64**	75.24	72.57	28.88	36.64	36.43	43.69	49.28	48.51	62010
Psychology	**82.09**	74.47	77.19	**36.33**	32.30	28.68	**50.37**	44.88	41.82	60189
Business	**79.91**	58.14	50.43	23.34	42.48	**50.86**	36.13	49.09	**50.64**	50482
Mathematics	**86.08**	74.95	71.69	45.93	50.80	**57.96**	59.90	60.55	**64.10**	48721
Political science	**64.71**	55.26	54.18	15.24	30.52	**42.74**	24.67	39.33	**47.78**	38785
Environmental science	**36.73**	28.22	31.16	**61.37**	50.24	47.41	**45.95**	36.14	37.61	37839
Physics	**60.81**	51.58	47.41	69.86	**76.43**	69.52	**65.02**	61.59	56.37	37638
Engineering	**35.15**	23.58	19.08	22.20	58.08	**67.06**	27.21	33.54	29.71	34567
Sociology	**61.42**	50.19	46.54	5.23	**12.35**	8.52	9.64	**19.82**	14.40	34547
Geography	20.45	**23.40**	20.39	1.52	**2.38**	1.77	2.82	**4.32**	3.25	32528
Economics	**76.15**	54.27	69.45	41.88	45.75	34.46	**54.04**	49.65	46.07	28726
Geology	**95.98**	87.76	89.73	13.02	11.78	**18.82**	22.92	20.76	**31.12**	24407
History	47.59	**52.62**	47.47	29.49	27.81	**29.69**	36.41	36.39	**36.50**	20508
Art	54.39	**64.86**	53.01	10.47	18.66	**22.58**	17.56	28.98	**31.67**	16283
Philosophy	**53.31**	37.03	31.01	14.89	33.36	37.66	23.28	35.10	34.01	10482
Micro Avg.	**69.15**	58.17	54.87	41.07	**45.66**	45.56	51.53	51.16	49.79	944092
Macro Avg.	**66.53**	58.02	56.09	33.78	39.98	**40.82**	40.40	42.98	42.73	944092
Weighted Avg.	**73.04**	63.91	62.39	41.07	45.66	45.56	48.28	49.53	28.58	944092
Samples Avg.	49.36	**54.74**	53.45	46.51	50.41	50.04	43.36	50.58	49.59	944092

Table 15: Models are trained on the silver train set and tested on the OpenAlex test set.

	Precision			Recall			F1			Count
	SVM	BERT	SciBERT	SVM	BERT	SciBERT	SVM	BERT	SciBERT	
Computer science	**46.73**	52.56	**52.58**	**76.38**	75.25	**76.86**	**57.99**	61.89	**62.44**	4525
Medicine	**85.12**	83.10	84.70	59.44	61.39	**64.61**	**70.00**	70.62	**73.30**	8312
Chemistry	**51.74**	49.11	49.94	74.50	74.06	**78.17**	**61.07**	**59.06**	60.94	5200
Biology	65.65	**67.09**	65.37	**49.80**	47.66	**50.14**	56.64	**55.73**	**56.75**	6916
Materials science	**34.34**	33.92	**34.81**	**86.89**	82.95	86.09	49.23	48.15	**49.57**	2997
Psychology	**71.94**	66.64	68.82	**74.66**	72.46	**77.03**	**73.28**	69.43	**72.69**	3722
Business	**32.52**	35.19	**35.89**	67.29	**75.28**	**74.34**	43.85	47.96	**48.41**	1064
Mathematics	**69.21**	64.16	66.48	64.59	69.95	**72.33**	66.82	66.93	**69.28**	4420
Political science	**44.14**	42.43	40.15	53.63	47.29	**57.91**	**48.42**	44.73	**47.42**	3153
Environmental science	**75.90**	71.24	73.65	**26.56**	27.38	**26.65**	**39.35**	**39.56**	39.14	6713
Physics	**88.20**	87.61	**89.09**	48.88	**48.07**	**52.60**	**62.90**	62.08	**66.15**	8912
Engineering	**46.79**	54.08	**54.36**	8.07	8.36	**9.37**	13.77	14.48	**15.99**	4523
Sociology	**20.91**	23.07	20.04	38.93	**34.13**	**47.11**	27.21	**27.53**	**28.11**	709
Geography	4.64	**5.16**	**6.17**	10.00	**10.57**	**11.06**	6.37	**6.94**	**7.92**	1230
Economics	**66.89**	**68.07**	64.34	16.00	37.88	**46.37**	**53.81**	48.68	**53.90**	2864
Geology	**41.71**	41.66	39.78	45.01	90.72	**92.23**	56.88	**57.10**	**55.58**	1519
History	73.07	**74.87**	73.92	**89.40**	32.77	**34.73**	**53.31**	45.59	**47.26**	5401
Art	**10.83**	**8.87**	8.82	41.96	**56.05**	**48.60**	**17.55**	15.32	**14.93**	430
Philosophy	**28.00**	22.12	20.48	**46.05**	21.98	31.45	17.71	22.05	**24.81**	919
Micro Avg.	**54.87**	53.95	54.37	52.35	51.37	**54.56**	53.58	52.63	**54.46**	73529
Macro Avg.	**50.44**	50.05	49.97	51.32	51.27	**54.61**	46.11	45.46	**47.08**	73529
Weighted Avg.	**63.92**	**63.45**	63.86	52.35	51.37	**54.56**	53.51	52.52	**54.47**	73529
Samples Avg.	53.33	53.16	**55.68**	55.62	54.65	**57.98**	52.51	52.15	**54.87**	73529

Table 16: Models are trained on the OpenAlex train set and tested on the silver test set.

	Precision	**Recall**	**F1-Score**
FoS SPECTER: semantic	25.57%	26.46%	26.01%
FoS SPECTER: syntactic	29.83%	31.31%	30.55%
FoS SPECTER: union	25.48%	34.83%	29.43%
FoS Word2Vec: semantic	27.98%	24.93%	26.37%
FoS Word2Vec: syntactic	32.83%	29.70%	31.19%
FoS Word2Vec: union	27.58%	33.09%	30.09%
hierarch. select. no underl.: k=1	18.19%	10.63%	13.42%
hierarch. select. no underl.: k=5	23.24%	11.31%	15.22%
hierarch. select. with underl.: k=1	17.66%	10.33%	13.03%
hierarch. select. with underl.: k=5	22.52%	10.99%	14.77%

Table 17: Table shows results of multiple model instances for each unsupervised model type.

Level	**With Tag (Pred.)**	**With Tag (Target)**	**Recall**	**Precision**	**F1 Score**
0	11,48%	98,90%	3,95%	5,25%	4,51%
1	30,17%	93,96%	5,00%	9,20%	6,48%
2	96,78%	93,05%	45,75%	30,47%	36,58%
3	79,05%	67,39%	41,53%	28,18%	33,58%
4	35,07%	25,08%	29,15%	25,85%	27,40%
5	14,14%	7,76%	15,08%	15,43%	15,25%

Table 18: Table shows the prediction metrics on each level for the best performing FoS Classifier with Word2Vector embedding compared to the OpenAlex corpus on test data

	Precision	Recall	F1-Score
hybrid_semantic	33.04%	38.51%	35.57%
hybrid_syntactic	36.67%	43.42%	39.76%
hybrid_union	31.31%	46.83%	37.53%

Table 19: Table shows results of multiple model instances of the hybrid model.