Topic Modeling Methods for Research Paper Classification

Vidushi Maheshwari

Georgia Institute of Technology

Table of Contents

Abstract

Topic modeling is crucial for document categorization, information retrieval and augmenting databases. This study assesses multilabel hierarchical classification of research papers, using different models, datasets, and metrics. We replicate the original OpenAlex Concept Tagger (v1) as a baseline and compare its performance with an updated OpenAlex Topic Classification model (v2) that uses 92% fewer concepts while efficiently mapping v1 concepts. Through experiments involving concept tree pruning, data cleaning, and embedding utilization, we evaluate the impact of various techniques on classification accuracy. Contingency analysis and sentence similarity assessments allow us to evaluate the unsupervised v2 model's performance against different ground truths. The findings contribute to advancing our understanding of topic classification methodologies and their real-world applicability. Model versions and code are available for reproducibility.

Keywords: NLP, Topic Classification, Topic Modeling

CHAPTER 1: Introduction

Commented [RK1]: Chapter 1. Introduction

In today's digital age, research papers have become increasingly abundant, spanning diverse subjects and disciplines. The sheer volume of academic content available across research journals, conference papers, and scholarly databases can make it quite challenging to find and make sense of specific information. Consequently, there is a growing need for practical computational tools that can help us sort, search, and make sense of this wealth of information. Keyword based information retrieval is not efficient to analyze the insights within this information. Hence, at the very basic level, these knowledge bases can be sieved down based on the topics they represent semantically. The creation of topics from academic articles has been explored by SciKGraph [1] which uses semantic analysis and clustering and using “condition” [2] that employs linguistic structure for logical relation extraction. These methods are useful for construction of knowledge graph and extraction of information. However, a bigger challenge is involved in making models that could generalize and merge these topics across multiple research databases with minimal human intervention. To achieve this goal, we need to identify a conventional hierarchical map of knowledge such that it maps any research paper to a set of few topics as well as update the map based on changing trends.

In this research, we analyze two such topic modeling methods and their corresponding hierarchical maps. We utilize two models developed by OpenAlex [3]. These models don't have official names, but for our purposes, we refer to them as the OpenAlex v1 and v2 model. These two models differ in their architecture and training dataset which are explained in Chapter 3. We train the OpenAlex v1 model using different parameters and dataset to test its accuracy and utilize the inference of OpenAlex v2 model to define metrics to compare different classification

Commented [RK2]: Can you add more details about the two models? e.g., both are unsupervised; main difference is the topic hierarchy methods. The datasets used are MAG (Microsoft Academic Graph) [4] and OpenAlex's publicly available data produced by the Concept Tagger. Overall, the contribution of this study is:

1. Creating a test benchmark for topic classification models on research articles using datasets provided by Microsoft Academic Graph and OpenAlex.
2. Developing and investigating different strategies to improve the classification performance of the OpenAlex Concept Tagger model.
3. A comparative analysis on two versions of topic classification models on the OpenAlex dataset, each utilizing different label sets. The analysis provides insights for researchers and practitioners seeking to optimize topic classification systems in the face of evolving data schemas and labeling requirements.

CHAPTER 2. Literature Review

The topic of any document being read by human is often unambiguous to the reader, but the process to attain this using a computer program is still not refined. Topic modeling refers to unsupervised machine learning techniques that analyze word patterns across documents to cluster them into "topics." While topic modeling methods have laid groundwork, this paper focuses more narrowly on supervised topic classification and make quantitative analysis on comparing unsupervised topic classification with ground truth.

Topic classification uses manually assigned document topics to train models for categorizing new texts. Several traditional statistics-based methods, like TF-IDF, LSA, and LDA, have been commonly employed. However, with the advancements in Deep Learning, methods based on it have become more prevalent.

2.1 Clustering Based Methods

Due to the lack of ground truth data, there have been attempts to create clusters of research papers using relevance in their title or abstract embeddings as well as bibliography – i.e., using a network graph to score similarity. Once the clusters are created, they are given a topic using the similarity in the research papers within the cluster. There have also been attempts to create hierarchical based clusters using the same approach. However, a major drawback is the absence of overlapping concepts due to the clustering-based approach.

Clustering analysis focuses on grouping elements (documents for instance) based on their similarities by representing them in a vector space and extracting dense clusters from the space. Clustering methods bring together similar elements where the similarity is based on their distance in a vector space from each other. A good clustering algorithm satisfies the three conditions: a) minimal requirements of domain knowledge, b) discovery of arbitrarily shaped clusters, and c) efficient on large data sets [7]. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a commonly used clustering algorithm with multiple optimizations like VDBSCAN [10], FDBSCAN [11], GRIDBSCAN [12], IDBSCAN [13], EDBSCAN [14]. The key idea of DBSCAN is that for each point in a cluster, the points reachable within a radius of ε are in the cluster. Moreover, if there are fewer than MinPts in the neighborhood, the point is either a border point or noise. MinPts thus acts as a threshold and determines how dense a cluster needs to be for it to be valid. We hope to find 5 this threshold through analyzing the distribution of the number of occurrences of different words in the corpus.

Unlike k-means (another clustering algorithm), DBSCAN is very efficient in finding clusters of arbitrary shapes and outliers and hence much more useful in a domain where humans do not know. Since MinPts and eps are the only two parameters, we do not have to define the number of clusters the model needs to provide. This is helpful for us since in a large dataset such as Web Of Science, we are unaware of the number of concepts present. The value of the eps parameter, however, is critical in the correct functioning of DBSCAN.

To build an informative knowledge database, we need to have subtopics of larger topics and hence create a hierarchy of topics. HDBSCAN [8] is a modification on DBSCAN such that it iteratively runs DBSCAN over varying values of eps (i.e., radius) to find clusters of different densities. Then, clusters of lower densities are combined to generate a higher-level topic. The paper also focuses on optimization techniques to prevent recomputing the nearest neighbors of all points in a “mini cluster” (i.e., cluster of lower density) by taking a few points as a representative of the cluster and simply storing a boolean matrix. However, due to its unsupervised nature, HDBSCAN cannot map clusters to pre-defined topics, which results in the requirement of Human in The Loop -- something less favorable.

2.2 Deep Learning Based Methods

BERTopic [9], which is an attempt to use Deep Learning in Topic Modeling, resolves the issue of pre-defined topics, by exploring semi-supervised clustering for topic modeling. It utilizes a pre-trained word embedding layer to create the embedding of words and hence a vector space. Then, it performs dimensionality reduction using parametric UMAP [15]. A UMAP algorithm identifies the underlying structure of a graph in the vector space and keeps the relative location of points and neighbors similar to before compression. While non-parametric UMAP optimizes UMAP loss directly over embeddings, Parametric UMAP applies the same cost function over an encoder network. 6 These additional losses, regularize both the vector space as well as the training objectives [15]. After performing Dimensionality Reduction using parametric UMAP, BERTopic performs HDBSCAN and tf-idf to obtain hierarchy-based topics, and documents representing those topics.

The concept tagging model by Open Alex is another state-of-the-art model that utilizes the hierarchy and data of the Microsoft Academic Graph (MAG) Database to tag new research papers in appropriate titles and hierarchies [17]. This model not only utilizes the title and abstract of the document but also the meta-data such as the Journal in which the paper was produced and the Document Type (for example, Patent, Thesis, Book, etc.). Many document classification models utilize only one of two classification systems, title or publication. However, comparative studies [16] have shown that utilizing any one of them in isolation often results in subpar results than using them together. The concept tagging model utilizes a Neural Network that firstly embeds the textual and meta information which then goes through attention layers and then the output is normalized using ReLU layers. An advantage of concept tagging over BERTopic is that it provides multiple topics with varying probabilities whereas BERTopic only provides one topic which corresponds to the cluster.

CHAPTER 3. Evaluation Benchmark

3.1 Models

OurResearch is a nonprofit that builds scholarly research more open, accessible, and reusable [3]. They have attempted to create two different model architectures to classify research papers on topics. These are OpenAlex Concept Tagger [17] and OpenAlex Topic Classification Model [19]. Because of the similarity of their name, we refer to them as v1 and v2 model instead.

3.1.1 OpenAlex v1 Model

The v1 model refers to OpenAlex Concept Tagger's v2 model [17]. It aims to solve a multilabel classification problem. Hence, it is designed such that it gives a predicted weight for each concept and after filtering the weights below a certain threshold, the top k concepts are used to tag. The target for this architecture comes from the 65k OpenAlex concepts described below. These targets are hierarchical in nature.

The input features for the model are paper title (32), abstract (256), document type (1), and journal name (1) [Number representing the length of token tensor].

The model contains trainable embedding layers for all inputs along with multi-head attention for the paper title and abstract. After getting the embedding we have 3 dense layers, each followed by ReLU non-linearity. Finally, a sigmoid layer gives the prediction for each class, and the top k outputs are predicted. The following figure shows the architecture from OpenAlex 2021.

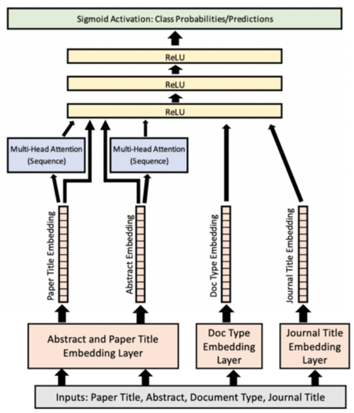

Figure 1: OpenAlex v1 Model Architecture as presented in the Concept Tagger whitepaper [17]

3.1.2 OpenAlex v2 Model

The OpenAlex v2 model refers to OpenAlex Topic Classification Model [19]. For the sake of fairness in comparison of the models, we utilize the topic classification model which uses only title and abstract – unlike the final model that also uses a citation graph to obtain network information.

The v2 model is different in its architecture and data used. The data used by the model consists of non-hierarchical 4.5k concepts which are derived from CWTS data described below.

The model is built by fine tuning on BERT multilingual base model and hence has the capacity to predict on non-English papers as well. The BERT (Bidirectional Encoder Representations from Transformers) model utilizes a multilayer bidirectional transformer encoder architecture with a deeper encoder and larger feedforward networks. To finetune the model, a classification head was added to the BERT model. This was tuned so that given a f-string: f"<TITLE> {title}\n <ABSTRACT> {abstract}", the top 10 classes predicted by the model are thresholded by their confidence and the model and compared with the ground truth.

3.2 Datasets

3.2.1 Microsoft Academic Graph

Microsoft Academic Graph (MAG) was a heterogeneous graph of research entities (including publications, journals, authors, institutions, and fields of study) and relationships between those entities (Sinha et al., 2015). However, it was discontinued in 2021.

Although the snapshot provides abundant information, our focus lies in getting the inference of v2 model specifically using the features 'clean abstract' and 'clean title', with the ground truths being 'fields.' The clean abstract and title remove non-Latin words and punctuations from the original abstract and title of the paper. They also convert them into all lowercase with a maximum length of 32 words for the title and 256 for abstracts.

Fields are the concepts that a paper is tagged to. The fields are divided into 6 hierarchical levels – marked from 0 to 5 – in increasing order of specificity. For example, level 0 consists of "Mathematics" as a concept whereas an example of a level 2 concept is "Directional derivative." Every concept at a lower level has at least one or more ancestors from any level above it. There are overall 71,000 concepts. Their distribution according to the level is presented in the table below:

Level	# of Concepts
0	19
1	292
2	134926
3	331840
4	134926
5	112274

Table 1: MAG Concept Level Distribution

We utilize the 2022-07-08 data snapshot provided by OpenAlex via this link. We use the first 250,00 data points from the Papers.txt. Every paper is tagged with an arbitrary number of concepts with a mean of 17.3 topics and a standard deviation of 5.3.

Commented [RK3]: Describe a bit more how you contructed your test dataset

Some examples of MAG data points are illustrated below:

Original Title: Do Shareholders Really Prefer Risky Projects
Clean title: do shareholders really prefer risky projects
Original Abstract: It is often argued that managers representing shareholders' interests tend to undertake risky projects because equity resembles a call option on a firm's assets. However, this conclusion is not generally true when bankruptcy risk is explicitly modelled. This paper compares the relative strength of the agency cost and the bankruptcy risk in determining managerial choice of cash-flow volatility in a continuous-time framework. Assume the existing debt has covenants which preclude additional borrowing and that bankruptcy is triggered when the cash balance hits zero, I show that for low levels of debt, shareholders prefer to minimize cash-flow volatility. I also work out the critical face value of the debt above which shareholders are risk-seeking rather than risk-avoiding. In short, bankruptcy costs being borne by equity mitigates shareholders' desire for risk.
Clean Abstract: it is often argued that managers representing shareholders interests tend to undertake risky projects because equity resembles a call option on a firm s assets however this conclusion is not generally true when bankruptcy risk is explicitly modelled this paper compares the relative strength of the agency cost and the bankruptcy risk in determining managerial choice of cash flow volatility in a continuous time framework assume the existing debt has covenants which preclude additional borrowing and that bankruptcy is triggered when the cash balance hits zero i show that for low levels of debt shareholders prefer to minimize cash flow volatility i also work out the critical face value of the debt above which shareholders are risk seeking rather than risk avoiding in short bankruptcy costs being borne by equity mitigates shareholders desire for risk
Fields: ['cash flow', 'bankruptcy', 'shareholder', 'equity', 'debt','business', 'agency cost', 'volatility', 'free cash flow', 'equity value', 'monetary economics', 'economics', 'actuarial science', 'financial economics', 'finance', 'agency cost', 'debt to equity ratio', 'financial economics', 'debt', 'shareholder', 'business', 'equity value', 'bankruptcy', 'equity', 'call option']

Figure 2. Example 1 showing a data point of OpenAlex MAG snapshot.

Original Title: in the Eye of the Squid, Loligo pealei
Clean title: in the eye of the squid loligo pealei
Original Abstract: The migration of the screening pigment was investigated in the retina of the intact squid. The action spectrum of pigment migration cor- responds to the action spectrum of the visual pigment, rhodopsin, rather than to the absorption spectrum of the screening pigment. The total number of quanta required for a fixed criterion of pigment migration is the same, when the quanta are delivered over any period of time from 6 s to an hour or more. When less than 3-10 % of the rhodopsin is isomerized, the screening pigment migrates out to the tips of the receptors with a time-course of 5-15 min, and back again over the same period of time. When rather more than 10% is isomerized, the outward migration takes 5-15 min, but the screening pigment does not migrate inwards, even after several hours in the dark. Indirect evidence suggests that the band of screening pigment, when it reaches the tips of the receptors, is ap- proximately equivalent to a filter of 0.6 log units. The spectral sensitivity of the optic nerve response was measured, and was found to be broader than the absorption spectrum of squid rhodopsin in vitro; the broadness could be ex- plained by self-screening, assuming a density of rhodopsin of 0.6 log units at 500 nm.
Clean Abstract: the migration of the screening pigment was investigated in the retina of the intact squid the action spectrum of pigment migration cor responds to the action spectrum of the visual pigment rhodopsin rather than to the absorption spectrum of the screening pigment the total number of quanta required for a fixed criterion of pigment migration is the same when the quanta are delivered over any period of time from 6 s to an hour or more when less than 3 10 of the rhodopsin is isomerized the screening pigment migrates out to the tips of the receptors with a time course of 5 15 min and back again over the same period of time when rather more than 10 is isomerized the outward migration takes 5 15 min but the screening pigment does not migrate inwards even after several hours in the dark indirect evidence suggests that the band of screening pigment when it reaches the tips of the receptors is ap proximately equivalent to a filter of 0 6 log units the spectral sensitivity of the optic nerve response was measured and was found to be broader than the absorption spectrum of squid rhodopsin in vitro the broadness could be ex plained by self screening assuming a density of rhodopsin of 0 6 log units at 500 nm
Fields: ['rhodopsin', 'squid', 'pigment', 'action spectrum', 'absorption', 'biophysics', 'retina', 'optics', 'chemistry', 'biology', 'action spectrum', 'biophysics', 'spectral sensitivity', 'pigment', 'biology', 'absorption spectroscopy', 'optics', 'rhodopsin', 'retina', 'squid', 'loligo pealei']

Figure 3. Example 2 of OpenAlex MAG Format data. Notice the loss of information in abstract due to removal of decimal points, percentage signs and other non-latin characters, alongside the repition of concepts in fields array.

Figure 4. OpenAlex Format data the of removing characters.

Original Title: BEFORE THE BOARD OF APPEALS FOR WASHINGTON COUNTY, MARYLAND
Clean title: before the board of appeals for washington county maryland
Original Abstract: ** * * This action is a\xa0request for a\xa0special exception to\xa0establish a\xa0contractor’s\xa0equipment storage area\xa0for Ram Home Builders with leasable storage bays for others in a\xa0proposed 30′x92′\xa0six-bay garage with an office and\xa0for a\xa0variance\xa0from the minimum lot\xa0area\xa0of 4 acres\xa0to\xa01.5\xa0acres\xa0and\xa0from the minimum 50′\xa0front and\xa0side yard setbacks to\xa040′\xa0and\xa015′, respectively. The subject property is located at 9839 Garis Shop Road, Hagerstown, Maryland,\xa0is\xa0owned\xa0by\xa0the\xa0Appellant,\xa0and\xa0is\xa0zoned\xa0Agricultural\xa0(Ru ral).
Clean Abstract: this action is a request for a special exception to establish a contractor s equipment storage area for ram home builders with leasable storage bays for others in a proposed 30 x92 six bay garage with an office and for a variance from the minimum lot area of 4 acres to 1 5 acres and from the minimum 50 front and side yard setbacks to 40 and 15 respectively the subject property is located at 9839 garis shop road hagerstown maryland is owned by the appellant and is zoned agricultural rural
Fields: ['bay', 'yard', 'archaeology', 'engineering', 'operations management', 'geography', 'agricultural economics', 'transport engineering', 'business', 'yard', 'engineering', 'civil engineering', 'law', 'subject', 'action']

Example 3 of MAG representing effectiveness non-latin

3.2.2 OpenAlex Concept Tagging Data

Commented [RK4]: 3.2 Topic classification models
3.2.1 Based on Wiki concepts
3.2.2. Based on CWTS classifcation

Once the MAG data was discontinued, in the attempt to continue tagging new papers, OpenAlex created its own tagger. We use the research topics tagged by this model to train the v1 model.

OpenAlex utilizes a refined version of MAG data with overall 65,000 concepts distributed within 6 levels. To achieve this, they remove topics that are present in less than 500 papers[17]. The distribution of levels is presented in the table below:

Level	# of Concepts
0	19
1	292
2	21460
3	24768
4	12406
5	6136

Table 2: OpenAlex Concept Level Distribution after thresholding MAG concepts with 500 frequency

Commented [RK5]: How is concept different in OpenAlex

We used 100,000 data points from 2023-08-21 snapshot available here [18] with features 'abstract,' 'title,' 'document type,' and 'journal name,' with the target being ‘concepts.’ Every paper is tagged with an arbitrary number of concepts with a mean of 13.7 topics and a standard deviation of 5.7.

3.2.3 OpenAlex Modified CWTS Data

The CWTS data is created by clustering research papers based on their citations network using Leiden Algorithm [20]. This algorithm creates 4 hierarchical level clusters, with a one-to-one mapping of child to its parent. This implies that a child can have only one parent. Through this algorithm they were able to create 4521 granular level clusters.

The OpenAlex team utilized LLMs with the titles of top 250 most-cited paper in each of the 4521 granular level cluster to give obtain an appropriate topic name which resulted in 4516 unique labels. They then used LLMs to find an parents for these level 3 topics from ASJC Classification codes. Through this, they built a topic hierarchy matching the ASJC's convention with an additional granularity. There is a one-to-one mapping of concepts because of the structure of ASJC.

For example, ASJC defines a domain name, field name, and subfield name to journals. A micro cluster labeled as "Galaxy Formation and Evolution in the Universe" is mapped to the subfield "Astronomy and Astrophysics", field "Physics and Astronomy" and domain "Physical Sciences" where the domain, field and subfield already exists in the ASJC hierarchy and OpenAlex adds a topic as their children.

The concept distribution with respect to level is:

Level	# of Concepts
0	4
1	26
2	252
3	4516

Table 3. CWTS heirarchical topic frequency

3.3 Metrics

To evaluate the performance of a multilabel classification problem, we modify the general definitions of evaluation metrics as described below. $\hat{y}_i$ describes the set of predicted labels and y_i is a set of ground truth labels:

3.3.1 Accuracy. Accuracy for each instance is defined as the proportion of the predicted correct labels to the total number (predicted and actual) of labels for that instance. Overall accuracy is the average across all instances. It is also referred to as the Hamming score.

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i \cap \hat{y}_i|}{|y_i \cup \hat{y}_i|}$$

3.3.2 Recall. It is the proportion of predicted correct labels to the total number of actual labels, averaged over all instances.

$$Recall = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i \cap \hat{y}_i|}{|y_i|}$$

3.3.3 Precision. It is the proportion of predicted correct labels to the total number of predicted labels, averaged over all instances.

$$Precision = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i \cap \hat{y}_i|}{|\hat{y}_i|}$$

3.3.4 F1 Score. It is defined as the harmonic mean of precision and recall. It can also be defined independently as follows:

$$F1\ Score = \frac{1}{n}\sum_{i=1}^{n}\frac{2|y_i \cap \hat{y}_i|}{|y_i| + |\hat{y}_i|}$$

CHAPTER 4. Experiments on v1 Model

We iterated the design of the OpenAlex v1 model by changing its parameters and dataset. The metrics used to quantify the results are described in 3.3. The results are shown below:

Model Name	Dataset notes	Accuracy	Precision	Recall	F1 Score
Baseline model	Vanilla 100k data points	0.17%	19.8%	91.1%	32.5%
Level 0	Only level 0 topics (100k)	21.2%	25.6%	8%	7.5%
Balanced Level 0	Unskewed level 0 topics (50k)	45.7%	50%	4%	5.6%
Pruned Data	Removed levels below level 3 (100k)	1.3%	7%	6.4%	3.2%
Reversed Map	Only level 6 (100k)	3.67%	5.2%	4.5%	3.46%
Static Embedding	Vanilla 100k data points	10.05%	19.7%	9.97%	13.4%

Table 4. Evaluation results on v1 model iteration after training on different datasets. The "Vanilla 100k" dataset comprises direct data from OpenAlex Snapshot. "Only level 0 topics" represents a refined subset with exclusively level 0 concepts. "Unskewed level 0 topics" is a balanced subset, consisting of 50k data points from "Only level 0 topics." "Removed levels below level 3" includes all data points from the snapshot but excludes levels below 3. Similarly, "Only level 6" encompasses data points solely related to level 6 concepts. The "Static Embedding" model employs GLoVE static embedding layers, while "Checkpoint" conducts inference using OpenAlex's provided model.

4.1 Baseline v1 Model

The Baseline OpenAlex v1 Model was created by running training on the OpenAlex's provided architecture on 100k data points from OpenAlex snapshot. There is an extremely high precision which is a result of model predicting the most frequent concepts with the highest weight but do not have enough confidence for less frequent concepts that, as a result, do not pass threshold.

This, along with model's low accuracy, we hypothesized, might be a result of sparse data. For example, every data point has a concept from level 0 but very few have concepts from level 3. Therefore, we narrowed our training data for only level 0.

4.2 OpenAlex level 0 Concept Tagger

As anticipated, the model's accuracy improved, although it remained suboptimal. Even after expanding the dataset to encompass 2 million data points, there was no noticeable enhancement in accuracy. Upon examining the dataset, we identified a highly skewed frequency distribution among level 0 topics within the 100k data points of the OpenAlex concept tagger as displayed in Figure 2.

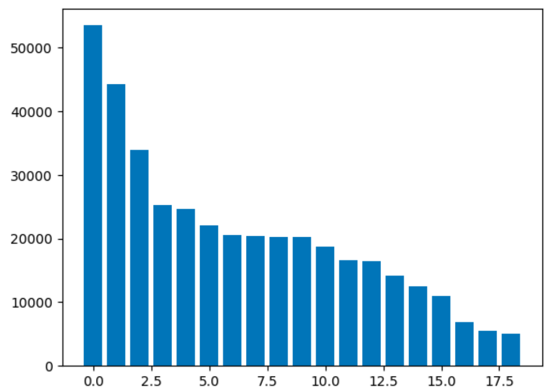

Figure 5: Frequency Distribution of OpenAlex Level 0 concepts

Topics such as biology and computer science came for around 50% of all data points, whereas environmental science and geology accounted for only 5% of the dataset. The authors of v1 Model have incorporated a Sigmoidal focal loss to prevent such distribution to ruin the model's performance, but it remained sub-par with the data distribution of snapshot. Hence, to reduce the variance of the number of topics that must be predicted by the model, we create a concept tagger for only level 0 concepts. Hence, we investigate balancing the dataset.

4.3 OpenAlex Balanced level 0 Concept Tagger

4.3.1 Processing of Unbalanced Multilabel Dataset

A naïve approach to balance this set is to randomly choose higher frequency concepts and drop the respective data points. However, due to data being multilabel, dropping concepts of higher frequency led to removal of lower frequency concepts. Moreover, this removal also drastically reduced the total number of data points as is evident with Figure 6 where only 2k data points were left and 6 of the 19 level 0 concepts were completely dropped.

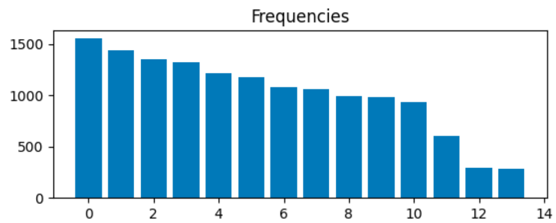

Figure 6: Frequency distribution after Naive removal of high frequency labels

So, we progressed to build out dataset through an additive algorithm. In this algorithm, we employ an iterative approach to systematically process the dataset, prioritizing the inclusion of least frequent elements over more common ones. Initially, we initialize an empty Data Frame and a set to track processed labels. Through a loop that continues until a predefined number of unique labels have been processed, we analyze the dataset's column to determine the frequency of each label. Subsequently, we select the least frequent element that has not been processed previously. We then calculate a threshold value representing the minimum number of elements to be included in the final Data Frame, ensuring adherence to a specified limit. Through an iterative process, we progressively add elements to the Data Frame, giving precedence to the least frequent ones. This iterative inclusion continues until either the threshold is reached, or all available elements have been processed. By adopting this methodological approach, we ensure a balanced representation of elements from the dataset, where fewer common elements are prioritized, thus promoting diversity and comprehensive coverage of the data.

This algorithm gives us a dataset of 50k points, the most frequent element accounted for 20% of labels whereas the least frequent with 3% -- a much better trade off than the naïve approach.

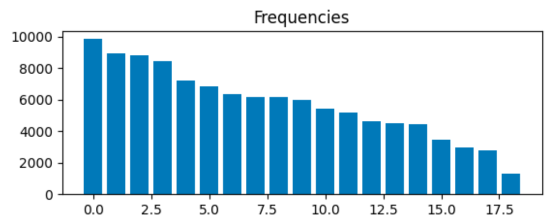

Figure 7: Frequency distribution after additive build of dataset

We then use this data to train the Balanced Level 0 model in the same fashion and obtain higher accuracy and precision. A lower recall implied that the model was bot confident in many of its predictions, so we attempt to reduce the threshold score but that resulted in lower accuracy and precision – meaning that the model was not only underconfident but also incorrect. Since there were little adjustments that could be made to the dataset, we go back to our vanilla 100k dataset which contains more information.

4.4 v1 Pruned Data

Balancing the frequencies of 65k concepts in vanilla 100k posed a significant challenge, necessitating substantial amounts of data and computational power. However, rather than immediately attempting to achieve perfect balance, we initially addressed the issue of data anomalies by pruning the tree. Our aim was to ensure that our model could acquire the maximum amount of information possible while still maintaining a balanced dataset. To achieve this, we opted to prune the tree by removing nodes at level 3 and below. We chose this threshold because of the disproportionately high number of concepts present at level 3, which could potentially skew the dataset's balance.

We create a model that could learn only till level 2 and if it works well, we can chain another model to learn the concepts in the lower levels. However, this accuracy for this came out to be suboptimal.

4.5 Reverse Map Tagger

Since there is a direct mapping of lower-level concepts to higher concepts, we attempt to create a model that could tag only level 5 concepts which reduced the number of classes by 90% to 6k concepts. However, we realized that in such a case, the model requires abundant data points as the frequency of each class is very low.

4.6 Utilizing static GloVe Embeddings on Reverse Tagger

To improve the metrics of the reverse map tagger, we change the trainable embedding layers to GloVe untrainable layers to reduce the effort that would be required by the model to train its embedding layers.

CHAPTER 5. Comparison with OpenAlex v2 Model

With the subpar performance of OpenAlex v1 model, we investigate the performance of OpenAlex v2 model using its inference and comparing it to ground truth.

The OpenAlex v2 model uses CWTS's LLM generated labels as ground truth for classification. CWTS acknowledges that they "did not perform a systematic evaluation of the quality of the LLM results, our impression is that the results are a substantial improvement over the results obtained using our original methodology, or variants of that methodology" [21]. Moreover, since these labels are generated by LLM, there is no direct mapping between these and the MAG concepts.

Consequently, to gauge the v2 model's efficacy, we employ MAG data, widely accepted for topic classification, as a quantitative benchmark. Given the disparity between MAG concepts and LLM-generated topics, we employ methods such as Similarity Search and Contingency

Analysis to compare the v2 model's outputs with MAG labels. These approaches not only facilitate present evaluations but also offer a framework for comparing various classification and clustering tasks in the future.

5.1 Contingency Analysis

Firstly, we perform inference on the MAG data to obtain v2 model predictions on it. Then, we create a matrix with columns as labels of the v2 model and rows as labels of v1. For any data point we find the labels given by both the models and increment the value of the cell associated to the two labels. Since our data was inherently biased, this associative matrix gave extremely strong associations in frequent fields but not the same in others. To resolve this, we also normalize each row with respect to the frequency of v2 labels. The two matrices show the mapping between the classes.

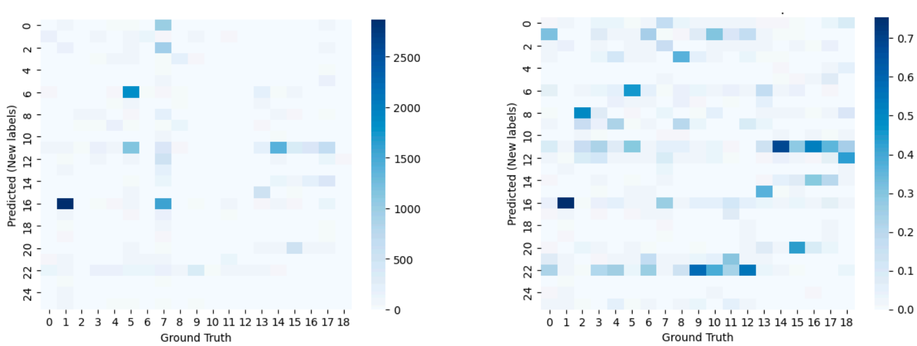

Figure 8: Association (Left) and Normalized Association (Right) Matrix

To concretely detect a statistically significant dependence between the labels, we run a chi-square test to compare if the dependence between the labels is by chance or due to an underlying relationship. Comparing the 19 level 0 labels of v1 with 26 level 1 labels of v2 model, we get a p-value of 0.0 indicating a high degree of relationship between the concepts. Similarly, we go down a level to compare the distribution of 292 level 1 concepts of v1 with 252 level 2 concepts of v2 and we get a p-value of 0.09. Therefore, a strong dependence of the two data hierarchies can be concluded within these concepts.

5.2 Sentence Similarity

Now that we have proved a dependence between the labels at individual levels, we wanted to test the semantic similarity of the words created by the new model to the ones by the old model. To do so, we find semantic similarity between labels marked by MAG and the inference by the v2 model. Since, this classification is multilabel, we find the semantic similarity between the two sets of labels for each data point by finding the centroid of the set embedding via averaging the embedding vectors of one set and then find the cosine similarity between the centroids.

We compare the cosine similarity score between the ground truth and all level 3 v2 labels using pretrained embedding models of sentence transformers library such as paraphrase-albert-small-v2 [22], and paraphrase-MiniLM-L6-v2[22] and allenai-specter-2 [6] which are all fine-tuned with datasets such as Wikipedia, and PubMed research papers making it fit for our analysis.

Of the 120k data points, we find the cosine similarity of the predictions with ground truth (GT-P) and compare them with the average similarity of ground truth with all possible predictions (GT-RP). We find that for every data point, the GT-P cosine similarity is at most 0.1 variant than GT-RP similarity.

Moreover, we find the rank of the similarity of the predicted label with respect to all the labels. The average rank of these came out to be 2281 of 4561, implying that the prediction's cosine similarity is in the bottom 50 percentile of all possible similarity scores.

Hence, using the outcome of both the tests, we can claim that the labels predicted by the Open Alex Classification model do not hold semantic significance with MAG data.

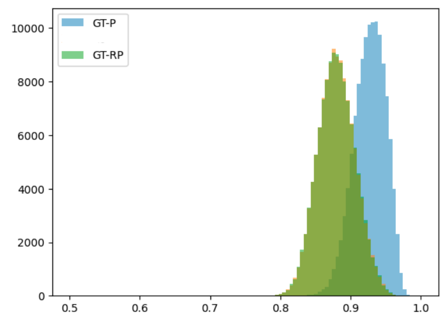

Figure 9. Cosine similarity of ground truth to predictions (blue) and ground truth to all predictions (green)

CHAPTER 6. Discussion

Commented [RK6]: Summarize the key takeways
- unbalanced
- relation between old and new model
- what are remaining issues
- what are the implications and why are we solving the problem etc.

Our research explored various hierarchical multilabel classification datasets and models for research paper tagging. We aimed to replicate the OpenAlex Concept Tagger model (v1) and experimented with data cleaning and architecture changes for enhanced results. We encountered challenges such as highly correlated and inherently biased data, which we addressed by using additive algorithms to construct a dataset and pruning the concept tree. Additionally, we employed pretrained embeddings, which led to improved model performance.

Upon exhausting different versions of the OpenAlex v1 model, we obtained inferences from the OpenAlex Topic Classification model (referred to as v2). Since v2 uses a distinct topic hierarchy compared to v1, we introduced metrics for comparing the two models, including similarity search and contingency analysis. These metrics aid in comparing different multilabel topic classification models. We were able to verify the dependent relationship between the topics of two models using chi-squared test while also proving that the labels are not semantically dependent.

We see that even though v2 model uses 92% less concepts, it can efficiently map v1 model's concepts through its topics. Moreover, unlike v1 dataset, v2's topics have a one-to-one mapping with its parents which make the concepts less complex. It's coherence with ASJC structure also provides an easy pattern to integrate the model with different databases.

Looking ahead, there are promising avenues for further enhancement. Incorporating citation graphs into the evaluation of the v2 model stands out as a crucial next step to deepen our understanding of its performance. Additionally, exploring models capable of dynamically updating topics based on emerging trends remains an open area for future research. Investigating deterministic unsupervised clustering and hierarchical models for similar tasks could also unlock new insights and improve model robustness.

The different models, their version logs, and their code can be found on this link https://github.com/OKN-CollabNext/topic-classification-model.

Commented [RK7]: we can make this repo public once we are ready

Works Cited

[1] Tosi, M. D., & dos Reis, J. C. (2021). SciKGraph: A knowledge graph approach to structure a scientific field. Journal of Informetrics, 15(1), 101109. https://doi.org/10.1016/j.joi.2020.101109

[2] Jiang, T., Zhao, T., Qin, B., Liu, T., Chawla, N. V., & Jiang, M. (2019). The role of "condition." *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. https://doi.org/10.1145/3292500.3330942

[3] Priem, J., Piwowar, H., & Orr, R. (2022). OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts. ArXiv. https://arxiv.org/abs/2205.01833

[4] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. 2015. An Overview of Microsoft Academic Service (MAS) and Applications. In Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion). ACM, New York, NY, USA, 243-246. http://dx.doi.org/10.1145/2740908.2742839

[5] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. ArnetMiner: Extraction and Mining of Academic Social Networks. In Proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'2008). pp.990-998.

[6] Singh, A., D'Arcy, M., Cohan, A., Downey, D., & Feldman, S. (2022). SciRepEval: A Multi-Format Benchmark for Scientific Document Representations. Conference on Empirical Methods in Natural Language Processing.

[7] K. Khan, S. U. Rehman, K. Aziz, S. Fong and S. Sarasvady, "DBSCAN: Past, present and future," The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014), Bangalore, India, 2014, pp. 232-238, doi: 10.1109/ICADIWT.2014.6814687.

[8] McInnes, L., Healy, J., & Astels, S. (n.d.). (working paper). hdbscan: Hierarchical density based clustering.

[9] Grootendorst, M. (n.d.). (tech.). BERTopic: Neural topic modeling with a class-based TFIDF procedure

[10] P. Liu, D. Zhou and N. Wu, "VDBSCAN: Varied Density Based Spatial Clustering of Applications with Noise," 2007 International Conference on Service Systems and Service Management, Chengdu, China, 2007, pp. 1-4, doi: 10.1109/ICSSSM.2007.4280175.

[11] B. Liu, "A Fast Density-Based Clustering Algorithm for Large Databases," 2006 International Conference on Machine Learning and Cybernetics, Dalian, China, 2006, pp. 996-1000, doi: 10.1109/ICMLC.2006.258531.

[12] O. Uncu, W. A. Gruver, D. B. Kotak, D. Sabaz, Z. Alibhai and C. Ng, "GRIDBSCAN: GRId Density-Based Spatial Clustering of Applications with Noise," 2006 IEEE International Conference on Systems, Man and Cybernetics, Taipei, Taiwan, 2006, pp. 2976-2981, doi: 10.1109/ICSMC.2006.384571.

[13] B. Borah and D. K. Bhattacharyya, "An improved sampling-based DBSCAN for large spatial databases," International Conference on Intelligent Sensing and Information Processing, 2004. Proceedings of, Chennai, India, 2004, pp. 92-96, doi: 10.1109/ICISIP.2004.1287631.

[14] A. Ram, A. Sharma, A. S. Jalal, A. Agrawal and R. Singh, "An Enhanced Density Based Spatial Clustering of Applications with Noise," 2009 IEEE International Advance Computing Conference, Patiala, India, 2009, pp. 1475-1478, doi: 10.1109/IADCC.2009.4809235.

[15] Tim Sainburg, Leland McInnes, Timothy Q. Gentner; Parametric UMAP Embeddings for Representation and Semisupervised Learning. Neural Comput 2021; 33 (11): 2881–2907. doi: https://doi.org/10.1162/neco_a_01434

[16] Shu, F., Julien, C.-A., Zhang, L., Qiu, J., Zhang, J., & Larivière, V. (2019). Comparing Journal and paper level classifications of Science. Journal of Informetrics, 13(1), 202–225. https://doi.org/10.1016/j.joi.2018.12.005

[17] OpenAlex: End-to-End Process for Concept Tagging. (n.d.). Retrieved from https://docs.google.com/document/d/1q3jBlEexskCZaSafFDMEEY3naTeyd7GS/edit#heading=h .gjdgxs

[18] Priem, J., Piwowar, H., & Orr, R. (2022). OpenAlex Snapshot - 23-08-21 (Complete) [Data set]. arXiv. https://docs.openalex.org/

[19] OpenAlex. (n.d.). OpenAlex topic classification whitepaper.docx. Google Docs. https://docs.google.com/document/d/1bDopkhuGieQ4F8gGNj7sEc8WSE8mvLZS/edit#heading=h.5w2tb5fcg77r

[20] Van Eck, N. J. (2024). Classification of research publications based on data from OpenAlex (Version 2023nov) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.10560276

[21] Eck, N., & Waltman, L. (2024, January 24). An open approach for classifying research publications. Leiden Madtrics. https://www.leidenmadtrics.nl/articles/an-open-approach-for-classifying-research-publications

[22] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in

Natural Language Processing. Association for Computational Linguistics. http://arxiv.org/abs/1908.10084