Hybrid Citation Count Prediction

Pranav Balgi (8674)
pgbalgi@cs.ucla.edu

Jason Tay (9377)
jtay20@g.ucla.edu

Garvit Pugalia (8127)
garvitpugalia@g.ucla.edu

Songyu Sun (5519)
songyusun@g.ucla.edu

Dang Nguyen (0983)
dangnth@cs.ucla.edu

Abstract

This paper employs hybrid models, that integrate citation graph mining and text-based analysis, for predicting citation counts over time. We first build a baseline longitudinal encoder-decoder model, employing graph convolutional networks (GCN) along with long short-term memory (LSTM), to predict the next-year citation counts for Physics papers. Then, we conduct experiments with various encoder architectures, metadata features, and text embeddings as a comparison. Additionally, we explore the feasibility of building hybrid models with incomplete citation graphs by experimenting with subsets of dynamic graphs. Our findings indicate that simpler graph encoding models such as GCN outperform more complex ones. In terms of the embedding method for the papers' abstracts, we achieved the best performance with SPECTER2. We found that additional metadata features such as author and number of pages do not significantly improve performance. Finally, we demonstrate that fairly accurate and robust citation count prediction can be achieved even with a subset of the original graph.[1].

1 Introduction

Citation count prediction is the task of predicting the number of citations a research paper will receive in the future. This prediction problem serves as a crucial task in academia, offering insights into the potential impact and significance of research papers. Citations not only inform readers about the sources behind the information found in a paper but also provide important context for the proposed research in a broader academic field. Thus, the task of citation count prediction is crucial for researchers, publishers, and academic institutions aiming to gauge the significance of the work soon after its publication. Traditionally, an h-index is used as a way to gauge the scientific productivity and perceived impact of a researcher. It is derived from the researcher's most frequently cited papers and the total number of citations they have garnered in external publications [7].

In the area of citation count prediction, two prominent methodologies have emerged: text-based approaches, that analyze the content of the papers or additional metadata, and citation graph mining approaches, that explore the structure and dynamics of citation networks. Both approaches have demonstrated efficacy in tackling the problem of predicting the citation counts for papers in the future using past citation graph networks or features such as the abstract and author. However, the use of hybrid models that employ both methodologies remains a fairly under-explored field. Notably, more research can be done to study how to best combine text-based approaches with citation graph networks to yield better citation count predictions. These hybrid encoder-decoder approaches are particularly useful, as they can apply to any task that combines some form of graphical data with additional metadata, such as airline traffic prediction [17].

[1] Code and data is available at https://github.com/pgbalgi/cs247.

Preprint. Under review.

Our research in this paper looks at developing these hybrid models effectively to integrate the in-depth text-based analysis of paper content with the broad insights of a citation network. We will be focusing on the regression form of the citation count prediction problem to predict the future counts for a given paper based on its past citation network structure and various features of the paper itself. The hybrid approach shows promise by leveraging both the quality of a paper's contents and the information on a paper's influence in the community. This means we can adjust to changes in research trends and citation practices over time to develop more robust models for citation count prediction.

2 Related Work

2.1 Text-based approaches

For text-based methods, we found that prior work focuses on extracting features using the titles, abstracts, and bodies of papers to predict future citation counts. [13] demonstrated a text-mining approach that leverages the actual text of research papers to forecast citation counts. Their research offers a novel approach to forecasting citation counts for academic papers by using many traditional natural language processing techniques to extract a variety of text features from the paper, such as bag-of-words representations, parts of speech (POS) tags, Doc2Vec, and improved Latent Semantic Analysis (I-LSA). They were able to achieve fairly accurate citation count predictions by optimizing feature selection and feeding them into a three-layer deep learning model. Similar papers have also harnessed other non-text features of the papers. [16] employed a linear regression model to predict citation counts with features such as author authority and author ranking. This research and other similar work on text-based citation count prediction highlight the importance of semantic and lexical features for determining a paper's impact, and building robust prediction models. Consequently, they show that the actual contents of the papers and the surrounding metadata hold valuable information on a paper's future citation trajectory.

2.2 Graph mining approaches

In contrast to the text-based approaches, citation graph mining uses the structure and dynamics of citation networks to predict citation counts. For example, [18] proposed an end-to-end deep learning network that takes in the whole citation network as input to develop a highly sophisticated model for citation count prediction. [12] introduced a new feature, based on graph pattern mining in the citation network, to be able to more accurately predict future citation counts. These two approaches underscore the significance of the graph structure behind paper citations in predicting a paper's future impact, and how its future influence grows as a result of this structure.

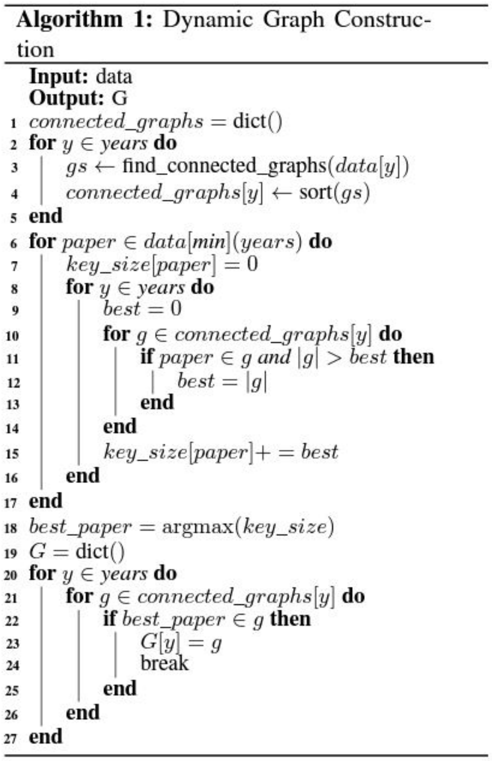
2.3 Hybrid approaches

There also exist models that recognize the complementary strengths of text-based and citation graph mining methods through hybrid models. [11] presented an innovative approach with a support vector regression model that combines text features with network features such as the g-index and h-index of authors. [9] offered a new perspective on this problem by using dynamic citation graphs instead of statically predicting citations over time. Their novel solution accurately predicts the number of citations a paper will receive over time by combining a dynamic citation network with feature vectors from the paper's abstract, author, and venue. This paper's proposed encoder-decoder model with a graph convolutional network (GCN) based encoder [10], and a recurrent Long Short-Term Memory (LSTM) based decoder [5, 8] serves as the baseline for our research.

3 Methodology

3.1 Baseline model

The baseline hybrid model utilizes both text features and the citation network. The spatial and temporal information is captured in dynamic citation graph embeddings. As shown in Figure 1, the model has an encoder-decoder structure. The encoder is a graph neural network (GNN) that takes in our feature vectors and a sequence of citation networks to get a temporal embedding. Since we are focused on the use of dynamic graphs, the encoder will generate sequences of temporal graph embeddings for each year. The graph input for the encoder module is structured as an adjacency matrix with the edges representing citations between papers. Each node has a matrix for its feature vectors to be contained in. This temporal embedding is then passed to an LSTM decoder which generates the subsequent citation count predictions for each paper in the dynamic graph. We used the same algorithm for constructing the dynamic graphs as the original paper in Algorithm 1. This algorithm helps us generate the sequences of citation graphs over time by finding the ideal dynamic graph. In this case, the ideal graph is simply the sequence of connected graphs that contain the most nodes over the whole duration of the data. Since the encoder is a GNN, it requires the feature vectors for every node in the network to produce the temporal embeddings. Therefore, we split the temporal embeddings into batches (i.e. training, validation, testing) to feed into the LSTM decoder.



Algorithm 1: Dynamic Graph Construction

```
Input: data
Output: G
$connected\_graphs = \text{dict}()$
for $y \in years$ do
    $gs \leftarrow \text{find\_connected\_graphs}(data[y])$
    $connected\_graphs[y] \leftarrow \text{sort}(gs)$
end
for $paper \in data[min](years)$ do
    $key\_size[paper] = 0$
    for $y \in years$ do
        $best = 0$
        for $g \in connected\_graphs[y]$ do
            if $paper \in g$ and $|g| > best$ then
                $best = |g|$
            end
        end
        $key\_size[paper] += best$
    end
end
$best\_paper = \text{argmax}(key\_size)$
$G = \text{dict}()$
for $y \in years$ do
    for $g \in connected\_graphs[y]$ do
        if $best\_paper \in g$ then
            $G[y] = g$
            break
        end
    end
end
```

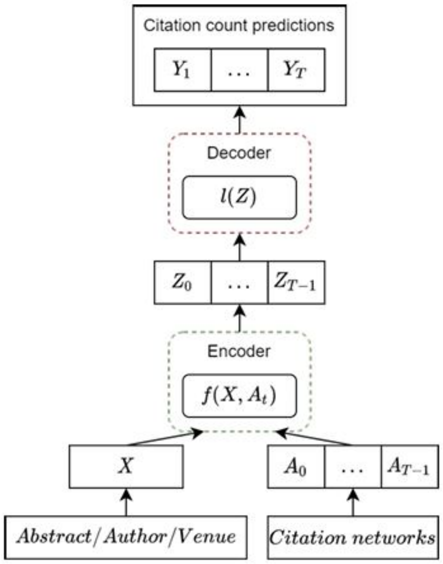


Figure 1: The baseline encoder-decoder model for prediction. The figure is taken from [9].

3.2 Encoder architecture changes

While the original baseline model employs a GCN encoder, we sought to investigate alternative encoder architectures to determine if there are models better suited to capturing the relationship between neighboring nodes and yielding improved results. To investigate this, we integrated three distinct encoder architectures: Graph Convolutional Network (GCN), Graph Attention Network (GAT), and GraphSAGE, each characterized by its unique approach to neighbor information aggregation. GCN, with its straightforward averaging method, posits equal weight across neighbors. GraphSAGE introduces a learnable function for adaptable aggregation, so it leverages node feature information to generate node embeddings for unseen data [6]. Thus, GraphSAGE could outperform the old GCN encoder by generalizing better to unseen papers and generating embeddings based entirely on its local citation neighborhood. [14] employs an attention mechanism to assign dynamic weights to neighbors so GAT could capture more nuanced relationships compared to the simple aggregation method used in the old GCN encoder. A deeper look into the encoder is shown in Figure 2.

3.3 Text embeddings

In our study, we adopt a similar approach as the original model by utilizing the pre-trained Bidirectional Encoder Representations from Transformers (BERT) model, as introduced by [3]. Specifically, we leverage the SciBERT variant, which has been pre-trained on scientific literature and is therefore

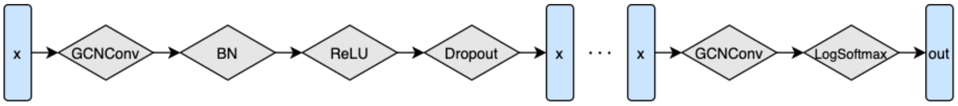


Figure 2: The inner architecture of the encoder module, where the GCNConv is modified to use other approaches. The figure is taken from Assignment 6.

well-suited to our dataset [1]. By feeding tokenized versions of the abstract text into SciBERT, we obtain informative embeddings that capture the essence of each paper's content. This feature vector, combined with the citation network structure, offers both fine-grained contextual information at the paper level and insights into broader graph structures. Additionally, we explore the effectiveness of another novel embedding model known as SPECTER, proposed by [2]. SPECTER utilizes a pre-trained language model to generate document-level embeddings for scientific papers, leveraging citation graphs to discern the relevance between documents. In our experimentation, we specifically employ SPECTER's successor, SPECTER2[2]. As the model was built for a multi-task benchmark, SciRepEval 2022, which included the task of citation intent classification, it is anticipated to yield more robust embeddings for the papers, more closely related with their respective citation counts.

3.4 Metadata features

In addition to the adapted text embeddings, our model incorporates additional feature vectors to capture author-related information, which could significantly influence a paper's impact over time, and page count, which could relate to the amount of content available to cite. Established authors often see their papers accrue citations rapidly in the years following publication, underscoring the importance of author representation. While the baseline model employs author rank, computed as the normalized ranking based on an author's total citation count across all papers, we propose two alternative author representation methods: the h-index and the g-index. The h-index, as proposed by [7], provides a robust measure of a researcher's scientific output by quantifying the number of papers published against their respective citation counts, offering a heuristic for gauging overall research impact. Conversely, the g-index, introduced by [4], assigns greater weight to highly-cited articles, offering a nuanced perspective on an author's scholarly contributions. These alternative metrics present promising avenues for enhancing our understanding of author impact within citation networks, potentially providing more refined insights compared to the straightforward author rank. For example, the g-index could have improved performance since it is more sensitive to high citation count papers under the assumption that researchers with highly successful papers garner greater attention overall for their publications.

3.5 Graph subsets

The original encoder-decoder model operates on the entire citation graph network, which in practice, can pose scalability challenges. Citation graph networks in popular domains often experience continuous expansion with newly published papers and citations. Constructing the complete citation network graph for the encoder entails significant computational overhead, as each citation must be labeled as an edge, and the addition of newly published papers introduces more nodes. Moreover, for interdisciplinary fields, generating comprehensive citation network graphs across multiple domains necessitates large datasets. Finding all the citations for a given paper requires more data mining overhead to locate the citations in papers from various databases. In our investigation, we explore strategies for constructing more resilient hybrid models capable of accommodating incomplete citation graphs while maintaining accuracy in citation count predictions. By training on subsets of the original graphs, our approach enhances model versatility, addressing constraints associated with obtaining complete graph data in diverse settings. To stimulate the real-world incomplete data collection scenario, we randomly select $x\%$ of all papers and apply our model to the corresponding subgraph. Alternatively, we randomly select $x\%$ papers published in each year. More advanced subset selection can also be applied to further improve the efficiency of the data curation process. However, it is out of the scope of this research so we leave it for our future direction.

[2] https://huggingface.co/allenai/specter2

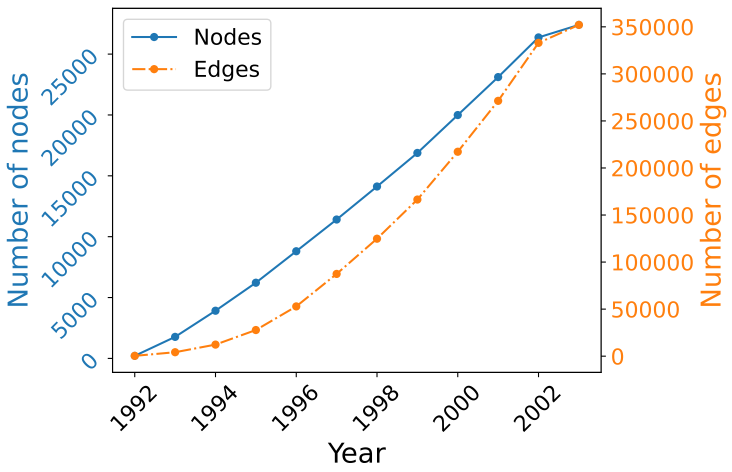


Figure 3: The growth of graph size in the dynamic graph extracted from the HEP-PH dataset.

4 Experiments

4.1 Data

The dataset utilized in our experiments is sourced from the arXiv repository, specifically the High Energy Physics Phenomenology (HEP-PH) citation graph[3]. This dataset encompasses papers published within the domain from 1992 to 2003, offering a comprehensive snapshot of scholarly activity during that period. The original graph comprises approximately 30,000 nodes and 350,000 edges, ensuring a substantial corpus for analysis. Within this network, the largest connected graph consists of approximately 28,000 nodes and 350,000 edges, reflecting the interconnected-ness of papers within the field. To facilitate model training and evaluation, we partitioned the papers into training, validation, and test sets, allocating 60%, 20%, and 20% of the data, respectively, to each split.

4.2 Data Processing

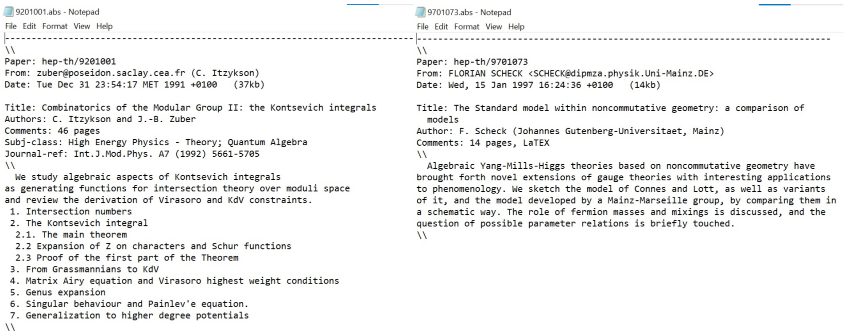
Dynamic graph construction: We adopt the procedure described in Algorithm 1 to build our dynamic graph construction. As the number of papers before 1992 is too small, we start to build graphs from the year 1992. While treating each citation graph as an undirected graph, we still consider the case when two papers cite each other when counting the number of citations.

Metadata extraction: Due to the inconsistent formatting of the metadata text files, as shown in Figure 4, we used regular expressions to extract the relevant data: abstract, title, list of authors, and page number. While some sections such as abstract and title were well-formatted, other parts had extra information that had to be removed (affiliated university for author) or were missing altogether (page count). For missing page counts, we imputed the mean page count for all papers published in the same year. We also had to take into account inconsistent whitespaces when parsing each text file.

4.3 Experimental Setup

Hyperparameters: In our training methodology, we employed a text embedding dimensionality of 768 to represent textual information. For the encoder architecture, we utilized a two-layer Graph Neural Network (GNN) with a hidden dimensionality of 256, coupled with a dropout rate of 0.5, Rectified Linear Unit (ReLU) activation, and one-dimensional Batch Normalization. As for the decoder architecture, we employed a Long Short-Term Memory (LSTM) network with a hidden dimensionality of 128. Our training regimen included an initial learning rate of 0.001, which was decayed by a factor of 10 whenever the validation loss failed to improve over a span of 10 epochs. We trained our models for a total of 200 epochs, with early stopping implemented if the validation loss

[3] https://www.cs.cornell.edu/projects/kddcup/datasets.html



```
------------------------------------------------------------------------------
\\
Paper: hep-th/9201001
From: zuber@poseidon.saclay.cea.fr (C. Itzykson)
Date: Tue Dec 31 23:54:17 MET 1991 +0100   (37kb)

Title: Combinatorics of the Modular Group II: the Kontsevich integrals
Authors: C. Itzykson and J.-B. Zuber
Comments: 46 pages
Subj-class: High Energy Physics - Theory; Quantum Algebra
Journal-ref: Int.J.Mod.Phys. A7 (1992) 5661-5705
\\
  We study algebraic aspects of Kontsevich integrals
as generating functions for intersection theory over moduli space
and review the derivation of Virasoro and KdV constraints.
 1. Intersection numbers
 2. The Kontsevich integral
  2.1. The main theorem
  2.2 Expansion of Z on characters and Schur functions
  2.3 Proof of the first part of the Theorem
 3. From Grassmannians to KdV
 4. Matrix Airy equation and Virasoro highest weight conditions
 5. Genus expansion
 6. Singular behaviour and Painlev'e equation.
 7. Generalization to higher degree potentials
\\
```

```
------------------------------------------------------------------------------
\\
Paper: hep-th/9701073
From: FLORIAN SCHECK <SCHECK@dipmza.physik.Uni-Mainz.DE>
Date: Wed, 15 Jan 1997 16:24:36 +0100   (14kb)

Title: The Standard model within noncommutative geometry: a comparison of
  models
Author: F. Scheck (Johannes Gutenberg-Universitaet, Mainz)
Comments: 14 pages, LaTEX
\\
  Algebraic Yang-Mills-Higgs theories based on noncommutative geometry have
brought forth novel extensions of gauge theories with interesting applications
to phenomenology. We sketch the model of Connes and Lott, as well as variants
of it, and the model developed by a Mainz-Marseille group, by comparing them in
a schematic way. The role of fermion masses and mixings is discussed, and the
question of possible parameter relations is briefly touched.
\\
```

Figure 4: The raw text representation of papers' metadata, showcasing the inconsistent formatting concerns.

Table 1: Performance (in MAE) comparison between two subset selection strategies. *All years* denotes selecting randomly from the set of all papers while *By year* denotes selecting paper each year randomly. For each method, we retain only 50% of the papers.

Method	Train (↓)	Validation (↓)	Test (↓)
All years	0.5556	0.6879	0.6997
By year	0.5362	0.6661	0.6783

did not exhibit improvement over a consecutive period of 20 epochs. These specifications ensured robust training and convergence of our models for accurate citation count prediction.

Label transformation: To enhance training stability, we applied a transformation to the citation counts (i.e., labels) using the logarithmic function $y = \log(1 + c)$, where c represents the original citation count. This transformation mitigates the impact of extreme outliers in the citation counts and promotes more stable training dynamics [9, 15]. By incorporating this transformation into our training pipeline, we ensure robust convergence and improved performance of our models in predicting citation counts accurately.

Evaluation: In our evaluation process, we employed the Mean Absolute Error (MAE) as our loss function, which quantifies the discrepancy between the predicted and actual citation counts. The MAE is calculated as:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where y_i represents the actual citation count, $\hat{y}_i$ represents the predicted citation count, and n is the total number of samples.

4.4 Varying encoder architectures

To compare various encoder architectures, we enhanced the baseline model by introducing a new parameter for a convolution layer. Subsequently, the model was adapted to generate a series of convolutional layers based on this parameter, while retaining other aspects such as batch normalization and dropout. We employed PyTorch's GCNConv, SAGEConv, and GATConv to implement GCN, GraphSAGE, and GAT, respectively. As can be seen in Figure 5, GCN surprisingly yields the best performance on the validation and test sets among three different encoders. GraphSAGE is possibly overfitting, producing the lowest MAE on the train set but a higher MAE on unseen papers compared to GCN. In terms of GAT, its performance is much worse than the previous two on all three splits. It is important to note that in our experimental setup, the more advanced models (GAT and GraphSAGE) received minimal hyperparameter tuning and regularization. This potentially explains the performance gap between them and GCN.

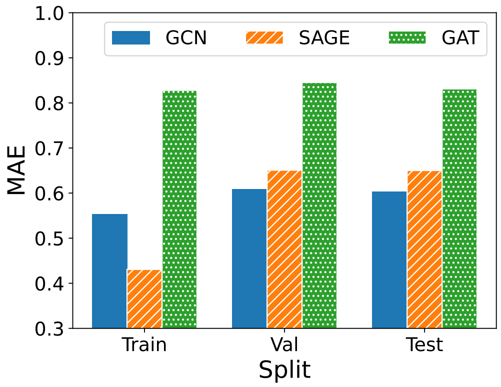


Figure 5: The effect of varying the encoder on the performance of our model.

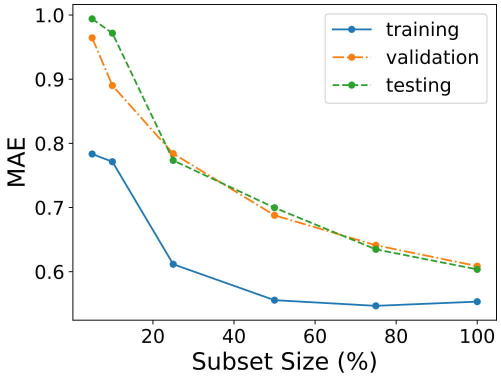


Figure 6: The effect of changing the subset size on the performance of our model. Increasing the graph size improves the performance in all 3 data splits.

4.5 Utilizing graph subsets

In order to evaluate the model's performance when trained on subsets of the original citation network graph, we employed multiple randomly selected subsets representing 5%, 10%, 25%, 50%, and 75% of the original dataset. Furthermore, we examined the alternative approach involving random paper selection by year for 50% of the papers. Figure 6 illustrates the performance of different subset fractions. As expected, the more data, the better performance illustrated by the downward trend in validation and testing losses. In terms of train loss, our model can achieve a decent result with just 50% of the total nodes. It is also worth noting that the low absolute size of the smaller subsets may contribute to the worse performance. For example, the 5% random subset only has 480 nodes. Comparing two different selection methods, selection papers by year perform slightly better in all three splits as shown in Table 1. This is because selection by year preserves a large enough number of papers for each year, which is not the case for selecting randomly without considering the publication year.

4.6 Modifying text embeddings

We generated the SciBERT text embeddings for the abstracts of all papers in our dataset. The abstracts underwent tokenization using the uncased, pre-trained SciBERT wordpiece tokenizer (scivocab). Subsequently, we established a mapping between paper IDs and the corresponding text embeddings generated by SciBERT. This mapping facilitated the incorporation of text embeddings into the dynamic graph. Similarly, we followed the same procedure with the SPECTER2 model, utilizing its respective tokenizer and model. Lastly, we also tested a fine-tuned version of SciBERT that uses the embeddings for all the abstracts in previous years to fine-tune the model for papers for the next year. The results are shown in Figure 7. SPECTER2 performed the best, which aligns with the initial training objective of the model and the large amounts of scientific literature it was trained on. The fine-tuning process did not produce better results, however, we ran into a Huggingface max token issue that restricted the size of embeddings to 512, as compared to 768 for the other baseline models.

4.7 Changing author representation

In addition to the author rank from the baseline model, we implemented g-index and h-index to test different author representations. Using the processed dataset, we filtered by the authors to iterate through the papers associated with each of them to conduct the calculations for g-index and h-index. For h-index, this simply required enumerating through the papers in descending order of citation count until the number of papers with citations greater than or equal to their position in the sorted list matched or exceeded the position itself. The h-index is then equal to the last position reached. For g-index, we sorted the papers associated with each author by citation count in descending order and calculated the cumulative number of citations. The g-index is the largest number such that the

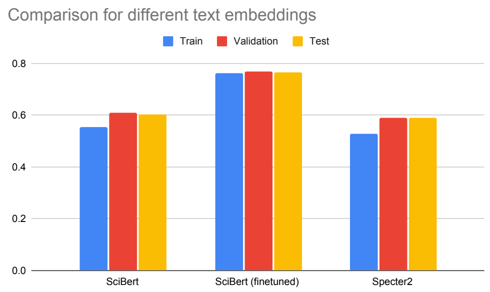


Figure 7: Comparison between different abstract text embeddings

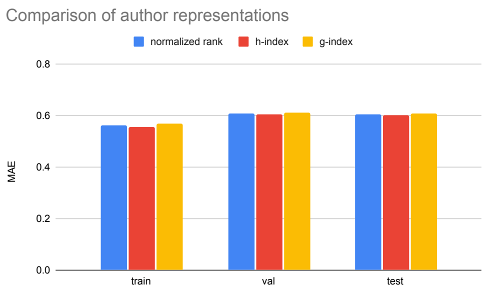


Figure 8: Performance for various representations of paper authors

top g papers together received at least g^2 citations. We iterated through the sorted list to find the maximum value of g satisfying this condition. With each of the precomputed author rank, g-index, and h-index, we added them as a feature vector to each node before encoding. The results are shown in Figure 8. Unsurprisingly, h-index performed the best, which further bolsters the metric's popularity in measuring author impact. Since we used the whole dataset to accumulate an author's citation count, there is a potential data leak concern that we hope to further analyze.

5 Conclusion

In this study, we explored the integration of textual data and graph-based structures to predict citation counts for research papers within the Physics domain. Our investigation revealed that simpler models, specifically the Graph Convolutional Network (GCN) as the encoder, not only outperformed but also generalized more effectively than complex neighborhood aggregation schemes like Graph Attention Networks (GAT) and GraphSAGE. This finding underscores the efficiency of simpler architectures in handling the intricacies of graph-based data for citation prediction tasks.

A significant discovery of our research was the effectiveness of these models even when only 50% of the graph data was utilized. This not only validates the robustness of our approach for scenarios with partial data availability but also highlights the importance of strategic data subset selection—opting for annual subsets over random distribution across the dataset resulted in superior model performance. This insight opens avenues for effective data management and utilization in predictive models where full data accessibility may be a challenge.

Further analysis into the components of paper metadata revealed that the textual content, specifically the abstract and title, played a pivotal role in enhancing the predictive capabilities of our models. While other metadata features like author details and page count had minimal impact, our findings stress the critical value of a paper's content in determining its citation potential. Our comparison between pre-trained scientific models demonstrated that the more recent SPECTER2 model surpasses SciBERT in performance, marking it as a superior choice for tasks involving scientific text understanding and prediction. Moreover, our evaluation identified the h-index as the most effective metric for measuring author rank, offering a reliable standard for future studies focusing on author contributions and their impact on citation counts.

This research not only advances our understanding of the factors influencing the citation counts of papers but also showcases the potential of combining textual analysis with graph-based models to enhance predictive accuracy. Our findings encourage the adoption of simpler, more efficient model architectures and highlight the importance of strategic data usage and the selection of textual features in predictive tasks. As we move forward, the insights garnered from this study could contribute to the development of more sophisticated and accurate predictive models in the realm of scientific research analytics and other tasks that display graphical structures along with node features.

References

[1] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: Pretrained language model for scientific text. In *EMNLP*, 2019.

[2] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. SPECTER: Document-level Representation Learning using Citation-informed Transformers. In *ACL*, 2020.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Leo Egghe. Theory and practise of the g-index. *Scientometrics*, 69(1):131–152, 2006.

[5] Alex Graves and Alex Graves. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pages 37–45, 2012.

[6] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[7] Jorge E Hirsch. An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences*, 102(46):16569–16572, 2005.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] Andreas Nugaard Holm, Barbara Plank, Dustin Wright, and Isabelle Augenstein. Longitudinal citation prediction using temporal graph neural networks. *arXiv preprint arXiv:2012.05742*, 2020.

[10] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[11] Avishay Livne, Eytan Adar, Jaime Teevan, and Susan Dumais. Predicting citation counts using text and graph mining. In *Proc. the iConference 2013 workshop on computational scientometrics: Theory and applications*, pages 16–31, 2013.

[12] Nataliia Pobiedina and Ryutaro Ichise. Predicting citation counts for academic literature using graph pattern mining. In *Modern Advances in Applied Intelligence: 27th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2014, Kaohsiung, Taiwan, June 3-6, 2014, Proceedings, Part II 27*, pages 109–119. Springer, 2014.

[13] Priya Porwal and Manoj H Devare. Citation count prediction using weighted latent semantic analysis (wlsa) and three-layer-deep-learning paradigm: a meta-heuristic approach. *Multimedia Tools and Applications*, pages 1–32, 2023.

[14] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018. accepted as poster.

[15] Gideon Maillette de Buy Wenniger, Thomas van Dongen, Eleri Aedmaa, Herbert Teun Kruitbosch, Edwin A Valentijn, and Lambert Schomaker. Structure-tags improve text classification for scholarly document quality prediction. *arXiv preprint arXiv:2005.00129*, 2020.

[16] Rui Yan, Jie Tang, Xiaobing Liu, Dongdong Shan, and Xiaoming Li. Citation count prediction: learning to estimate future citations for literature. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1247–1252, 2011.

[17] Jiangni Yu. A new way of airline traffic prediction based on gcn-lstm. *Frontiers in Neurorobotics*, 15, 2021.

[18] Qihang Zhao. Utilizing citation network structure to predict citation counts: A deep learning approach. *arXiv preprint arXiv:2009.02647*, 2020.