# Foundation Of Data Science

DR.Khalaj


Electrical Engineering Department

Parsa Hatami 400100962

**GitHub**

Project Phase 1

February 9, 2025

Research Assistant

# Overview

You are meant to develop an AI assistant that can help a user find relevant papers. You will be given a dataset of scholarly articles up to the year 2017. You will try to infer useful information about the articles and use this information to enhance your AI assistant's inference. More data will be scraped from internet resources. The data you have crawled will be used as testing data throughout the course of the project.

You are required to provide your code for this project in the courseware alongside the report. Moreover, both of these files should be uploaded to **GitHub** using Git. You must add us as collaborators to your repository, as it will make monitoring/grading your results much more straightforward.

# Crawling

This section should have been completed from the previous phase until now. Keep this data as it will prove useful later on.

# EDA

## *Basic*

- Create 3 bar charts. For each bar chart draw the number of publications in a given year range. Do this for the ranges 1937-1950, 1950-1970, 1970-1990. Compare the three.
- Create a bar chart of the number of references over the years.
- Create a bar chart of the number of authors over the years.
- Find the Pearson correlation coefficient and Spearman Rank correlation coefficient between the *number of authors* and *number of references*.
- Find the Pearson and Spearman correlation coefficient between the *number of authors* and *number of citations*.
- Draw a bar chart of the title length over the years.
- Draw a wordcloud of the abstracts.

- Find a fitting correlation coefficient between the *title length* of each paper with the *title length of the papers it references*.
- Find the top 10 authors with the most publications.
- Find the top 10 authors with the most citations.
- Find the top 10 papers with the most references.
- Find the top 10 papers with the most citations **within the dataset**.
- Find a way to see how well the number of publications can predict the number of citations for a given author.

## *Network Analysis*

### *Citation Network (Paper-Paper Network)*

**Nodes:** Papers, **Edges:** Citation relationships (from `references` field)

- Plot the clustering coefficient over time to observe how interconnected the citation network becomes.
- Compute the average path length and diameter to understand the network's reachability.
- Identify influential papers using PageRank.

### *Co-authorship Network (Author-Author Network)*

**Nodes:** Authors, **Edges:** Co-authorship relationships (if two authors have co-authored a paper, they are connected)

- Compute network density per year to analyze how collaborations evolve.
- Identify influential researchers using centrality measures (degree, betweenness, closeness).
- Find communities of authors working in similar fields.

### *Venue Network (Conference-Journal Network)*

**Nodes:** Conferences/Journals (from `venue` field), **Edges:** If a paper cites another paper from a different venue, an edge is created between venues.

- Analyze interdisciplinary collaborations between venues.
- Find the most influential venues using centrality metrics.
- Identify emerging fields based on newly formed venue connections.

### *Temporal Evolution of the Citation Network*

Construct the citation network for different years and analyze how the structure changes over time.

- Plot the network density per year to observe citation growth.
- Identify bursts of influential papers (papers that suddenly get many citations).
- Examine how new papers integrate into the existing network.

## Data Extrapolation via Clustering

### *Community Detection*

1. Find the author-author network. Make sure each author retains the data about the papers it has published.
2. Find the author communities in this network using a fitting clustering algorithm. You are allowed to use any algorithm, as long as the results make sense (Louvain, Walktrap, Hierarchical, Spectral, Newman, etc.). You must train 3 clustering algorithms.
3. Evaluate your clustering algorithm using the clustering coefficient and find the best clustering algorithm from the 3 clustering algorithms.

**Hint:** You can use a subset of the data to train and evaluate your clustering algorithms, find the best algorithm, and then run the best clustering algorithm on the whole data to reduce computation time.

### *Naming the Communities*

1. Each paper has an *abstract* and a *title*. We want to extract **keywords** from these text fields. Use KeyBERT for extracting keywords from each abstract/title and aggregate them together to form a single list of words as *keywords* for each *paper*.
2. Associate each paper with the corresponding community of authors. A paper might be in two or more communities, so make sure to handle this correctly.
3. Aggregate the keywords of each paper from each community to get the *keywords of the community*. Explain your aggregation method, and report the keywords of each community in your report.

4. Give each community a name, and explain how you came to choose these names.

### *Paper-Paper Clustering via Embedding*

1. Embed the abstract and title of each paper using a fitting embedding model such as BERTopic or SentenceBERT, and aggregate them together.
2. Your embedding should contain the information of *keywords of the community*. An easy way would be to append the keywords as plaintext.

   **Note:** A paper might be included in several communities! An easy way to overcome this is to use the union set of these keywords.
3. Use any clustering algorithm of your choice to cluster these papers together.
4. Evaluate your clustering algorithm using DBI and Silhouette scores.

   **Hint:** Silhouette scores are calculated for each point. To evaluate your overall clustering technique, you must somehow aggregate them together, and report the aggregate. For instance, the *average silhouette score*. This means you can choose a small subset of points to calculate their average on, however, the silhouette score for each of these points must be calculated over the whole data. We are unsure if current Python packages support this, but you are allowed to use these optimizations in case long runtimes become an issue.
5. Report the unique venue values of the dataset.
6. Compute the Jaccard Similarity Index of your clusters with the clustering of **venues**. The venue of each paper is a decent label for clustering as well!

## Citation Regressor

In this section, we want to create a model that can estimate how many **citations** a given paper will have based on its *abstract, title, year*, and (**Bonus**) author-author network.

1. The training data is the abstracts of the entire dataset. Preprocess the training data, and use appropriate embeddings. You are allowed to use LLM embeddings (e.g. BERT models such as BERTopic or SentenceBERT) to boost your model's performance. Do not forget the train/validation split.

2. Find the best regression model and hyperparameters using **AutoML** or **GridSearch**.
3. **This is where your crawled data will come into play!** Your crawled data is the *test set*. Use your crawled data to find the estimated citation count and report important regression metrics (RMSE, MAE, R2, etc.). Give explanations of the reasons your model has overcome/failed these major challenges:
   (a) **Time gap** between training/validation set and test set.
   (b) **New concepts/authors** that may be introduced over time, not being in the training data.

**Bonus: Author-Author Network**

The idea is that scholarly paper authors have relations with each other that may be a great predictor of how many citations a given paper is going to have. To utilize this network, you will have to resort to using a **GNN**. You are free to use any architecture.

## Product

The end goal is to create a product called **Research Assistant**. The Research Assistant is an LLM enhanced by RAG that can:

1. Find corresponding papers for a given topic;
2. And (hopefully) the LLM can perform analysis on these relevant papers and provide reasonable reports.

### *RAG*

Create a pipeline that takes a string input from the user, and will retrieve relevant papers from the database of papers. The database is a **vector database**, and is indexed based on the embeddings of abstracts/titles of your original dataset + crawled dataset and returns the relevant papers using a similarity metric (cosine similarity, L2, etc.). Use your model as the embedder for this stack. You are allowed to use **any** tech stack for this, such as:

1. **(Bonus) Classic**: Use a database such as Postgres, add the `pgvector` or **(Even more bonus)** `pgai` extension, and
2. **Langchain**

3. **LlamaIndex**
   - The user should also be able to retrieve papers based on the **author**. Use NER to your advantage. **Bonus**: You may use the LLM to find the author name, and then somehow search the database by that.
   - You should use KeyBERT to extract keywords from the user's input so that the embeddings match the embedding method from previous sections.

### *Research Assistant (Bonus)*

Prompt-engineer an LLM such that it gives reports/summaries of the retrieved papers from 4.1. Your model must be an LLM text generation model. OpenAI API, any HuggingFace model, Llama, or DeepSeek.ai is accepted. If you do not see your model of choice in this list, chances are it is accepted. You can consult the teaching assistant if you have any questions.

# Implementation ans Results

## *EDA*

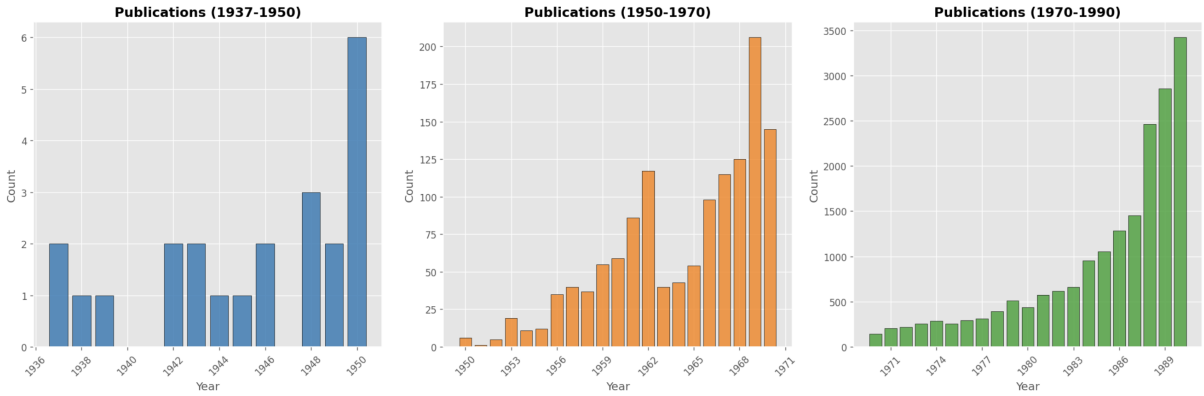



**Figure 1.** Bar Charts

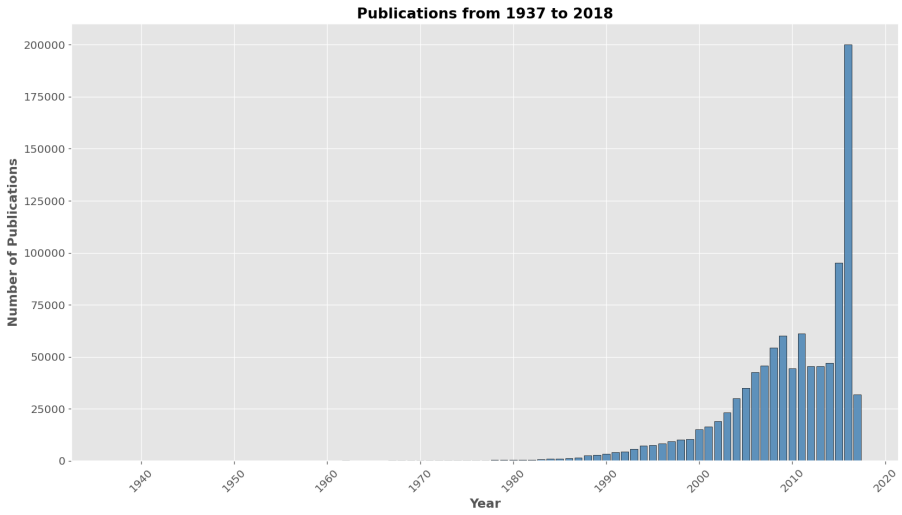



**Figure 2.** Bar Chart

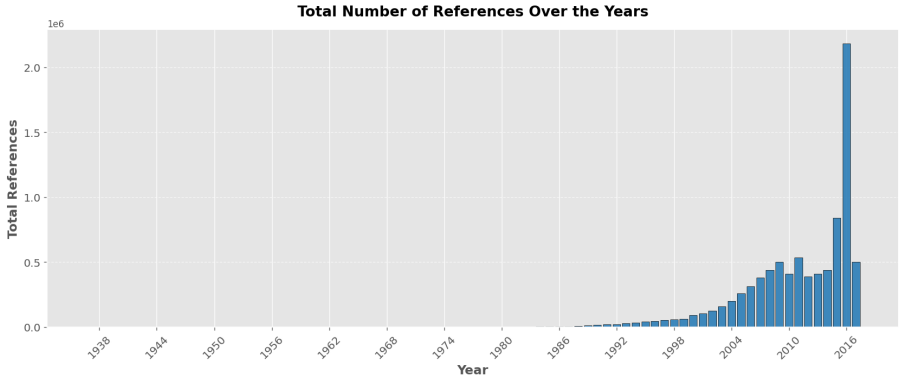



**Figure 3.** Refrences

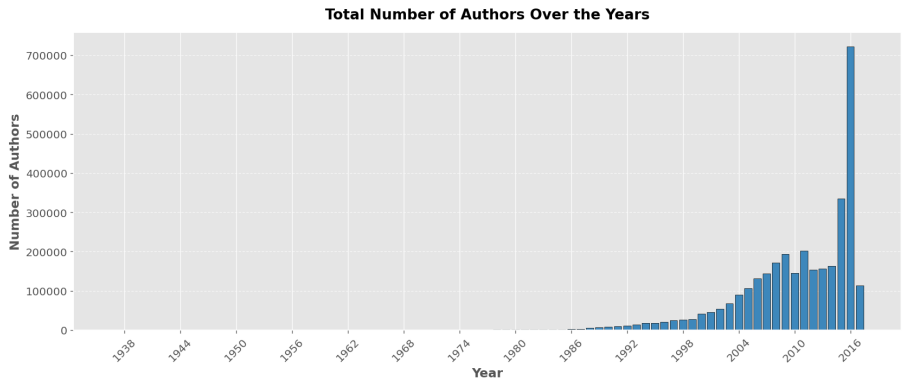



**Figure 4.** Authors

| Metric | Value |
|---|---|
| Pearson Correlation | 0.056 |
| Spearman Correlation | 0.0872 |

Table 1: Correlation Metrics

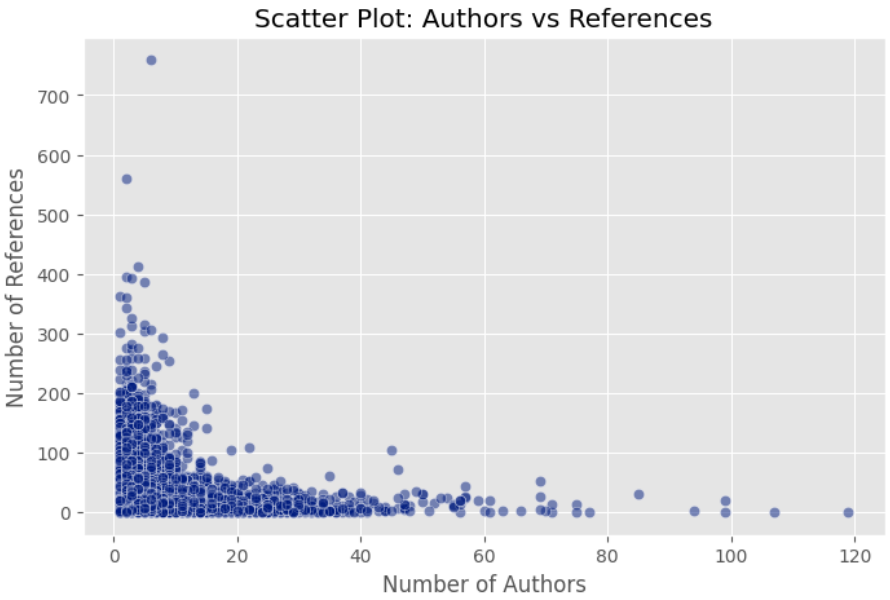



**Figure 5.** Correlation

| Metric | Value |
|---|---|
| Pearson Correlation | -0.0028 |
| Spearman Correlation | -0.0166 |

Table 2: Correlation Metrics

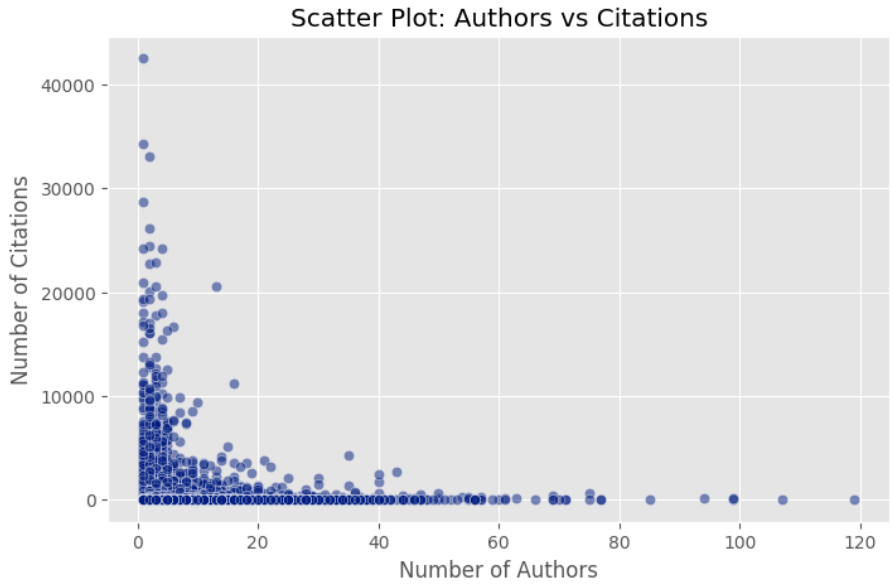



**Figure 6.** Correlation

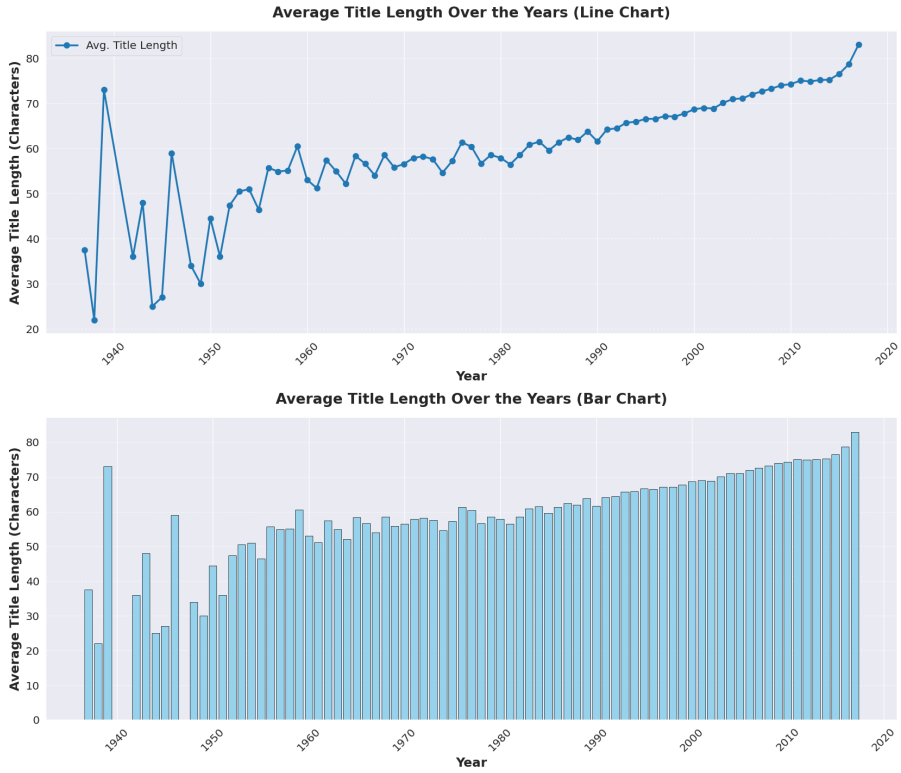



**Figure 7.** Length over the years

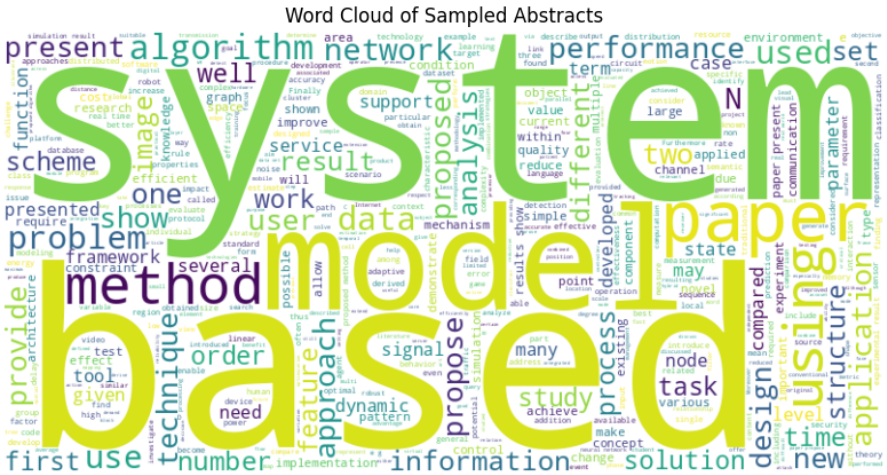



**Figure 8.** Worldcloud

| Metric | Value |
|---|---|
| Pearson Correlation | 0.269 |
| Spearman Correlation | 0.2708 |

Table 3: Correlation Metrics

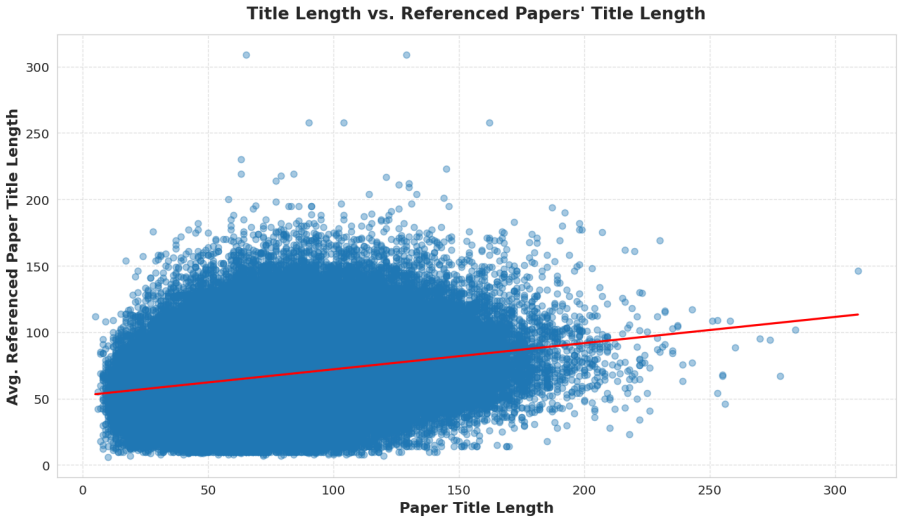



**Figure 9.** Correlation

| # | Author | Publications |
|---|---|---|
| 0 | Wei Wang | 950 |
| 1 | Wei Zhang | 657 |
| 2 | Yang Liu | 629 |
| 3 | Lei Zhang | 579 |
| 4 | Wei Li | 559 |
| 5 | Jun Wang | 544 |
| 6 | Lei Wang | 519 |
| 7 | Lajos Hanzo | 458 |
| 8 | Wei Liu | 456 |
| 9 | Jun Zhang | 455 |

Table 4: Top 10 Authors by Number of Publications

| # | Title | Num References |
|---|---|---|
| 371369 | Comprehensive frequency-dependent substrate noise analysis using boundary element methods | 759 |
| 780292 | Time in Qualitative Simulation. | 561 |
| 104143 | Bibliography on cyclostationarity | 412 |
| 214646 | Fifty Years of MIMO Detection: The Road to Large-Scale MIMOs | 396 |
| 484969 | An Exploration of Enterprise Architecture Research | 394 |
| 223901 | Structure and dynamics of molecular networks: A novel paradigm of drug discovery: A comprehensive review | 386 |
| 302124 | The NP-completeness column: An ongoing guide | 363 |
| 707510 | Digital geometry | 361 |
| 325083 | Deep Learning: Methods and Applications | 343 |
| 538381 | Review: learning Bayesian networks: Approaches and issues | 326 |

Table 5: Top 10 Papers by Number of References

| # | **Title** | **Number of Citations** |
|---|---|---|
| 332760 | Distinctive Image Features from Scale-Invariant Keypoints | 42508 |
| 358174 | LIBSVM: A library for support vector machines | 33016 |
| 716671 | Random Forests | 28679 |
| 18485 | Support-Vector Networks | 26114 |
| 45248 | MapReduce: simplified data processing on large clusters | 24381 |
| 81801 | A fast and elitist multiobjective genetic algorithm: NSGA-II | 24245 |
| 150727 | A theory for multiresolution signal decomposition: the wavelet representation | 24182 |
| 458466 | ImageNet Classification with Deep Convolutional Neural Networks | 22884 |
| 442067 | Histograms of oriented gradients for human detection | 22795 |
| 687881 | Compressed sensing | 20915 |

Table 6: Top 10 Papers by Number of Citations

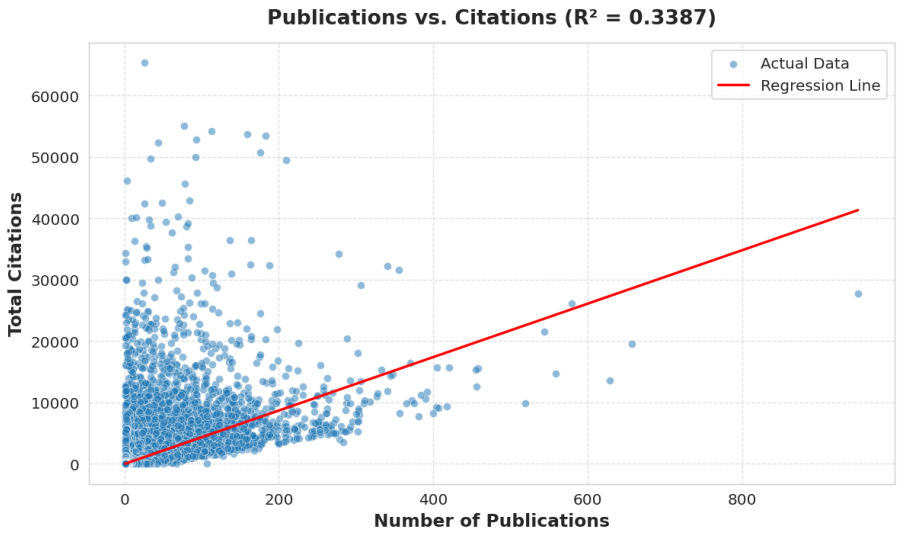



**Figure 10.** Publication vs Citations

## *Network Analysis*

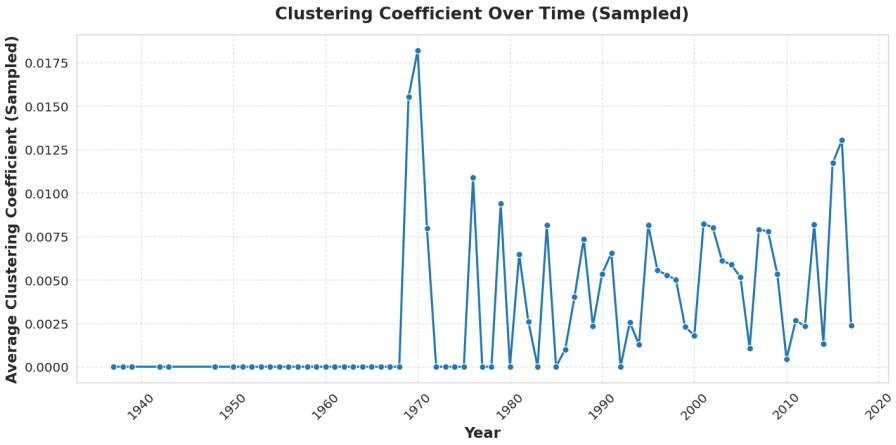



**Figure 11.** clustering coefficient over time

| Metric | Value |
|---|---|
| Average Path Length | 11.288 |
| Diameter | 29 |

Table 7: Network Metrics (Largest Strongly Connected Component)

| Rank | Paper ID | PageRank Score |
|---|---|---|
| 1 | 6d4c5b32-8e13-4022-b67f-1ace7ffc91d0 | 8.7e-05 |
| 2 | ebe93176-7150-4ce6-8d77-880a4e13beb0 | 7e-05 |
| 3 | 34f71039-a18b-4a9c-807d-38cdc9af47db | 6.3e-05 |
| 4 | 0d3b901e-4179-4ed6-badf-7ce4237dadd4 | 5.6e-05 |
| 5 | cc593d94-4a37-4ea1-8f2c-c48dc1afeab4 | 5.6e-05 |
| 6 | 917672b3-f0d5-4f32-8818-aa7f61f36457 | 4.8e-05 |
| 7 | cd41b3d1-69e6-4163-87b9-bf1cb3fb5059 | 4.8e-05 |
| 8 | a1c66cbc-77ad-4167-9787-4936221427ad | 4.7e-05 |
| 9 | 95044198-6db4-4224-a499-d8e8a3e1f52d | 4.6e-05 |
| 10 | 04cfc753-72b6-42c8-8c9f-3a8bc2428c6d | 4.5e-05 |

Table 8: Top 10 Influential Papers by PageRank

| Rank | Title | PageRank Score |
|---|---|---|
| 1 | Bibliography on cyclostationarity | 8.7e-05 |
| 2 | An Exploration of Enterprise Architecture Research | 7e-05 |
| 3 | On the Role and the Importance of Features for Background Modeling and Foreground Detection | 6.3e-05 |
| 4 | Digital geometry | 5.6e-05 |
| 5 | An Updated ERP Systems Annotated Bibliography: 2001-2005 | 5.6e-05 |
| 6 | Twelve years of diagrams research | 4.8e-05 |
| 7 | A Survey on Information Visualization for Network and Service Management | 4.8e-05 |
| 8 | Time in Qualitative Simulation. | 4.7e-05 |
| 9 | Structure and dynamics of molecular networks: A novel paradigm of drug discovery: A comprehensive review | 4.6e-05 |
| 10 | Research Issues in Smart Vehicles and Elderly Drivers: A Literature Review | 4.5e-05 |

Table 9: Top 10 Influential Papers by PageRank

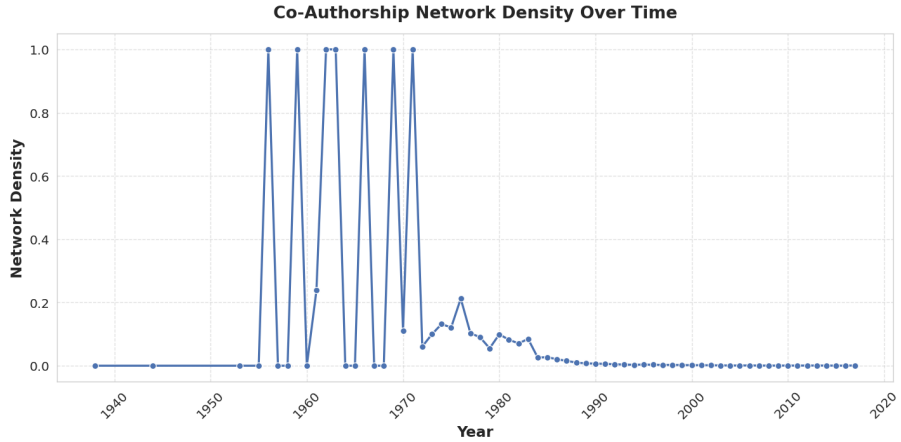



**Figure 12.** Co-Authorship Network Density Over Time

| Degree Author | Degree | Betweenness Author | Betweenness | Closeness Author | Closeness |
|---|---|---|---|---|---|
| Zhen Zhang | 0.0012 | Wei Li | 0.0092 | Wei Wang | 0.0712 |
| Henning Hermjakob | 0.0011 | Wei Wang | 0.0092 | Peng Zhang | 0.0696 |
| S. Avino | 0.0011 | Jun Kong | 0.0090 | Peng Wang | 0.0694 |
| F. Acernese | 0.0010 | Jaap Kamps | 0.0087 | Jing Wang | 0.0687 |
| P. Amico | 0.0010 | Peng Wang | 0.0080 | Yang Liu | 0.0686 |
| M. Alshourbagy | 0.0010 | Fan Zhang | 0.0078 | Jun Wang | 0.0678 |
| S. Aoudia | 0.0010 | Djoerd Hiemstra | 0.0072 | Yan Zhang | 0.0677 |
| D. Babusci | 0.0010 | Ke Zhou | 0.0072 | Bin Liu | 0.0676 |
| G. Ballardin | 0.0010 | Lin Chen | 0.0071 | Wei Li | 0.0674 |
| R. Barille | 0.0010 | Haizhou Li | 0.0070 | Jian Wang | 0.0671 |

Table 10: Top 10 Authors by Centrality Measures (Optimized)

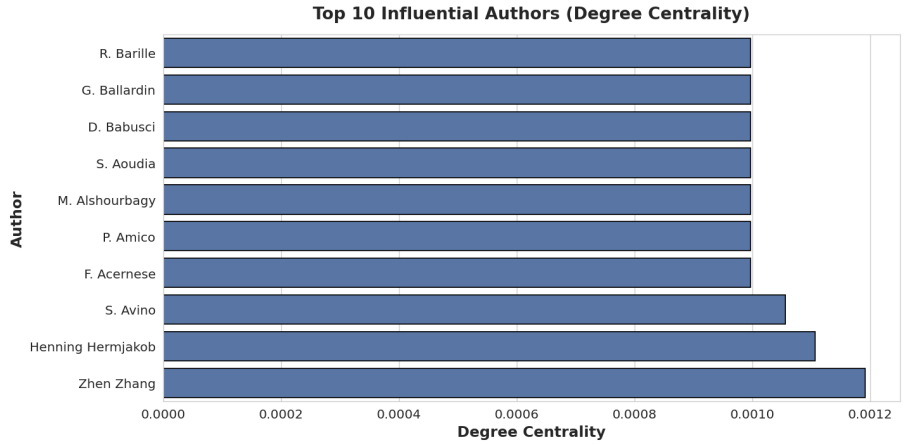



**Figure 13.** Top 10 Influential Authors (Degree Centrality

| Community ID | Number of Authors |
|---|---|
| 6 | 1670 |
| 32 | 1530 |
| 47 | 1201 |
| 313 | 1079 |
| 177 | 979 |

Table 11: Top 5 Largest Author Communities

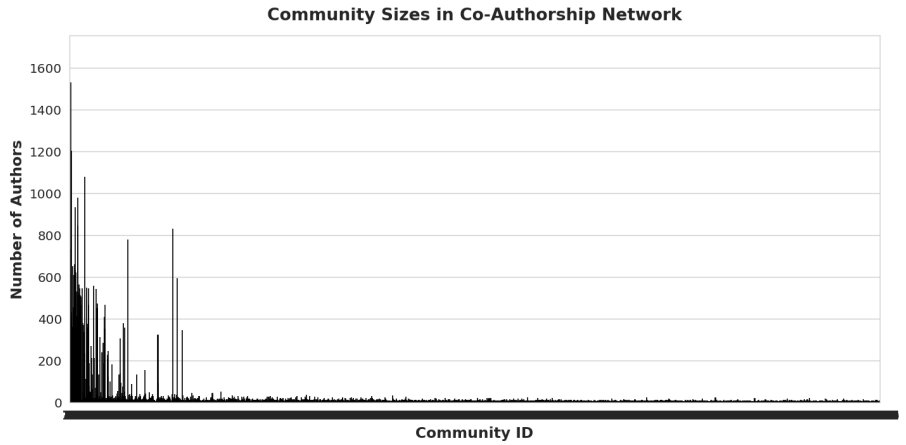



**Figure 14.** Community Sizes in Co-Authorship Network

| Metric | Value |
|---|---|
| Total Nodes | 942 |
| Total Edges | 1747 |
| Unique Venues | 2253 |

Table 12: Venue Network Summary

| **Degree Venue** | **Degree Centrality** | **Betweenness Venue** | **Betweenness Centrality** |
|---|---|---|---|
| IEEE Transactions on Pattern Analysis and Machine Intelligence | 0.0446 | neural information processing systems | 0.0336 |
| neural information processing systems | 0.0436 | IEEE Transactions on Information Theory | 0.0323 |
| IEEE Transactions on Communications | 0.0393 | IEEE Transactions on Computers | 0.0306 |
| IEEE Journal on Selected Areas in Communications | 0.0383 | computer vision and pattern recognition | 0.0255 |
| computer vision and pattern recognition | 0.0372 | international conference on acoustics, speech, and signal processing | 0.0251 |
| Journal of Machine Learning Research | 0.0329 | IEEE Transactions on Signal Processing | 0.0235 |
| international conference on robotics and automation | 0.0329 | international conference on computer communications | 0.0207 |
| IEEE Transactions on Information Theory | 0.0319 | systems man and cybernetics | 0.0193 |
| IEEE Transactions on Signal Processing | 0.0319 | IEEE Transactions on Communications | 0.0182 |
| IEEE Transactions on Software Engineering | 0.0308 | IEEE Transactions on Software Engineering | 0.0180 |

Table 13: Top 10 Venues by Centrality

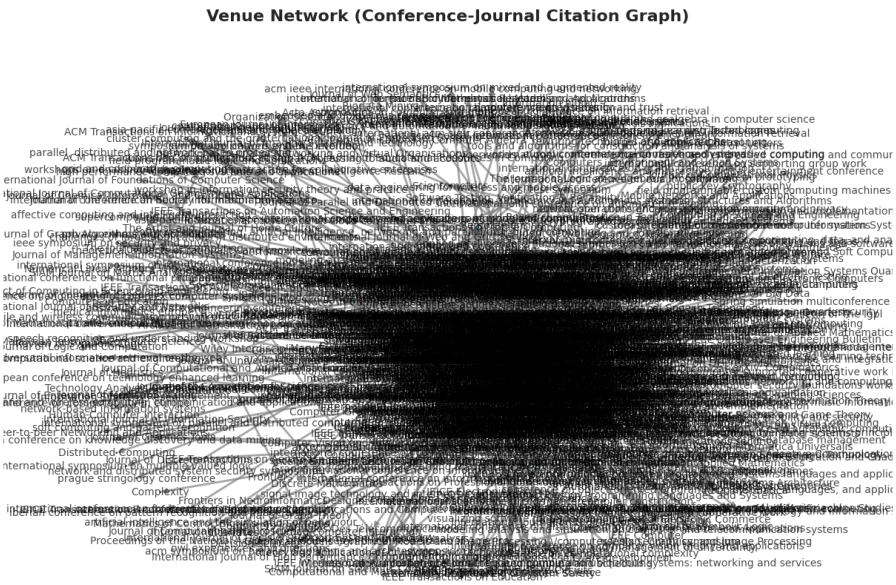



**Figure 15.** Venue Network (Conference-Journal Citation Graph)

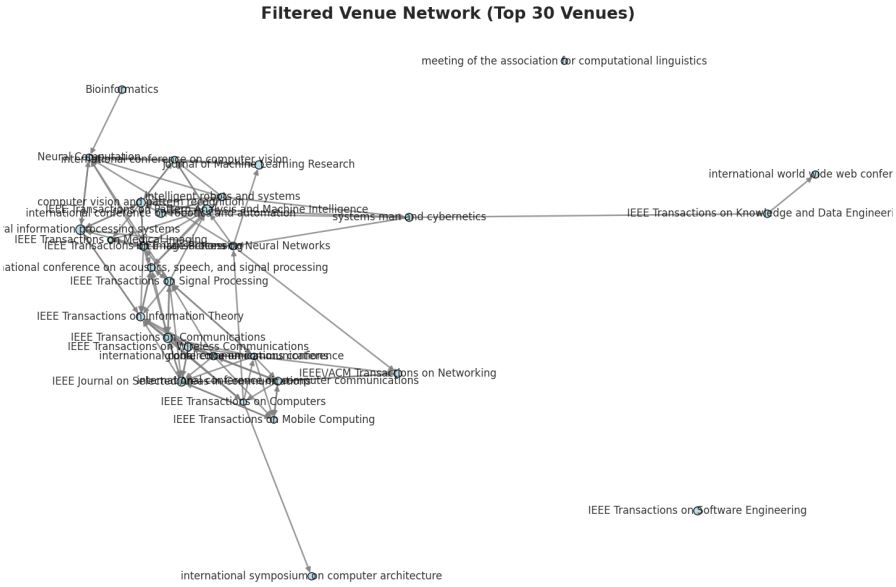



**Figure 16.** Filtered Venue Network (Top 30 Venues)

The output for some parts was too long to be in the report file of the project. so you can see them in the notebook file.

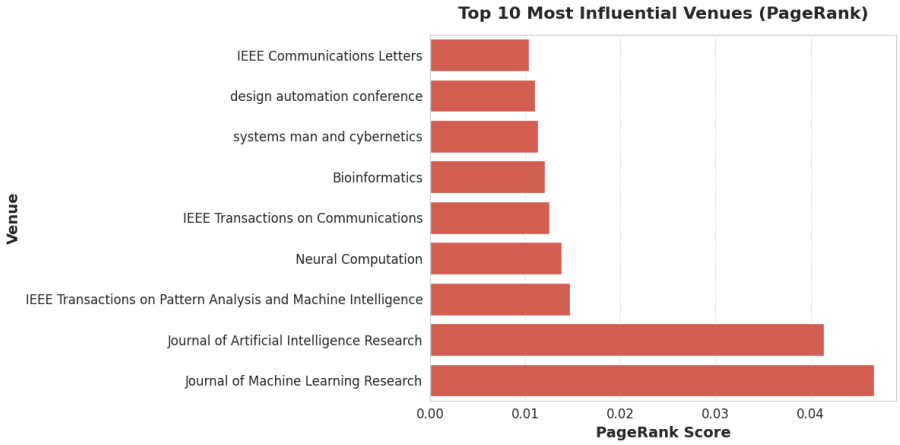



**Figure 17.** Top 10 Most Influential Venues (PageRank)

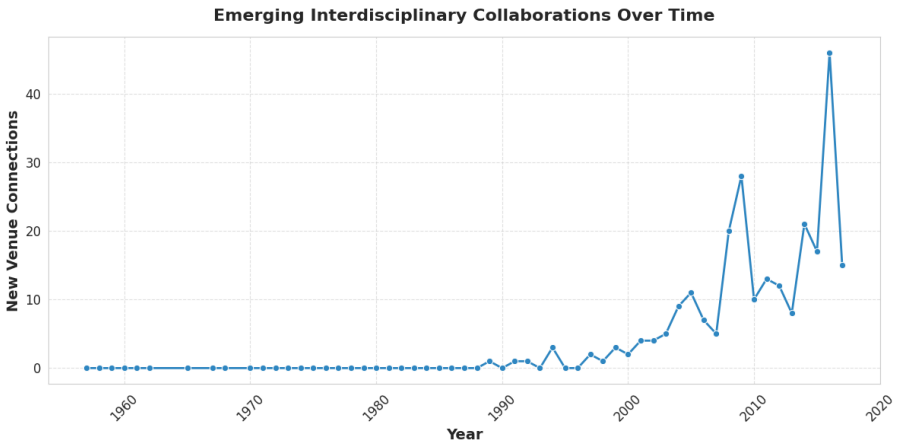



**Figure 18.** Emerging Interdisciplinary Collaborations Over Time

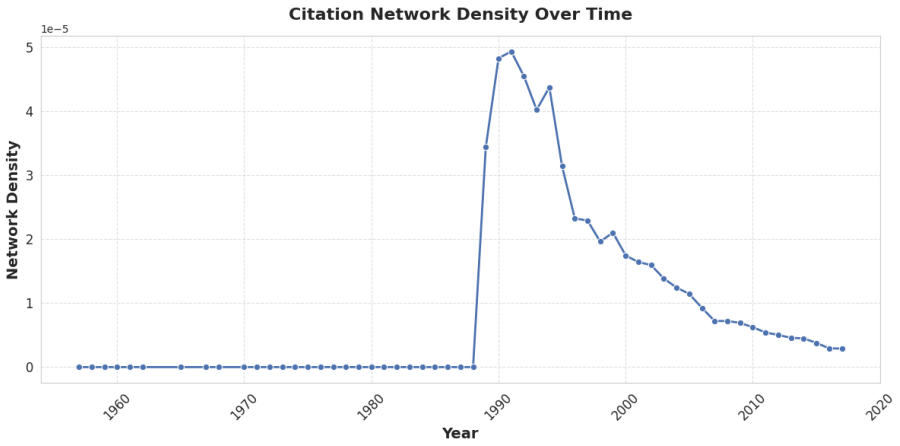



**Figure 19.** Citation Network Density Over Time

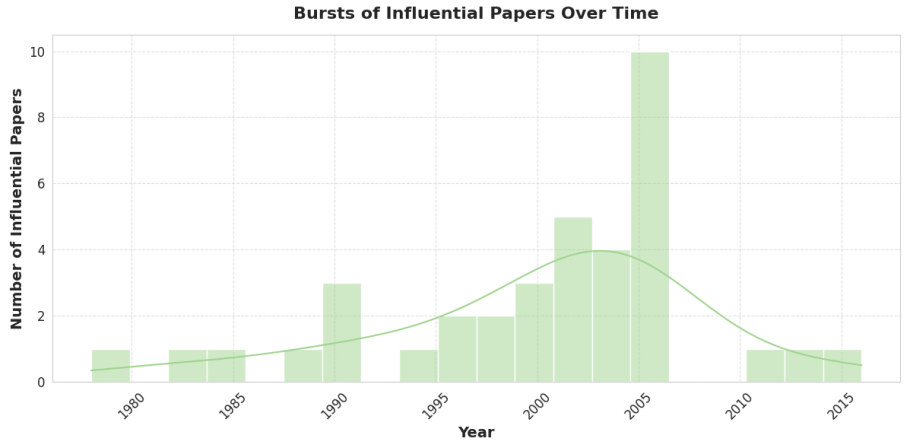



**Figure 20.** Bursts of Influential Papers Over Time

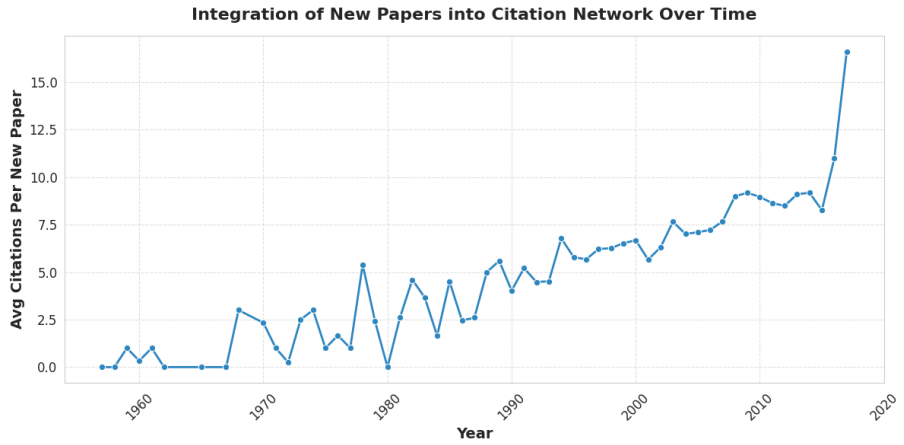



**Figure 21.** Integration of New Papers into Citation Network Over Time

In this report, we analyze how these challenges affected our model and to what extent we were able to overcome them.

## Challenge 1: Time Gap Between Training and Test Set

One of the major hurdles in citation prediction is that the training dataset (DBLP) consists of papers published in earlier years, while the test dataset (our crawled data) contains **newer papers**. This creates a challenge because citation patterns change over time.

### *Why Does This Matter?*

- Older papers in the training set follow a **different citation pattern** than recent papers.
- Recent papers may have **fewer citations** because they have not existed long enough to accumulate many citations.

- **Emerging fields** such as Generative AI and Large Language Models (LLMs) might gain citations much faster than traditional research areas.

### *How Did Our Model Handle This?*

**Overcoming the Challenge:**

- We applied a **log transformation** to citation counts, which helped normalize differences in citation growth across time periods.
- Our **Graph Neural Network (GNN)** leveraged co-authorship networks, which often remain stable over time.
- **FAISS-based document retrieval** helped find similar works based on paper embeddings, rather than relying solely on historical citation data.

**Where It Struggled:**

- The model often underestimated citations for **groundbreaking recent papers**, as similar high-impact papers were absent in the training set.
- Prediction errors were higher for **papers published in the last 1-2 years**, as they had limited citation history.

## Challenge 2: New Concepts and Authors

Another challenge is that research is constantly evolving, with **new topics, methodologies, and authors** emerging over time. Since our training set (DBLP) does not contain data on these new topics, generalization to the test set (crawled data) becomes difficult.

### *Why Does This Matter?*

- New researchers who have never appeared in the training dataset lack a **co-authorship history**, reducing the effectiveness of GNN-based predictions.
- Cutting-edge topics (e.g., Diffusion Models, Generative AI) were not present in the older training dataset, making it hard for **FAISS-based retrieval** to find similar works.
- **Keyword-based feature extraction** using KeyBERT struggled to understand **entirely new terminology** that did not exist in earlier datasets.

### *How Did Our Model Handle This?*

**Overcoming the Challenge:**

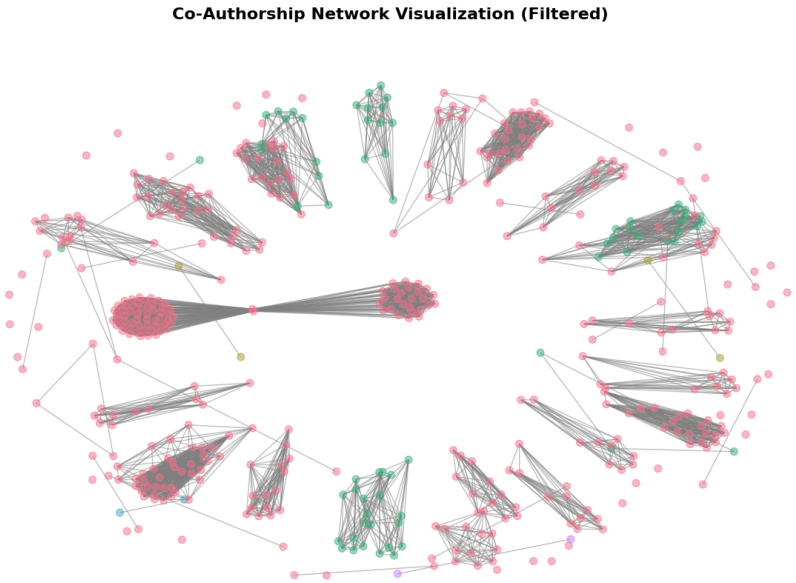
- **FAISS Vector Search:** Instead of only relying on structured citation data, we used **semantic similarity search** to find the most relevant papers.
- **Keyword Extraction (KeyBERT):** Despite concept drift, our system was able to extract meaningful keywords from new research topics.
- **Graph-based Features:** Even when encountering new authors, our GNN was still able to infer their potential impact based on collaborations.

**Where It Struggled:**

- **Completely new research areas** (e.g., Large Vision Models) had no close matches in the training dataset, leading to **higher prediction errors**.
- **Newly published authors** without prior citations had **no co-authorship network**, making their citation predictions difficult.
- **NLP-based models struggled** to assess research importance for completely new fields with no historical data.

| Metric | Value |
|---|---|
| Number of Authors (Nodes) | 14820 |
| Number of Co-Authorships (Edges) | 26409 |

Table 14: Co-Authorship Network Summary



| Metric | Score |
|---|---|
| Davies-Bouldin Index | 4.97 |
| Silhouette Score | 0.0285 |

Table 15: Clustering Metrics

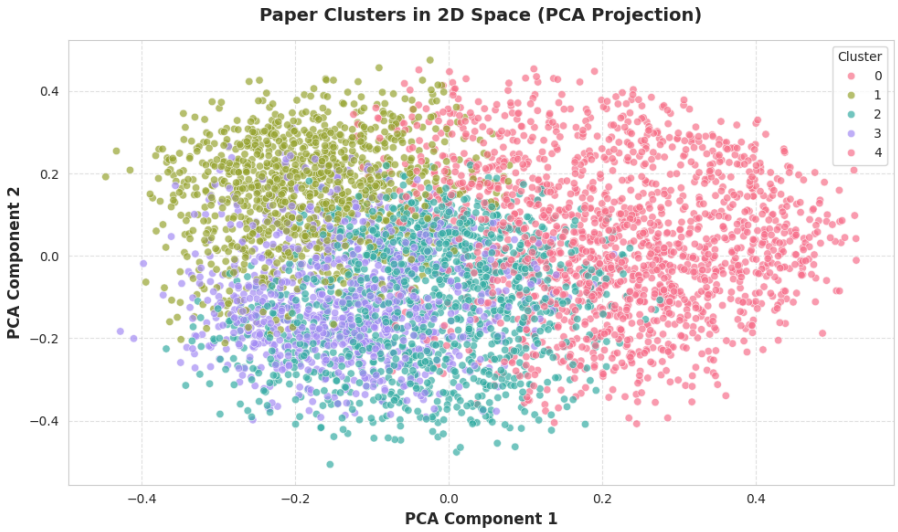



**Figure 22.** Enter Caption

| Metric | Value |
|---|---|
| Validation MAE | 39.4089 |
| Validation RMSE | 116.943 |
| Validation $R^2$ | -0.0511863 |

Table 16: Validation Set Metrics

| Metric | Value |
|---|---|
| Crawled MAE | 59.6743 |
| Crawled RMSE | 362.777 |
| Crawled $R^2$ | -0.021728 |

Table 17: Crawled Dataset Metrics

| # | Title | Predicted Citations |
|---|---|---|
| 0 | A Perspective on the Theoretical Foundation of Dual Process Models | 22.7704 |
| 1 | Business Model Innovation for Sustainability: Towards a Unified Perspective for Creation of Sustainable Business Models | 14.8118 |
| 2 | Vertical load-carrying behavior and design models for micropiles considering foundation configuration conditions | 7.01206 |
| 3 | Development of rapid three-dimensional finite-element based rigid airfield pavement foundation response and moduli prediction models | 6.02443 |
| 4 | Measurements and models of electric fields in the in vivo human brain during transcranial electric stimulation | 19.9301 |

Table 18: Top 5 Predicted Citations from Crawled Dataset

| Metric | Value |
|---|---|
| Mean Absolute Error (MAE) | 47.149 |
| Root Mean Squared Error (RMSE) | 215.64 |
| $R^2$ Score | -0.0240036 |

Table 19: GNN Model Evaluation Metrics

| Author | Predicted Citations |
|---|---|
| Saman Vaisipour | 48.4352 |
| Jean-Philippe Pernot | 48.3698 |
| Sang-Su Lee | 48.3001 |
| Mohammad Esmaeilpour | 47.5464 |
| Guilherme Maia | 47.5295 |

Table 20: Top 5 Predicted Most Cited Authors

| # | Author | Predicted Citations |
|---|---|---|
| 14468 | Shuonan Wang | 239.344 |
| 94788 | Ju H. Park | 225.841 |
| 88373 | P. Mukhopadhyay | 222.032 |
| 91596 | B. Subramanyam | 191.763 |
| 90495 | S. Parviainen | 176.357 |

Table 21: Top 5 Predicted Most Cited Authors (Crawled Data)

Outputs such as LLM text generated and RAG examples and research assistant are in the notebook file.