Data Science I Project Report

by: Constance Streitman (ID: 2253221)

ABSTRACT

This project investigates the evolution of CS research by analyzing the DBLP dataset, which contains metadata for over 2.5 million academic papers. My objective was to quantitatively map research trends, identify citation anomalies, and uncover the structure of research communities. The methodology employed TF-IDF vectorization for feature extraction, log-transformed Z-scores for anomaly detection, and spectral clustering on co-authorship matrices.

Key findings include the detection of "unprestigious high-impact" outliers, where papers utilizing specific, rigorous technical terminology (e.g., "network coding," "linear systems") significantly outperformed their venue's baseline compared to papers using generic buzzwords (such as “machine learning”). Temporal topic analysis quantitatively visualized the paradigm shift in artificial intelligence, marking the sharp decline of support vector machines (SVM) and the simultaneous rise of deep learning following 2012. Finally, author network analysis revealed a massive "giant component" of interconnected researchers, alongside distinct, isolated communities in specialized fields such as cloud computing. Altogether, this paper’s task results provide a data-driven perspective on the dynamic nature of academic influence and collaboration in CS.

DATA DESCRIPTION

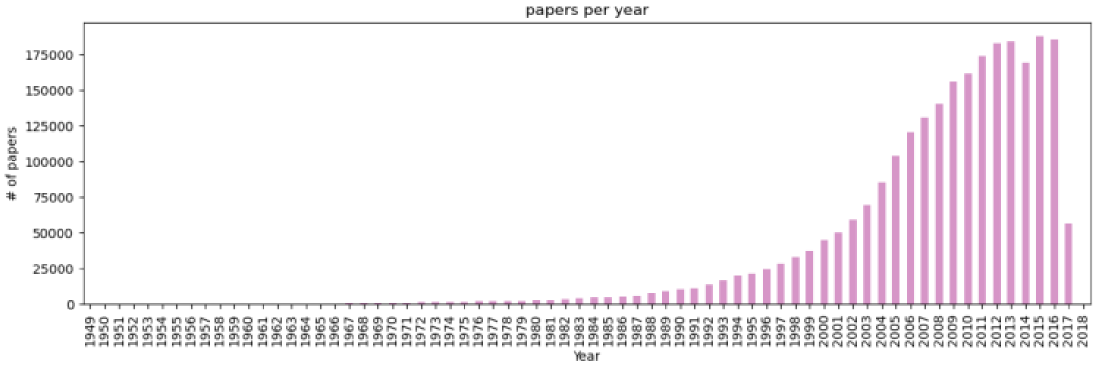
I, of course, used the DBLP dataset, and to save time and space I won’t bother listing its characteristics as that’s already known for this project. I would, however, like to quickly summarize how I wrangled the data. First, I merged the different json files into a single dataframe using the *sorted* and *concat* functions in *pandas*. I eliminated the ID column for redundancy. Given the vastness of the dataset, I also considered it safe to remove any samples that lacked a title or abstract.

I also had a problem where some author and reference entries were lists, but some were strings; as a solution, I applied a custom function to standardize these into clean lists of strings, ensuring consistent iteration for the network analysis.

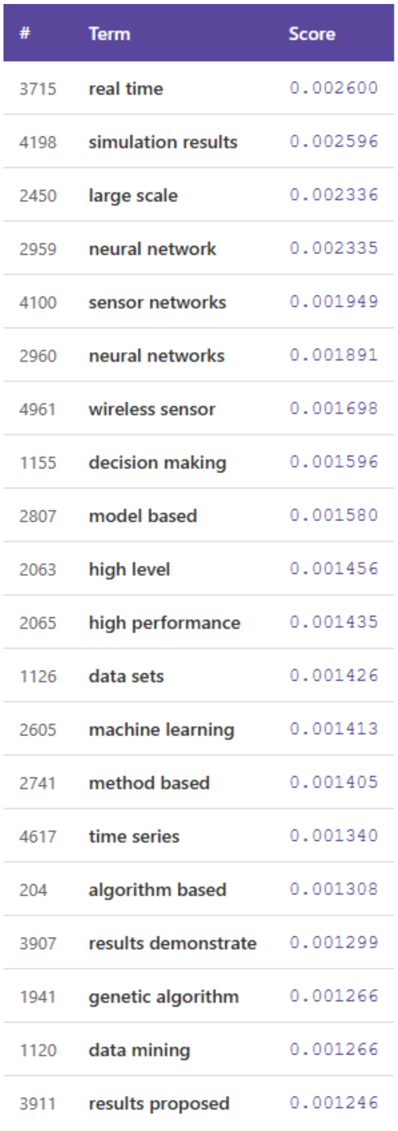
After that, I created a unified *text* column by concatenating the title and abstract columns and normalizing them to lowercase. This served as the primary input for the TF-IDF vectorization. Other data was wrangled, but it was task-specific, and will be elaborated upon as appropriate below.

TASK #1: EXPLORATORY DATA ANALYSIS

I began by analyzing the temporal distribution of publications (*year*) to understand the dataset's growth trajectory. I also examined the distribution of *n_citations* across different venues just to see if anything popped out to me. Nothing did, and it was irrelevant later anyway.

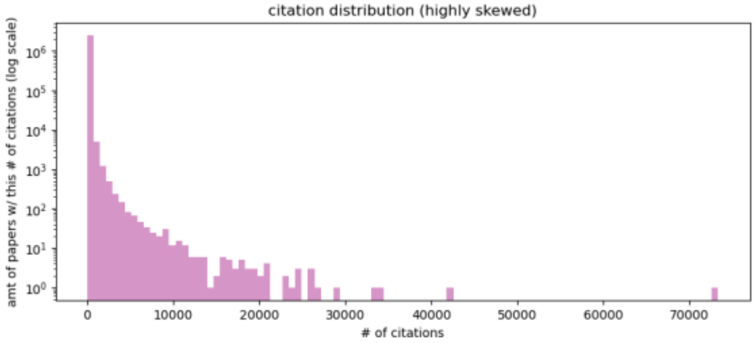


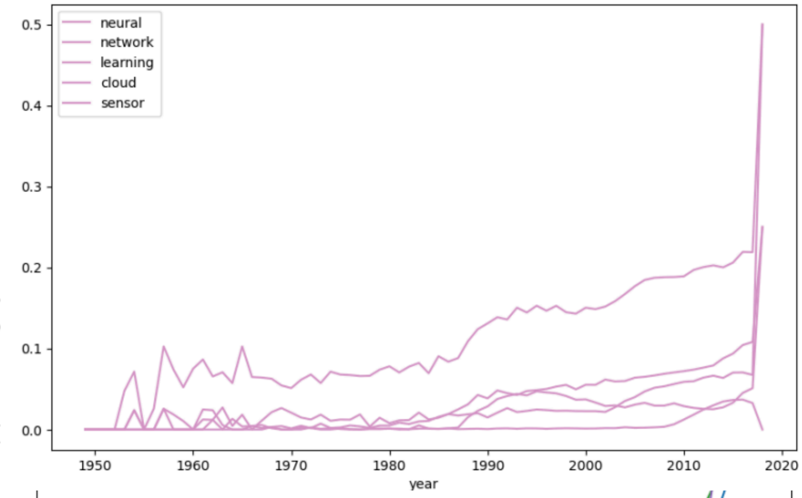
After that, I decided to go ahead and do the TF-IDF vectorization for the dataset. The first time I ran it, the words were entirely useless, with terms like "based" and "paper" coming out on top. To fix this, I made a list of boiler-plate phrases, aptly named *boiler_phrases*, to overcome this issue and get more useful results. However, I was unable to add the list to the *stop_words* parameter inside the *TfidfVectorizer* function, so did the correction after the fact. Also, in the name of getting a more apt output, I changed it from a 1-term requirement to a 2-term one, allowing phrases like "neural network" and "genetic algorithm" to show up. Also, my first try at doing the TF-IDF vectorizer took over 2 hours and still wasn't completed, so I opted to take a random subset using seed 2253221.

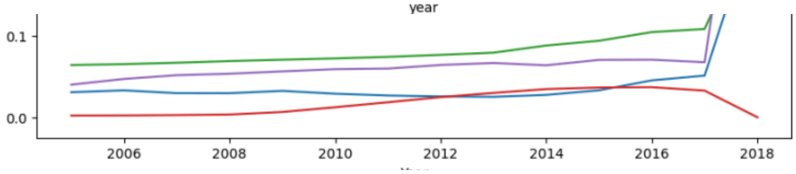


#	Term	Score
3715	real time	0.002600
4198	simulation results	0.002596
2450	large scale	0.002336
2959	neural network	0.002335
4100	sensor networks	0.001949
2960	neural networks	0.001891
4961	wireless sensor	0.001698
1155	decision making	0.001596
2807	model based	0.001580
2063	high level	0.001456
2065	high performance	0.001435
1126	data sets	0.001426
2605	machine learning	0.001413
2741	method based	0.001405
4617	time series	0.001340
204	algorithm based	0.001308
3907	results demonstrate	0.001299
1941	genetic algorithm	0.001266
1120	data mining	0.001266
3911	results proposed	0.001246

After that, I made a quick histogram of citations per paper, which was so highly skewed that the y-variable (# papers) had to become a log scale.



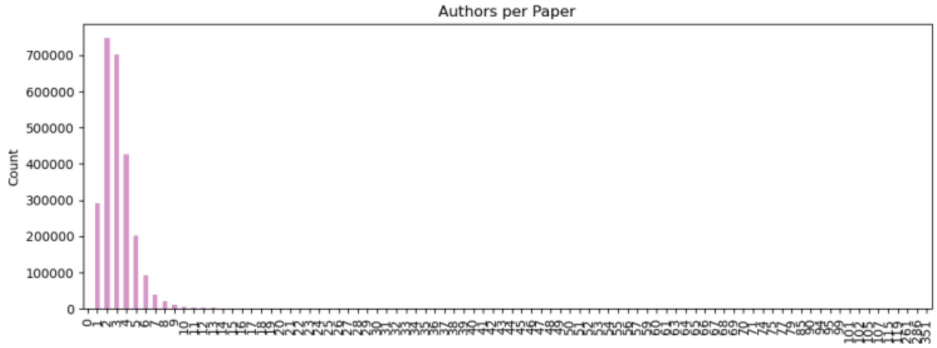




I then made a scatterplot comparing the amount of papers published from a venue to the average amount of citations a paper from that venue would get, but the outliers flattened the graph so much it was basically useless. More usefully, I plotted a time-series graph tracking key hot-button terms over time (left, pink).

I then refined the time series to only after 2005 to try and get a better gauge on things. Curiously, the lines all got a lot flatter before shooting up in 2018. The dataset for 2018 is rather empty, so I'm unaware as to whether the upshoot in the terms (excluding cloud) are due to genuine research explosions or just a limited dataset biasing the graph. Due to the "papers per year" histogram above, I'm fairly certain it has nothing to do with my taking of a random subset, especially since the subset still had n=200,000.

Additionally, in the name of exploration, I made a "authors per paper" histogram, which is clearly heavily skewed, and then also a list of the authors of the top 20 authors by paper count.



Everybody except one was Chinese, which I assume is because many Chinese people have the same name in the Latin alphabet and not because of a plethora of extremely prolific Chinese researchers. The lone exception is a man named Lajos Hanzo, co-author of 911 papers; he was the editor-in-chief of IEEE Press, so I assume it was via that position that he got his name on so many papers.

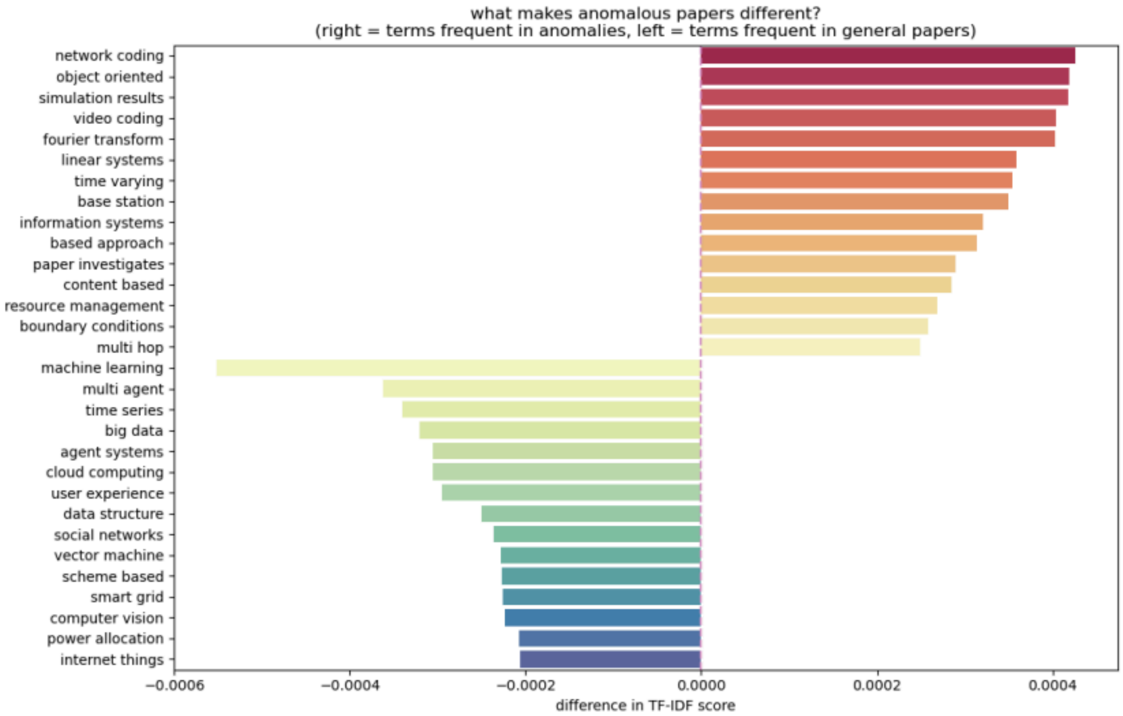
TASK #2: CITATION ANOMALY DETECTION

My objective here was to identify outlier papers i.e. papers in typically high-citation ("prestigious") venues with anomalously low citation counts, and papers in typically low-citation ("unprestigious") venues with anomalously high citation counts. However, as noted from the above EDA, the distribution of citations per paper is heavily skewed. To make it usable, I applied the transformation: *x′ = log(1 + x)* to the dataset in order to normalize it.

I then categorized the venues into prestigious and unprestigious bins—the "prestigious" venues being ones in the 85th percentile and above by metric of average citations per paper, and the "unprestigious" venues comprising everything under the 50th percentile. Harsh, but true. Then I calculated the Z-score for each paper by comparing its citations to the mean and standard deviation of its venue's citations. Then, after that, I made a new set out of all the papers with |Z| > 2.5, the idea being that 2.5 was >99th percentile and somewhat of a "round" number.

I got a decent-sized dataframe, but it looks incredibly ugly, so I won't include it. I did find a few problems with my outputs—for one, there were several "prestigious venue but low citation" papers with a whopping 0 citations, which I suspect may be an input error or have something to do with the nature of how it was published rather than truly being a normal paper in a prestigious journal that was just a massive failure.

After that, to understand why papers became anomalies, I compared the TF-IDF term vectors of the anomalous papers against the general population. I then calculated the difference in mean TF-IDF scores to identify terms unique to outliers versus generic terms Here is it plotted:

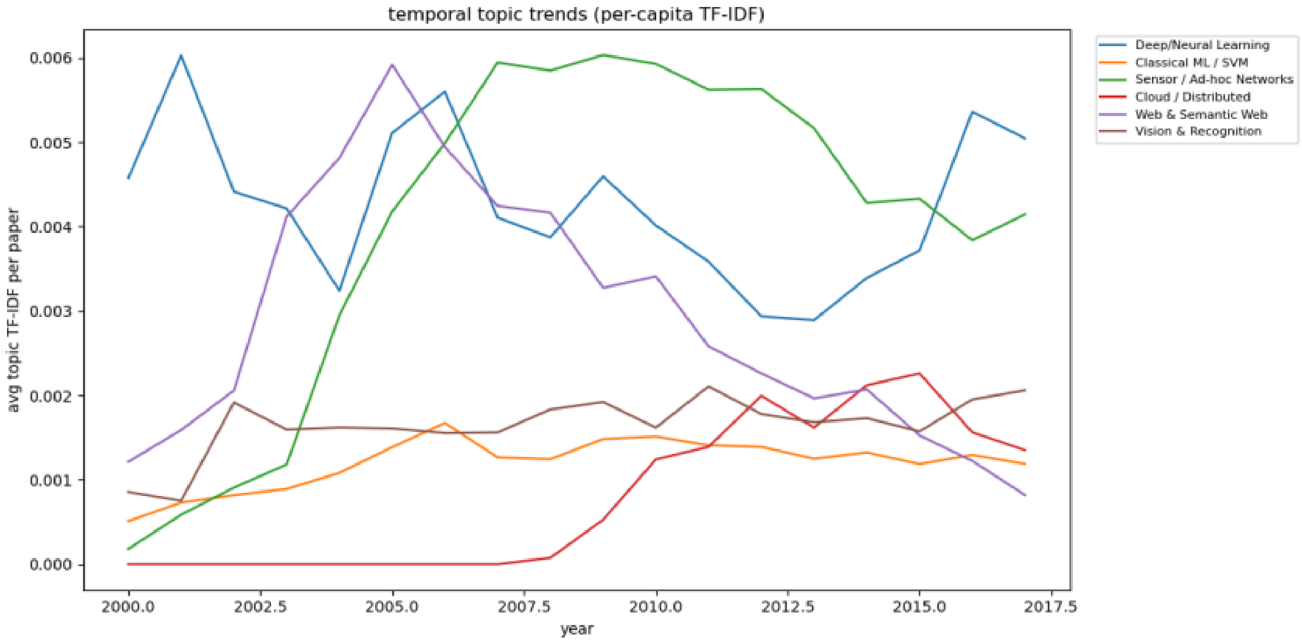


At first, “machine learning” being non-anomalous was both surprising and unsurprising to me. I suppose that since it’s a hot topic, all journals are picking it up and familiar with the nuances of it, meaning it’s far less likely to be “misplaced” into a journal of the wrong caliber compared to more theoretical fields like fourier transforms. As for the other anomalous topics, I have no explanations or answers, but it’s a step up to be aware in the first place that “information systems” and “simulation results” are indeed anomalous.

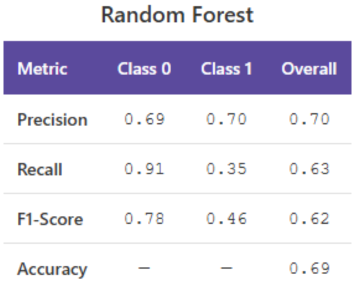
TASK #3: TEMPORAL TOPIC ANALYSIS

For this task, my goal was to track the "rise and fall" of specific research topics and keywords over time by analyzing the abstract and title fields. Due to the computational cost of vectorizing millions of text fields, I only sampled a random subset of 200,000 papers published after 1990 to focus on modern trends.

Instead of doing unsupervised categorical modeling (like LDA), I employed a somewhat manual approach. I defined distinct topic dictionaries (e.g. mapping "convolutional neural" and "deep learning" to the topic "Deep/Neural Learning") to track specific research trends accurately. In all, I had 12 main categories—not all papers made it, if they didn’t include the terms, but over 20k papers from my 200k subset still did. For each year and topic, I calculated a normalized score: the sum of TF-IDF weights for the topic's keywords, divided by the total number of papers published that year. As such, it’s a per-capita metric, and accounts for the exponential growth in total paper volume. Plotted below is the time-series graph, though only the top 6 topics by metric of TF-IDF are featured:

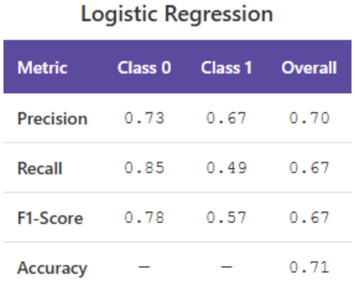


There’s a large rise in "deep/neural learning" starting circa 2012, which aligns with the AlexNet moment that catalyzed the modern deep learning revolution. Prior to this inflection point, the field was relatively dormant, as shown in the graph. "Sensor / ad-hoc networks" shows perhaps the most dramatic pattern, with a sharp rise beginning around 2004 and peaking for several years before declining. This trajectory likely reflects the hype around wireless sensor networks, which subsequently matured and became less cutting-edge as these systems became more commonplace. "Cloud / distributed" computing was entirely absent until 2008 or so, coinciding with the commercial launch of major cloud platforms like AWS and Azure. However, its growth appears more modest and plateaus relatively quickly, suggesting that while important, cloud computing, like sensors, was no longer a glamorous research focus. "Web & semantic web" peaks earlier than the others in ~2005, representing the tail end of the semantic web vision that dominated early 2000s research. Its subsequent decline suggests that this particular vision never fully materialized, being largely supplanted by statistical and learning-based approaches. "Vision & recognition" maintains a relatively stable presence throughout the timeline, with modest growth. I find this a bit surprising, considering how important it is in modern AI, but I suppose that maybe the research in that post-dates when this dataset ends, i.e. that ML-based image recognition only takes off post-2018.



Random Forest

Metric	Class 0	Class 1	Overall
Precision	0.69	0.70	0.70
Recall	0.91	0.35	0.63
F1-Score	0.78	0.46	0.62
Accuracy	–	–	0.69



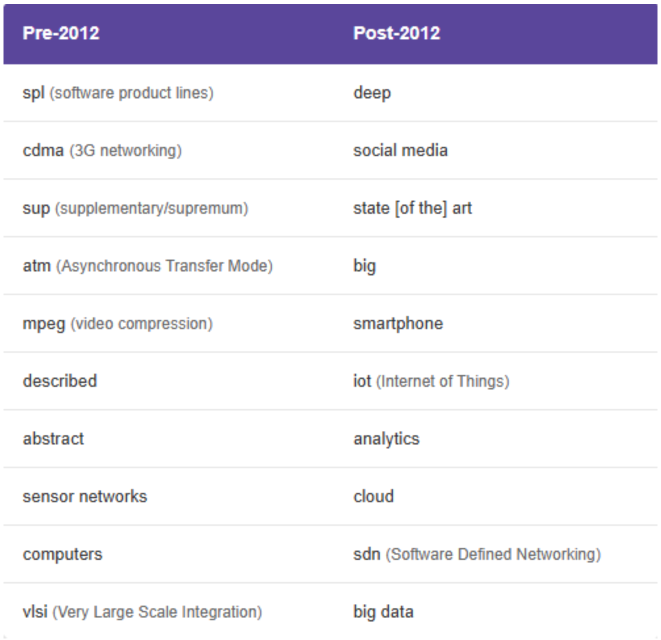
Logistic Regression

Metric	Class 0	Class 1	Overall
Precision	0.73	0.67	0.70
Recall	0.85	0.49	0.67
F1-Score	0.78	0.57	0.67
Accuracy	–	–	0.71

Relatedly but perhaps less rigorously, I tried using classification to determine if the vocabulary of computer science research has changed distinctively enough after 2012, such that a ML algorithm could accurately classify a paper as having been published before or after 2012 based solely on its abstract+title text. Given 2012 from the above graph was the start of the deep learning explosion, I wanted to see how large the impact truly was.

I used my existing TF-IDF matrix and made a logistic regression with 0 representing pre-2012 papers, and 1 post-2012 papers. The idea was that since its coefficients were interpretable, I could take the words with the largest coefficients as indicators of time period. I also ran a random forest, but got lower accuracy and recall, so I decided to stick with the logistic regression for my analysis (model stats above).

When I first ran the regression, I immediately ran into a problem similar to earlier; many of the top words were years. Of course a paper with “2013” in it would be published after 2012, right? So I blacklisted all years from the vocabulary pool and tried again, with the table above showing the top time-period-divergent words. I trained both old and new models on an 80/20 train-test split, and my accuracy only dropped from 71.4% to 70.9%, which I don't think is half bad in either case. As for the words themselves—well, it's no surprise that "deep" and "social media" top the more modern subset of papers, is all I have to say.



Pre-2012	Post-2012
spl (software product lines)	deep
cdma (3G networking)	social media
sup (supplementary/supremum)	state [of the] art
atm (Asynchronous Transfer Mode)	big
mpeg (video compression)	smartphone
described	iot (Internet of Things)
abstract	analytics
sensor networks	cloud
computers	sdn (Software Defined Networking)
vlsi (Very Large Scale Integration)	big data

Now, to figure out why this model struggles with precision, I made a case study out of Index 118886, which was a false positive. The model predicted the paper to be post-2012, but it was actually published in 2010. Although the model was technically incorrect regarding the year, the linguistic cues strongly align with post-2012 trends. The text contains high-signal keywords such as "data", "privacy", and "electronic medical records", which are all themes of the post-2012 data science boom. The model failed not because it was confused, but because the paper was a semantic precursor—it discussed modern topics just two years before the 2012 cutoff.

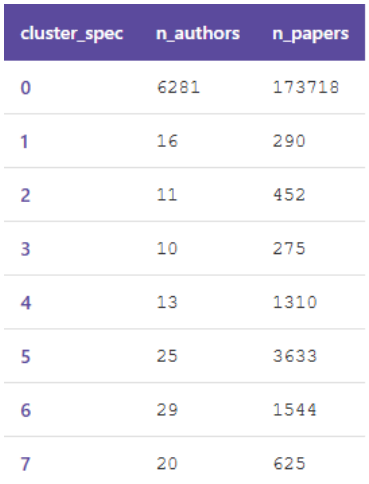
TASK #4: AUTHOR COMMUNITY CLUSTERING

For this task, I grouped authors into distinct research communities based on their co-authorship patterns, and thereafter assigned the clusters into subfields by content of their publications.

First, I filtered the dataset to identify "core" authors using these criteria: at least 30 papers, at least 5 distinct coauthors, a career span of at least 7 years, and at least 5,000 total citations. This resulted in 6,424 core authors out of the 1.5+ million unique authors in the dataset. Since the co-author clustering requires a matrix, it needed a small pool, and this was my way of minimizing said pool while still getting worthwhile data out of it. I wanted n<10,000 and figured that requiring 5,000 total citations wasn't too unreasonable, and nor was 30 total papers, but was weary to strengthen the standards; luckily, this was enough to get the desired low n.

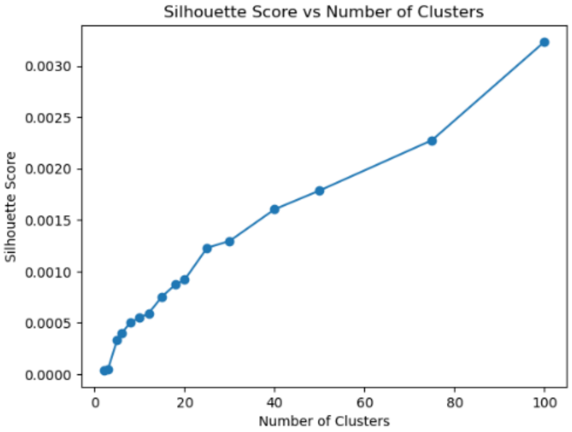
I then constructed a co-authorship matrix where each cell (i,j) represents the number of papers authors i and j have written together. To build this, I iterated through all papers containing at least two core authors, incrementing the matrix entry for each pair of coauthors on that paper. This produced a 6,424 x 6,424 sparse matrix, which I then filtered to remove authors with zero connections, leaving 6,405 active authors.

For clustering, I applied two complementary algorithms: hierarchical and spectral clustering. I set both to produce 8 clusters, though this parameter could be adjusted and was honestly arbitrary. The hierarchical clustering produced a disappointing result: one massive cluster containing 6,397 of the 6,405 authors, with only seven tiny clusters containing 1-2 authors each. Total dud, essentially, useless for any type of analysis and probably just simply an unsuitable model for the task if it truly had such tiny non-dominant clusters. I quickly moved on to spectral instead of attempting to make a mountain out of a molehill with the hierarchical method.



cluster_spec	n_authors	n_papers
0	6281	173718
1	16	290
2	11	452
3	10	275
4	13	1310
5	25	3633
6	29	1544
7	20	625

Spectral clustering performed better, yielding more balanced clusters, though still with one dominant cluster of 6,281 authors. The smaller clusters ranged from 10 to 29 authors, suggesting these represent more isolated or specialized research communities. To characterize each cluster, I collected all papers written by authors in that cluster, then calculated the fraction of papers mentioning keywords from each of my 12 topic categories from the topic dictionary from Task #3, with the top three topics providing a research profile for each cluster. The results revealed that the giant mega-cluster covers all topics fairly evenly, with slight emphasis on "sensor / ad-hoc networks"—essentially, it's the mainstream of CS research. The smaller spectral clusters, however, showed more distinctive profiles, with Cluster 1 specializing in "web & semantic web" and "NLP / Text", Cluster 5 emphasizing data mining, and Cluster 7 concentrated on classical ML methods. I posited initially that they likely represented tightly-knit communities working in more niche areas, or perhaps research groups from specific institutions that collaborate primarily within their own network.



But, to test everything out and be sure, I tried to find out what an optimal cluster count might be, and ran the model again and again with different cluster counts. Surprisingly, there seems to be no upper bound where the improvement in silhouette score begins to decrease. What this plot is really telling us is that the entire network is a continuum of collaboration dominated by that one enormous, interconnected web. As I increase the cluster count, the algorithm is just continuously slicing off smaller, better-defined little sub-communities from the main cluster, and will continue to do so unless I were to apply more advanced algorithms beyond the scope of this course.

LIMITATIONS

The most glaring limitation affecting this report is the reduced scope and possible bias brought in by taking a subset for the TF-IDF vectorizer function. Additionally, the 2018 dataset was so incomplete I had to scrap it for some parts. If I had far more RAM and updated data to 2024 or so to include the AI boom, I could probably garner much more useful insights for every task.

CONCLUSION

My investigation into the DBLP dataset revealed that CS research authorship is highly interconnected rather than a fragmented field and that while the community structure is dense, distinct semantic boundaries exist. The temporal analysis confirmed a massive paradigm shift in 2012, where deep learning keywords (eg "neural networks," "deep") rapidly displaced classical methods/terms. Furthermore, anomaly detection highlighted that "prestigious" outliers are often defined by rigorous, theoretical terminology (eg "linear systems"). Collectively, these models demonstrate that while scientific trends shift rapidly, the underlying structure of academic collaboration remains robust and centralized.