

# Analyzing Research Trends in Computer Science Using the DBLP Dataset

Olutoba "Toba" Sanyaolu (2135924), Diamond Oladeji (2164567), Diego Coronado (2303693),

Data Science I Final Report

December 05, 2025

---

## Contents

1 · Introduction .....	1
2 · Dataset Description .....	1
3 · Task 1: Data Preprocessing and Feature Generation .....	2
3.1 · Data Cleaning .....	2
3.2 · Stratified Sampling .....	3
3.3 · Text Normalization .....	3
3.4 · Black-Box Feature Generation .....	3
3.5 · Interpretable Feature Engineering .....	4
4 · Task 2: Topic Clustering .....	5
4.1 · Overview .....	5
4.2 · Determining an Appropriate Number of Topics .....	5
4.3 · Global Topic Interpretation ( $k = 19$ ) .....	6
4.4 · Refining the Large "Blob" Cluster .....	6
4.5 · Final Topic Visualization After Blob Refinement .....	7
5 · Task 3: Temporal Analysis .....	8
6 · Task 4: Predictive Modeling (Classification) .....	8
7 · Task 5: Network Analysis .....	8

---

## 1 · Introduction

Scientific publishing continues to expand rapidly, making it essential to develop scalable methods for understanding research dynamics. In this project, we analyze a large corpus of academic papers through preprocessing, text mining, clustering, temporal trend analysis, predictive modeling, and network analysis. Text features are extracted using TF-IDF and dimensionality reduction, enabling us to group papers into coherent topic clusters. Temporal analysis then reveals how these clusters evolve over time, highlighting both emerging areas such as machine learning and declining paradigms. Predictive models including Logistic Regression, Random Forest, and LightGBM are trained to classify papers by cluster, with performance evaluated using accuracy and F1 metrics.

In addition, we construct citation networks to measure influence through PageRank and centrality, identifying key papers and communities. Together, these methods provide both technical rigor and actionable insights into how ideas spread, venues specialize, and fields evolve. This report serves as a methodological guide and analytical narrative, bridging machine learning, network science, and interpretive clarity.

## 2 · Dataset Description

The dataset for this project comes from the DBLP Computer Science Bibliography, a large collection of metadata on computer science publications. We worked with JSON files (`dblp-ref-0.json` through `dblp-ref-3.json`) containing records with titles, abstracts, authors, publication years, venues, references, and citation counts. After loading the files into a single pandas DataFrame, we cleaned the data by dropping entries missing critical fields, filling missing values, and restricting the year range to 1950–2017.

For prototyping, we sampled subsets of 50,000–100,000 papers, while the full pipeline scales to millions of records. This dataset is valuable because it combines textual content for clustering and trend

analysis with citation networks for influence and community detection, providing a rich foundation for exploring how research topics evolve and spread over time.

### 3 · Task 1: Data Preprocessing and Feature Generation

#### 3.1 · Data Cleaning

The four DBLP JSON files were merged into a single dataset before any sampling or feature extraction. Papers missing a title or publication year were removed at the start, since both fields are required for text analysis and for understanding changes in research output over time. In later tasks (Task 2: Topic Clustering) we also found distinct clusters of German papers, so we added a filter containing common german words to remove these since they would provide no value in classification modeling.

##### 3.1.1 · Venue Filtering and Missing Venue Removal

Because the DBLP dataset included thousand of venues, many of which appear only a handful of times, we first calculated venue frequencies and kept only venues with at least fifty papers. During this step, we discovered that more than 506,699 papers had an empty venue value (""). To correct this, we removed all papers with missing venue metadata before sampling. After this fix, the most frequent venues in the cleaned dataset were recognizable and legitimate publication outlets confirming that the venue field was now reliable.

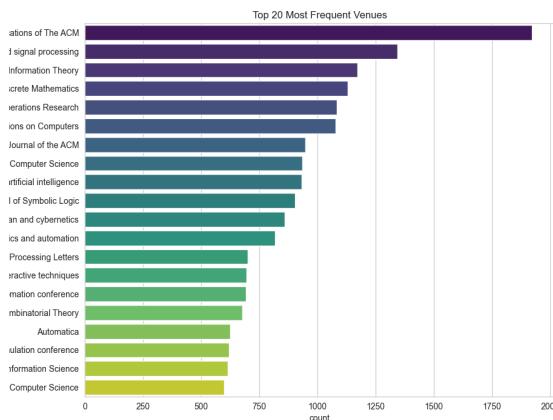


Figure 1: Venue Frequency Plot

##### 3.1.2 · Removing a Citation Artifact

When we later plotted the citation distribution using a log-scaled histogram, we observed a large spike at exactly 50 citations, which did not align with the expected heavy tailed distribution that is observed in bibliometric datasets. Because the spike was evidence of a possible data artifact, we removed all papers with `n_citation = 50` before sampling. After this correction, the updated histogram displayed a smooth decline in the upper tail.

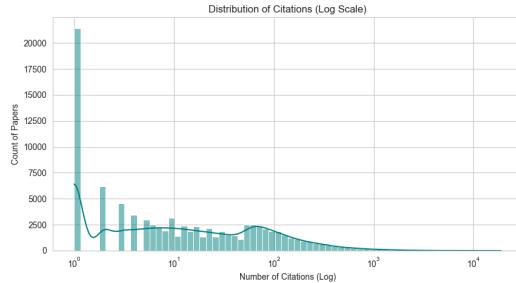


Figure 2: Citation Distribution

The distribution now matches what we expect from large bibliometric datasets.

### 3.2 · Stratified Sampling

After cleaning the venues and removing the citation artifact, we drew a stratified sample from the dataset by selecting up to 2,000 papers per publication year. We sampled per year to prevent the sample from being dominated by modern papers and in order to preserve long-term temporal structure. Sampling was intentionally done after cleaning the dataset, in order to ensure that the sample did not include missing venues or distorted citation values.

### 3.3 · Text Normalization

After sampling, text fields were normalized: missing abstracts were replaced with empty strings, titles and abstracts were converted to lowercase for consistent tokenisation, and a combined text field was created ( $\text{text} = \text{title} + " " + \text{abstract}$ ). This combined field served as the input for TF-IDF.

### 3.4 · Black-Box Feature Generation

#### 3.4.1 · TF-IDF Representation

Once the combined text field (title + abstract) was prepared, we transformed each paper into a numerical representation using TF-IDF. To keep representation focused and manageable, we limited the vocabulary to the 5,000 most informative terms and removed standard English stop words. This allowed the model to capture meaningful short phrases (e.g. “neural network”, “data mining”) rather than only being able to capture isolated keywords.

This produced a 5,000 dimensional matrix, where each paper is represented by a term of weights. Although TF-IDF preserves important distinctions between documents, the feature space is large and not ideal for clustering or visualization, which motivated the next step.

#### 3.4.2 · Choosing the PCA Dimensionality

In order to choose an appropriate number of components, we evaluated cluster quality across a range of PCA sizes. For each value in  $\{50, 75, 100, 125, \dots, 300\}$ , we reduced the TF-IDF matrix to  $k$  components and computed the silhouette score after running K-means. The silhouette score consistently decreased as dimensionality increased, and the highest score was obtained at 50 components. Although the silhouette scores and cumulative explained variance values are low this is expected for text datasets since TF-IDF variance is extremely spread out across thousands of components and text datasets also do not form sharply separated clusters.

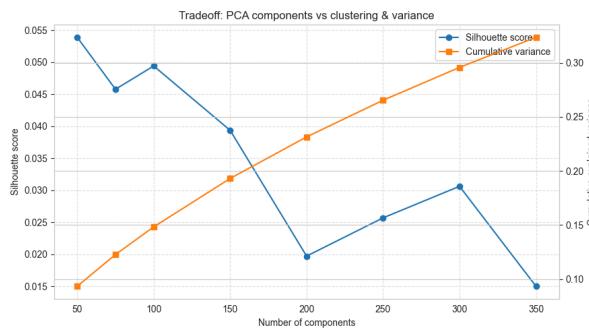


Figure 3: PCA n\_dimension tradeoff

### 3.4.3 · Venue and Author Embeddings

The PCA features were also aggregated to represent entities larger than individual papers.

For venues we took all papers that appeared in the same venue and averaged their 100-dimensional PCA vectors. This gives each venue a single vector that captures the thematic profile of that venue. Since empty venues were removed earlier every venue embedding corresponds to a real publication source.

For authors, we expanded the dataset so that each author-paper pair became an individual row. We then averaged the PCA vectors for all papers written by the same author. This produces one vector per author that summarizes the kind of topics they commonly work on.

These aggregated embeddings allow us to compare authors, venues, and individual papers using a consistent feature representation, which is useful for clustering, similarity analysis, and studying broader patterns in the DBLP dataset.

## 3.5 · Interpretable Feature Engineering

We also generated several interpretable metadata features:

- Number of authors: This measures collaboration size. When plotted across years, it shows a clear upward trend, increasing from roughly one author per paper in the early decades to about 3.8 authors by 2018.- title length in characters
- Title length: Recorded as the number of characters in the title. This offers a simple structural indicator of how concise or descriptive titles are.
- Number of references: The count of cited papers included in each record. This is useful for examining how referencing depth changes over time or across venues.
- Citation velocity: Defined as citations divided by the paper's age ( $2018 - \text{year} + 1$ ). This normalizes citation counts so older papers do not automatically appear more influential.

### 3.5.1 · Metadata Correlation Check

A correlation heatmap confirmed that these engineered features are not redundant as shown below.

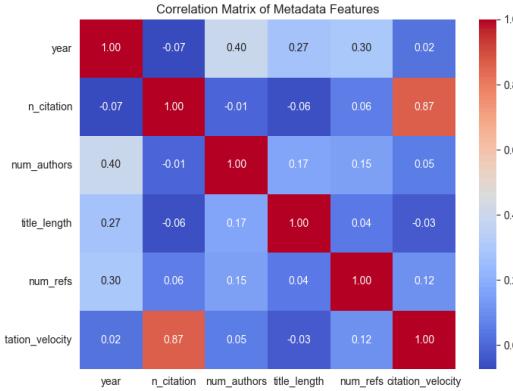


Figure 4: Metadata Correlation Plot

- number of authors increases with year ( $r \approx 0.48$ )
- number of references also increases with year ( $r \approx 0.42$ )
- citation velocity is almost perfectly correlated with citation ( $r \approx 0.90$ )

Some relationships were weak, and indicates that the metadata features capture different aspects of each paper.

## 4 · Task 2: Topic Clustering

### 4.1 · Overview

This task focuses on identifying major research themes within the DBLP dataset using unsupervised learning techniques. Because the dataset contains hundreds of thousands of computer science publications, we applied textual feature extraction, dimensionality reduction, clustering, and visualization methods to uncover coherent topical structures. During this process, we also encountered a large, heterogeneous cluster that required hierarchical re-clustering to fully resolve its internal thematic organization.

### 4.2 · Determining an Appropriate Number of Topics

#### Silhouette & Inertia Optimization ( $k = 6$ to 31)

The first step in topic discovery was selecting an appropriate value of  $k$  for K-Means. To do this, we ran K-Means for each  $k$  from 6 to 31 and computed two metrics.

1. Inertia - measures compactness of clusters (Elbow Method)
2. Silhouette Score - measures separation between clusters

Because computing the Silhouette score over the full dataset ( $\approx 90,000$ ) is extremely slow, we evaluated it on a random sample of 15,000 papers.

The results are shown below:

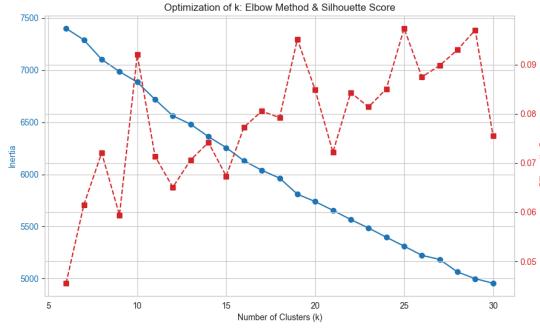


Figure 5: *K-Means Optimization for  $k = 6-31$*

local Silhouette peaks occurred around  $k \approx 10, 19, 25, 29$ . While  $k = 29$  achieved the highest score, the resulting clusters were highly fragmented and often mixed unrelated research areas. After computing cluster keywords for the  $k$ -values with local peaks and inspected topic coherence manually we found that  $k = 19$  produced the cleanest, most interpretable, and semantically coherent topics with stable boundaries.

#### 4.3 · Global Topic Interpretation ( $k = 19$ )

After choosing  $k = 19$ , the final K-Means model was run on the full PCA-reduced dataset (50 components). For each cluster, we extracted the top TF-IDF keywords, which revealed research areas including:

- Algorithms & Optimization
- Graph Theory
- Parallel Computing
- General CS & Application (the blob)
- ...

However, unlike the other clusters. Cluster 12 contained 34,806 papers far exceeding all others in size. Its keywords (e.g. *performance, process, used, applications*) indicating that it was not a coherent research field but rather acted as a catch all general CS region overlapping multiple ares. This required refinement.

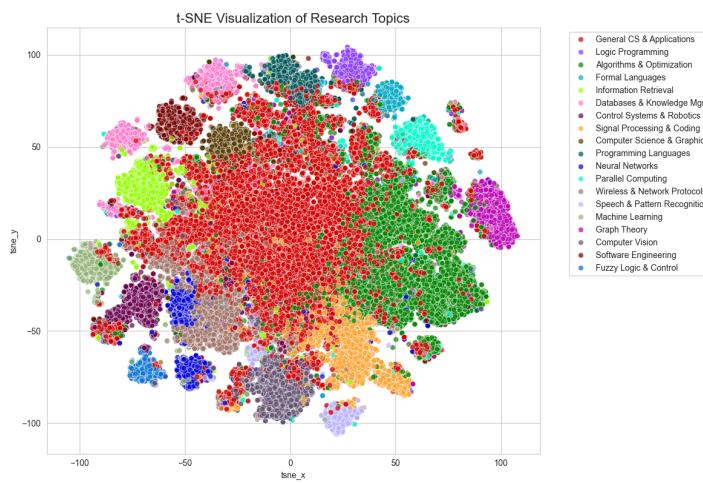


Figure 6: *t-SNE confirmed this by identifying a single dominant red cluster corresponding to Cluster 12.*

#### 4.4 · Refining the Large “Blob” Cluster

We isolated Cluster 12 and applied K-means again using the same 50-dimensional PCA embedding. We tested  $k = 5 - 15$  and computed silhouette and inertia specifically for this subset.

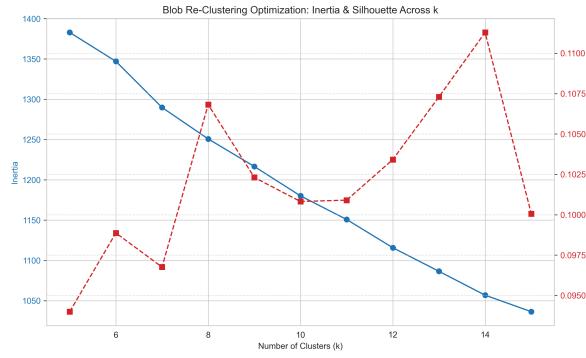


Figure 7: *Silhouette and inertia curves for re-clustering the 34,806 general purpose cluster.  $k = 14$  provides the optimal subtopic separation*

This resulted into 14 distinct subtopics.

### Subtopics Identified Within the Blob

Re-clustering revealed clear, interpretable subfields, including:

- Logic/Proof & Semantics
- Type & Recursion Theory
- Mobile Robotics & Navigation
- ...

These 14 subtopics were assigned new labels and merged back into the global dataset, replacing the monolithic “General CS & Applications” cluster. This expanded the topic taxonomy from 19 broad clusters to 32 refined research topics.

### 4.5 · Final Topic Visualization After Blob Refinement

After applying the refined labels, we regenerated PCA and t-SNE visualizations to show the full, updated topic landscape.

#### PCA Visualization (Refined Topics)

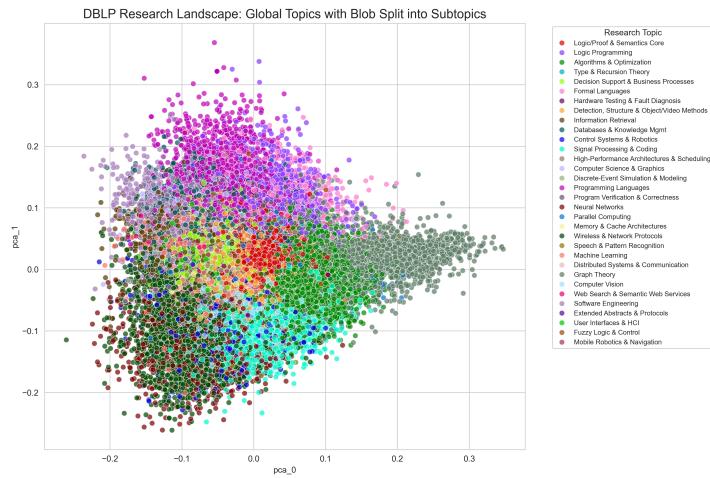


Figure 8: *PCA projection of all papers after refinements. Cluster 12 is now replaced by 14 distinct subtopics, improving interpretability despite PCA's inherent overlap*

#### t-SNE Visualization (Refined Topics)

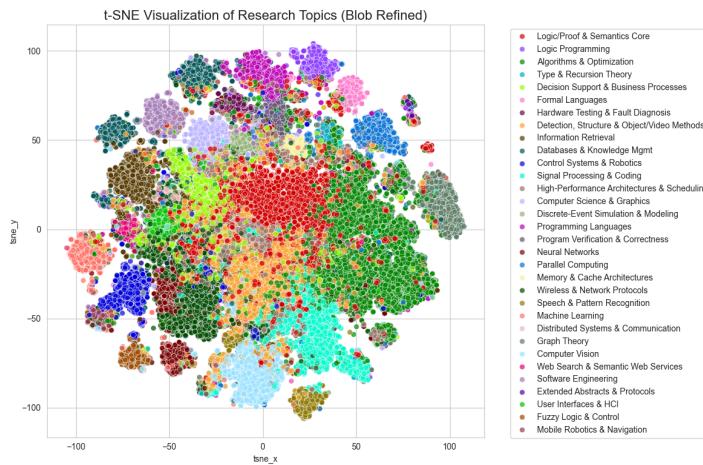


Figure 9: *t-SNE projection showing strong spatial separation across 32 final research topics, confirming improved thematic structure after blob subdivision*

## 5 · Task 3: Temporal Analysis

## 6 · Task 4: Predictive Modeling (Classification)

## 7 · Task 5: Network Analysis