Building a Domain-Specific Database for MLSys Papers

Aymaan Shaikh & Takuto Ban

Advisors: Prof. Hui Guan & Lijun Zhang

Our Team

Aymaan Shaikh

B.S. in Computer Science

Role: Student

Relevant Experience

- Incoming Data Science Intern @ Microsoft
- 2x Ex-AI/ML Engineering Intern @ Verizon
- 2x Ex-Full Stack Software Engineering Intern @ Fidelity Investments

Takuto Ban

B.S. in Computer Science and Mathematics

Role: Student

Relevant Experience

- Incoming Software Engineering Intern @ Capital One
- Ex-Software Engineering Intern @ GoDaddy

Table of contents

01

Research Objectives

02

Preliminary Work

03

Process

04

Next Steps

01

Research Objectives

Research Objectives

- **Goal 1: Build a curated database for MLSys papers.**
- **Goal 2: Analyze the database to extract insights about the field.**

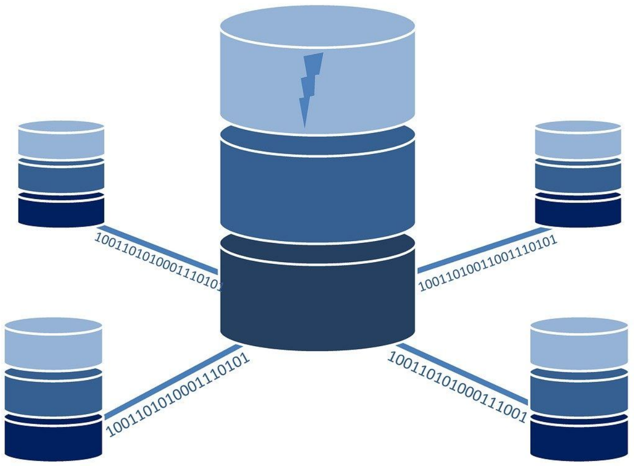
Specific Questions:

- What topics have been studied in MLSys?
- How have these topics evolved over time?
- Who are the key contributors, and what are the landmark papers?
- Can we predict future trends in the field?

Building a MLSys paper database

Instead of manually curating the database, we want to utilize Machine Learning to automatically explore papers and classify them as MLSys or not.

Our research focuses on creating a pipeline for effectively classifying domain-specific academic papers constructing a database automatically



Building the database

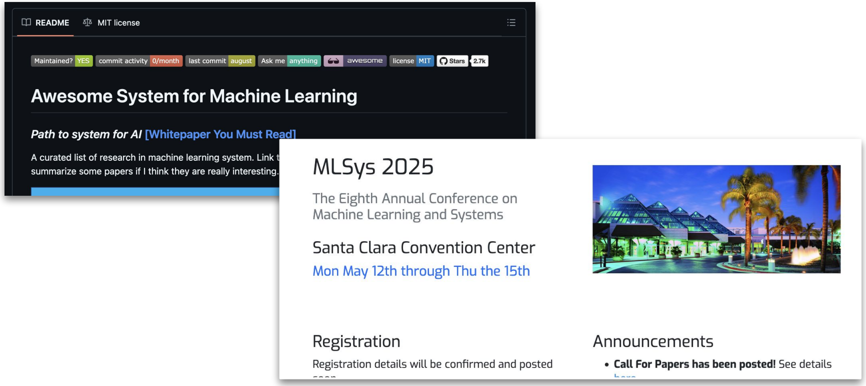
Step 1: Building a “seed” paper database

Step 2: Expand paper database through labeling & heuristics

Step 3: Experimenting with Machine Learning Classifications

Step 1: Building a "seed" database

- Scraped "seed" papers: papers that serve as the baseline for being MLSys
    - MLSys Conference (2018-2024) Accepted Papers: ~450 papers
    - Awesome System for Machine Learning GitHub Repo: ~150 papers
- Papers are stored along with Semantic Scholar Paper ID for access to metadata (e.g. citations, embeddings)



README MIT license

Maintained? YES | commit activity 0/month | last commit august | Ask me anything | awesome | license MIT | Stars 2.7k

Awesome System for Machine Learning

Path to system for AI [Whitepaper You Must Read]

A curated list of research in machine learning system. Link t
summarize some papers if I think they are really interesting.

MLSys 2025

The Eighth Annual Conference on Machine Learning and Systems

Santa Clara Convention Center

Mon May 12th through Thu the 15th

Registration

Registration details will be confirmed and posted

Announcements

- **Call For Papers has been posted!** See details

Semantic Scholar

Step 2: Expand paper database through labeling & heuristics

- Labeling
  - Collected 200 papers that cite one of our “seed” papers and manually labeled them
  - Results: 75 negative papers & 125 positive papers
- Heuristics (negative papers)
  - ML but not Sys: NeurIPS 2023 papers that does not include keywords (“systems”, “hardware”, “GPU”) in title/abstract; 250 papers
  - Sys but not ML: ASPLOS papers that does not include keywords (“Neural Network”, “Machine Learning”, “Deep Learning”, “CNN”, “Language Model”); 250 papers
  - Other Non-CS Disciplines: Biology & Chemistry; 100 papers
- Total: ~650 positive papers, ~650 negative papers

Step 2.5: Literature Review

Domain-specific long text classification from sparse relevant information: D’Cruz et al.

- Compile a list of target terms for the domain, then filter sentences containing such terms
- Use transformer encoder for each sentence to generate an embedding, then use linear layer for classification
- Experimented with medical records, achieved 60-80% accuracy

Knowledge-based Document Classification with Shannon Entropy: Chan

- Compile a list of target terms for the domain, then for a document, compute its concept abundance (TFIDF) and concept diversity (Shannon Entropy)
- Score is the product of TFIDF and Shannon Entropy, use classifical machine learning classification algorithm
- Experimented with documents of various genres, achieved better performance than Logistic Regression

Attention is Not Always What You Need: Towards Efficient Classification of Domain-Specific Text: Wahba et al.

- Instead of computationally expensive transformer models, use linear SVM model with TFIDF vectorization for more efficient, explainable classification
- Experimented with IT tickets, achieved similar accuracy compared to transformer-based methods

A transfer learning approach to interdisciplinary document classification with keyword-based explanation: Huang et al.

- First, train single-discipline classification models using single-discipline labeled data
- Transfer knowledge from these models to initialize a multi-disciplinary model, then train it with multi-disciplinary data
- Addresses the scarcity of multi-disciplinary by combining knowledge from individual disciplines

Step 3: Experimenting with Machine Learning Classifications

- Convert papers into numerical representation through an embedding model
- SPECTER (v2)
  - transformer-based embedding model adapted from SciBERT for document-level relatedness of *academic papers*; available through Semantic Scholar API
  - Pros: embedding model made specifically for academic papers ⇒ more accurate representation
  - Cons: model training based on citation/reference relationships, which our method is also based on, potentially creating bias
- Sent2Vec
  - an NLP model that generates vector representations for individual sentences, which can be aggregated to represent entire documents. Leverages pre-trained word embeddings (e.g., Word2Vec) and enhances them with contextual features like bigrams, capturing nuanced relationships between words within a sentence.
  - Pros: can be run locally with decent efficiency; no need to rely on external API
  - Cons: too basic of an embedding model, might have less rich representation

ML Techniques

- Unsupervised (K-Means):
    - Determine cluster thresholds for relevance classification.
- Supervised (KNN):
    - Train on balanced positive and negative datasets for relevance classification.

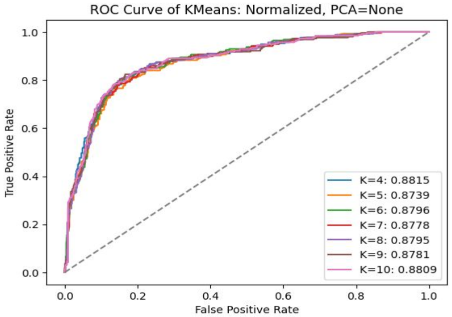
- Embedding Performance
    - Problem:
        - Uncertainty in embedding effectiveness.
    - Solution:
        - Conduct ablation studies to compare:
            - Sent2Vec vs. Specter2
            - Cosine similarity vs. Euclidean distance

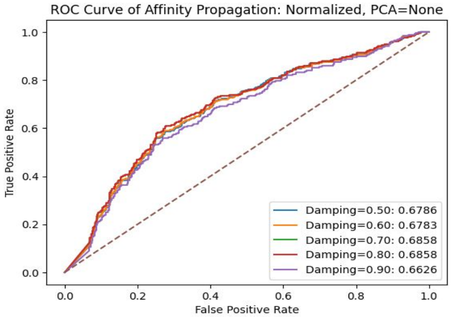
ML – Unsupervised Learning

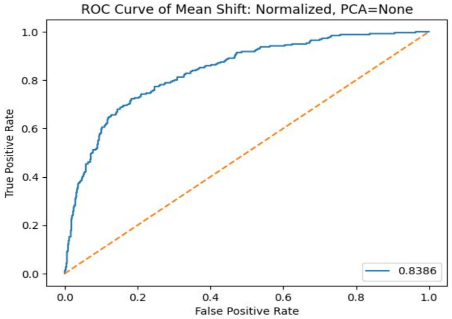
- Unsupervised Learning: under assumption that only positive data points (only MLSys papers) are available
    - Thus, no need to collect negative data points for training
    - Although negative data points were used during hyperparameter tuning, requires much less data samples
- Approach: Clustering Based Scoring Method
    - Group positive paper data points into clusters through various algorithms
    - For given data point, find closest cluster centroid
    - Score is the negative distance (Euclidean or cosine) between the data point and the closest centroid
- Algorithms Used
    - K Means
    - Affinity Propagation
    - Mean Shift
    - Gaussian Mixture Model

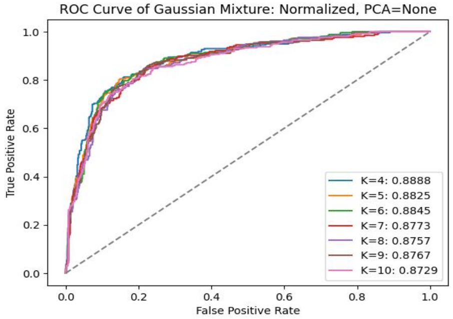
ML – Unsupervised Learning

- Results: K Means and Gaussian Mixture Model performed the best; K Means was significantly faster computationally than GMM
- Experimented with applying Principal Component Analysis to a lower dimension and Normalization; Normalization without PCA performed the best
- Experimented with different distance metrics; Euclidean distance performed better than cosine similarity
- Best ROC-AUC Score: ~0.88



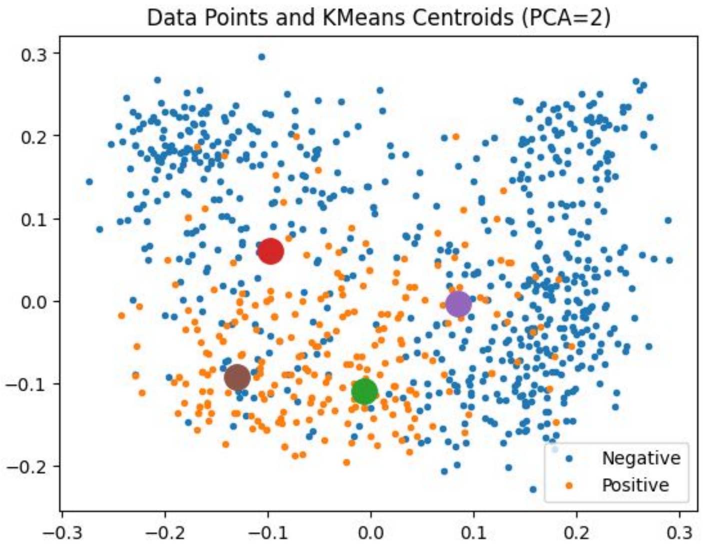


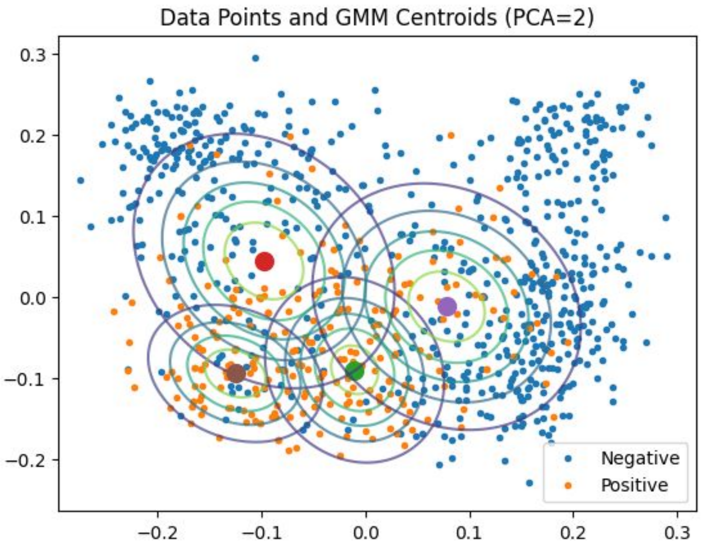




ML – Unsupervised Learning

- Comparing the cluster centroids between K Means and GMM (after applying PCA to 2 dimensions); both algorithm resulted in similar centroids



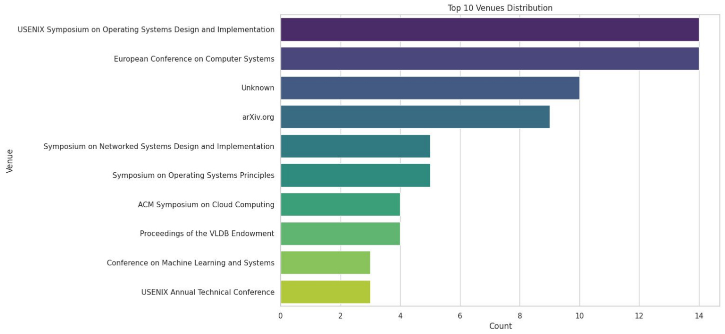


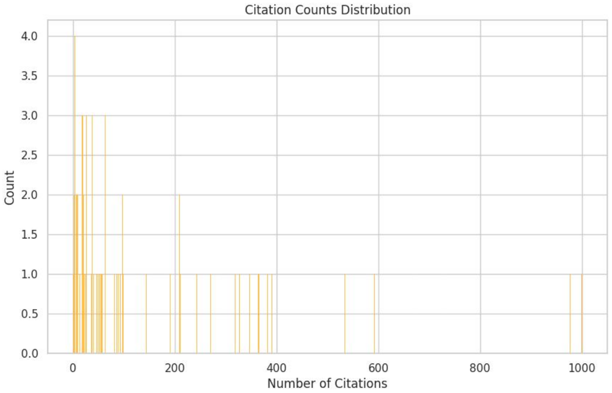
Preliminary Results

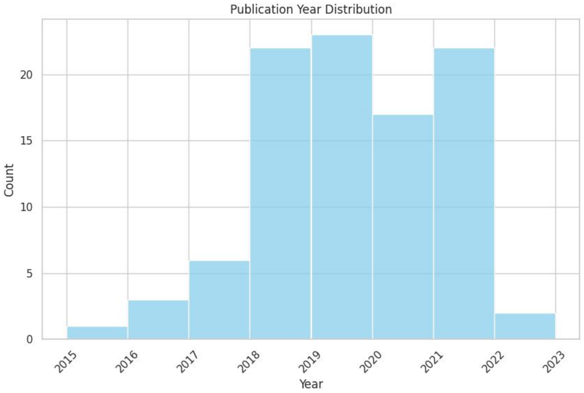
- K-Means Analysis:
    - Visualized clustering effectiveness using PCA/Histogram analysis.
    - Threshold fine-tuning improved precision.
- Supervised Approach:
    - KNN showed promising results with well-curated samples.
    - Evaluated similarity metrics: Cosine similarity performed better.

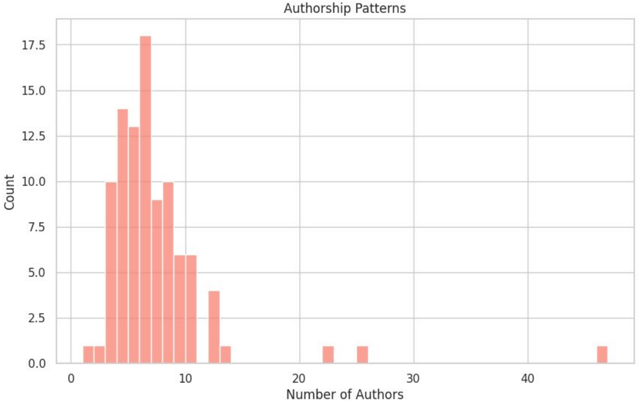
Example of Exploratory Data Analysis (EDA)

- MLPapersFinal.csv
- Example of how we are getting our data and the types of info we are looking for









Future Work

Expand and refine the classifier

- Experiment with embeddings like Specter2 vs. Sent2Vec.
- Evaluate multi-class vs. binary classification.

Enhance database accuracy

- Use clustering-based filters for negative samples.
- Improve unsupervised and supervised pipelines.

Data insights

- Identify trends in MLSys topics and predict emerging research areas.
- Analyze co-authorship and publication venues for influential contributions.

Pipeline Automation

- Automate data ingestion, processing, and classification.

Insight Extraction

- Graph-based analysis:
- Visualize paper connections (nodes: papers, edges: citations).
- Identify influential authors, venues, and emerging topics.
- Trend analysis:
- Time-series visualization of topic evolution.

Improving Classifiers

- Supervised Approach
- Unsupervised Approach

Broader Impact

Facilitate MLSys research by providing a robust, structured database.

Offer methodologies applicable to other interdisciplinary domains.

Lay groundwork for predictive analysis of future research trends.

Thank You!