Information Retrieval from Research Papers using Text Mining

Group 11
Gagandeep Mangat - 20111019 (gagandeep20@iitk.ac.in)
Rohit Katariya - 20111050 (rohitk20@iitk.ac.in)
Mallampet Adhi Lakshmi - 20111003 (lakshmi20@iitk.ac.in)
Naga Durga Krishna Mohan Eaty - 20111037 (kmohan20@iitk.ac.in)

December 6, 2020

CS685 Data Mining
Department of Computer Science
Indian Institute of Technology Kanpur

Abstract

Research papers can be categorized based on their main topic and different journals can be analyzed for the topics of interest of the papers published in them. The collection of research papers available on the IITK Thesis webpage provides a good resource for researchers. With this volume comes a need for organization, a task which we have performed in this project.The synergy between them helps to discover different interesting text patterns in the retrieved articles. In this project three main goals were completed. First, we developed a sensible clustering of the articles based on textual similarity. Second, Analytics of the complete repository is done and wordclouds for each department is built, from which we can understand what are the most frequently used words, which from we can get a understanding of the major research work going in that department or under a particular supervisor. And lastly, we built a web interface for a content-based recommendation system, which recommends top five relevant articles based on content of some particular user-chosen article and five other recommendations based on supervisor of user-chosen article. The work done in this project is limited to only IITK thesis repository, the idea from this can be used to extend to many other bigger repositories.

Problem Statement

The IITK Thesis repository enables the user to only search for research articles based on certain keywords such as article name, supervisor name, department, etc. It would be beneficial if statistical analysis was made available to the user for leveraging the available knowledge in a better way. Also, because of the large number of research papers available, a user might not find an article that is most relevant to his/her purpose. For this, we have tried to apply some text mining techniques to analyze the various parameter of the dataset consisting of all the research papers. Also, to tackle the second problem, we will develop a recommendation system.

Introductiom and Motivation

Nowadays, almost all of the existing information in different institutions is preserved in electronic documents in which it contains semi-structured data. In these documents, the “abstract” is an example of unstructured text component.A study has stated that text mining has become one of the trendy fields that has been incorporated in several research fields such as computational linguistics, Information Retrieval and data mining. Information recovery methodologies like text indexing techniques have been developed for handling unstructured documents. In conventional researches, it is assumed that a user mostly searches for known terms, which have been previously used or written by someone else. The main problem is that the search results are not relevant to the user’s requirements. One solution is to use text mining in order to find out relevant information, which is not indicated explicitly nor written down so far.

Structured data can be handled through data mining tools while unstructured or semi-structured datasets like full-text documents, emails, etc can be handled through text mining. The major objective of text mining is to find out the unknown information, something that is not recognized by anyone. Data is the basic kind of information, which is required to be organized and mined for the knowledge generation. Discovering patterns and trends from huge data is a significant challenge. The common structure of text mining involves two consecutive stages: text refining and knowledge distillation. In text refining, free-form text documents are converted into an intermediate form, whereas in knowledge distillation, patterns or knowledge are derived from intermediate form. Various research areas, techniques, and models are involved in different research domains. The hottest topics of the research domains are the primary focus of many research papers. The research results of a particular domain may influence other research domains since some research domains may have similar topics.

The primary goals completed in this project are (1) Using text mining techniques for identifying the abstracts of a all research papers on the IITK thesis webpage and developing a hierarchical connection among them. (2) Using visualization tools for presenting both the topics and the association among them as a convenient way to help users to determine relevant topics.(3) Developing a recommendation system to help the user get access to relevant research papers.

Datasets Used

Source of Dataset : The research papers were be obtained from the IITK Thesis repository from the link **http://172.28.64.70:8080/jspui/**(accessible only on IITK network). The papers are organized according to various different attributes which include title, author(s), department, supervisor(s), subject(s), and abstract.

It will consist of abstracts of about 18036 research papers published by the people of IIT Kanpur since 1963.

We have scraped the papers off the website using some web scraping techniques in python. Some of the libraries that will be used for scraping are mentioned below.

- Requests - The requests module allows you to send HTTP requests using Python.
- Beautifulsoup - Python library for pulling data out of HTML files.
- html5lib - html5lib is a pure-python library for parsing HTML.
- pandas - pandas is a library used for data analysis and manipulation. It will be used for organizing and maintaining the scraped data.

Methodologies

Web-Scraping

We have scraped the papers off the website using some web scraping techniques in python. Some of the libraries that will be used for scraping are mentioned below.

- Requests - The requests module allows you to send HTTP requests using Python.
- Beautifulsoup - Python library for pulling data out of HTML files.
- html5lib - html5lib is a pure-python library for parsing HTML.
- pandas - pandas is a library used for data analysis and manipulation. It will be used for organizing and maintaining the scraped data.

After scraping the data, the data is stored in csv file and it is used in further stages of cleaning and analysis etc.,

Text Mining

Major steps in text mining include Pre-processing, Feature Extraction and Clustering. Lets discuss these in detail.

Pre-Processing : The text data scraped from the web requires pre-processing for text analysis. As the raw text contains punctuation and stopwords, also these are not much of relevance so deleting these would not be any significant loss of useful information. This also reduces the text size as well as deletes unnecessary content from the original text. The techniques that we are going to implement are

- **Capitalization** - Convert all the words to lowercase. This makes the complete text uniform.
- **Stop Words** - Stop words are commonly used words in English (a, an, the, to etc.). These words do not signify any meaning in the text.
- **Tokenization** - The process of splitting paragraphs into sentences and sentences into words. These splits are called tokens.
- **Stemming/Lemmatization** - The process of conflating the variant forms of a word into a common representation called the stem. This can be seen as removing tenses from the text. For example, words like "presented, presentation, presenting" can be commonly represented as "represent".
- **Word Count/Density** - The dataset consists of 17289 entries, out of which the average word count for the abstracts is 332 words per abstract. Most of the words present in the abstract are not relevant to the context of the or are stop-words. We find these redundant words using the feature extraction process.

For Implementing the above techniques, various python libraries like nltk, scipy are used. After the cleaning process, the average word count for the abstracts comes down 189 from 332.

Feature Extraction The feature extraction techniques are used to find the similarities between the parts of texts. This is required because machine learning algorithms cannot directly work on raw text data. So we need feature extraction techniques to convert the raw text in natural language into the feature matrix.

The feature extraction process starts with counting the occurrences of words in the abstracts and creating a bag of words model. A vocabulary is created using the words present in the dataset. The BoW model then represents the feature vector for every entry using the word count of every word present in the vocabulary. The resulting sparse matrix is of the dimensions 17293x107255 i.e. 17293 articles with a vocabulary of 107255 words. Considering the huge vocabulary size, we only take into account a vocabulary of 10000 words based on the frequency of words present in the dataset. The top unigrams, bigrams, and trigrams based on the generated BoW model are shown in the plots below.

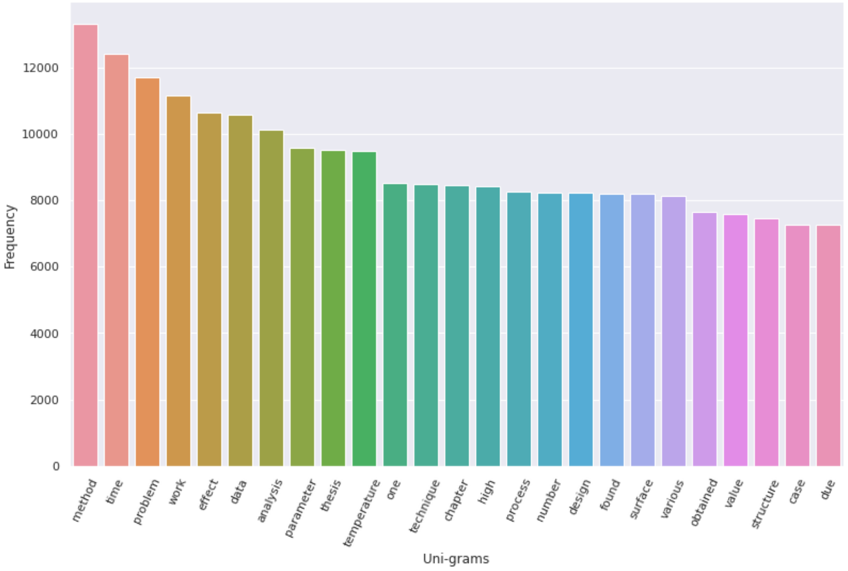


Figure 1: Unigram words of complete dataset

The BoW model only takes into account the frequency of words in the dataset which may not be the best measure to find the relative importance of words in the text. This is since a more common word irrelevant to the text may appear much more than a lesser common but important word to the context. To overcome this flaw, TF-IDF process is used which assigns a numbered importance to a word in the given text and database according to its term frequency (TF) in the given text and its inverse document frequency (IDF) across the entire dataset.

$$\text{TF} = \frac{\text{Frequency of term in document}}{\text{Total number of terms in the document}}$$

$$\text{IDF} = \frac{\log(\text{Total Documents})}{\text{No of documents with the term}}$$

Essentially, we create a new dataset where the abstract of every text is replaced by the top 50 most important words in that text according to the TF-IDF scores of the words. This dataset can further be used to cluster the documents and provide better results for the clusters as the texts can be related better in terms of only their most important keywords.

WordCloud - After the feature extraction process, we visualize the results based on the word clouds which can be generated based on the departments or supervisors. A word cloud is a visual representation of words wherein the size of the words denote their relative importance to the context with larger

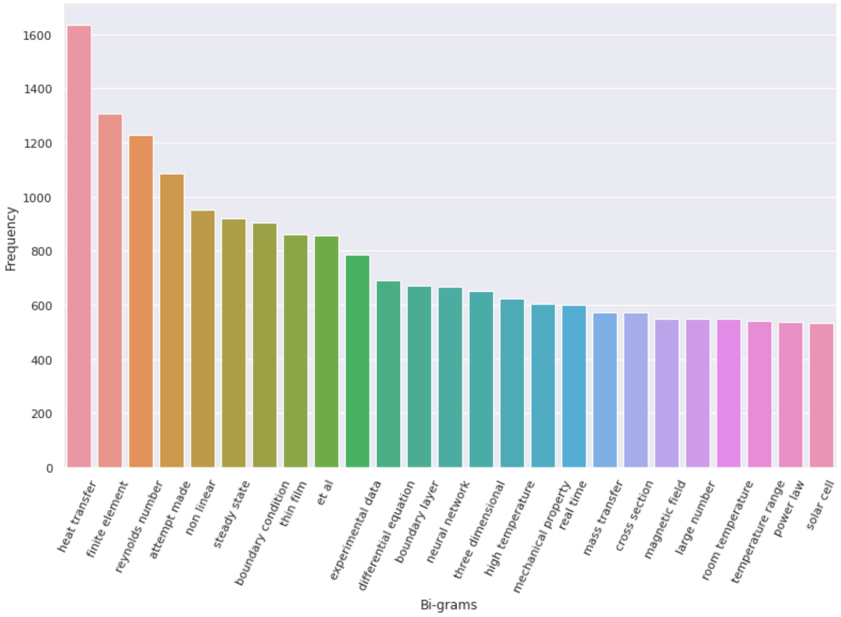


Figure 2: Bigram words of complete dataset

words signifying more value. The process used for generating word clouds is summarized below:

- Find all the texts relating to the department or the supervisor.
- Generate the vocabulary of words for those texts and apply TF-IDF to find the importance of keywords in the texts.
- Select the top 50 keywords from every text, and add them to the list of the keywords.
- Generate the word clouds based on the list of keywords and their TF-IDF scores.

The functions to generate these word clouds are present in the file 'feature-extraction-and-visualization.py', named word_cloud_dept() and word_cloud_sup() respectively.

K-Means Clustering - It is unsupervised learning, in our case clustering algorithm classifies a dataset into a K number of clusters. here K is unknown. We learn the best K using cross validation technique. Main intention is clusters will be defined by K centroids, where each centroid is a point that represents the center of a cluster. This algorithm works interactively, where initially each centroid is placed randomly in the vector space of the dataset and move themselves to the center of the points which are closer to them. In each new iteration

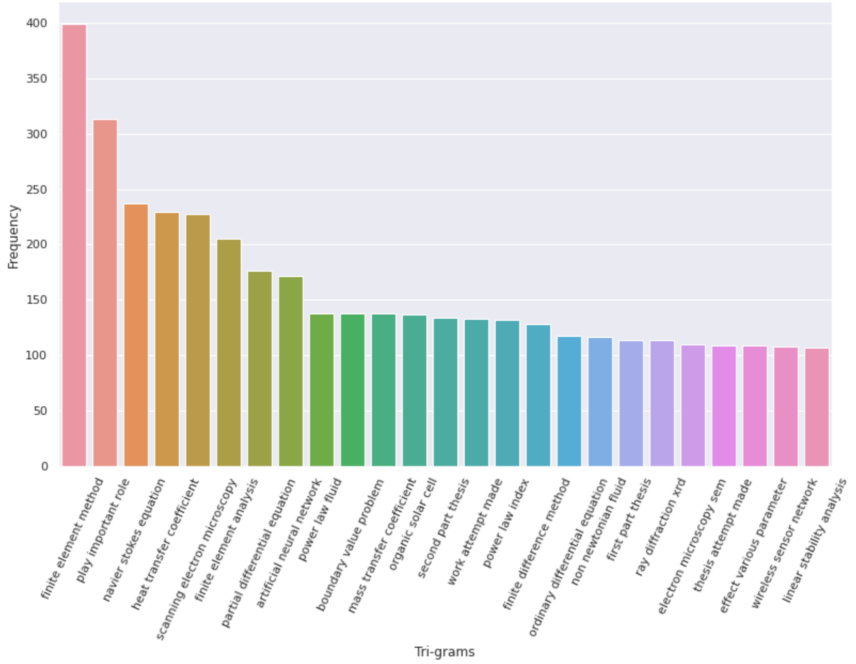


Figure 3: Trigram words of complete dataset

the distance between each centroid and the points are recalculated and the centroids move again to the center of the closest points. The algorithm is finished when the position or the groups don't change anymore or when the distance in which the centroids change doesn't surpass a pre-defined thresholds.

Algorithm classify the keywords into respective clusters. **Example** : All keywords related to machine learning field are grouped together.

From the previous step, Feature extraction, we get a feature matrix which will be passed as input to the clustering algorithm and the output from this will be cluster label for each data point and cluster centers, with these two things we can visualize it in 2-D or 3-D and draw some conclusions. The outputs can also be used for recommender system, for faster computation. For implementing this scikit-learn library is used.

For Recommendation system - In model-building stage, the system first find the similarity between all pairs of items, then it uses the most similar items to a user's selected items to generate a list of recommendations in recommendation stage. The similarity will be derived from the description of the item. Then each item will be represented by a TF-IDF vector. We have already explained TF-IDF earlier. After defining the TF-IDF value for the words, then we can make the keyword vectors for each abstract.

Other method like sigmoid kernel is also tried for implementation, but for this the runtime is huge which cannot be used on webpage.

Compare the Similarity of the item TF-IDF vector - To compute how similar the item vectors are, we can use methods such as Cosine Similarity, Euclidean Distance, etc. Then the recommender will give recommendation based on the most similar abstract. We tried to implemented both methods, found no significant difference in the recommended items.

WebApp Development

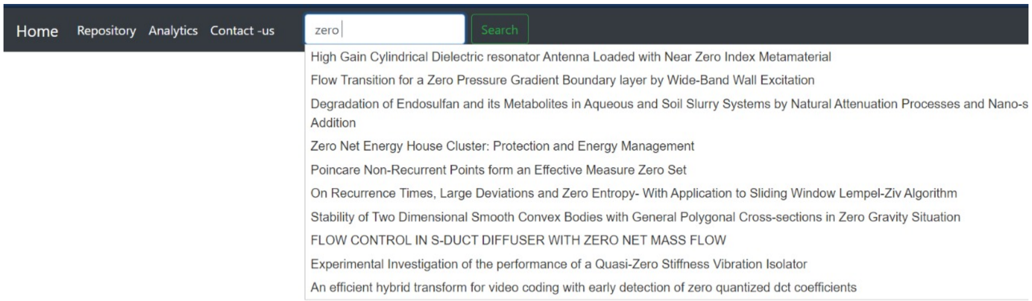
Major packages that are used for doing webapp development are

- **Frame work** :Django
- **Backend Database**: postgressql
- **Styles**: Bootstrap
- **Model deployment**:joblib
- **Auto complete feature**: jQuery
- **Dynamic charts**:charts.js

The webpage developed is very a basic webpage of our project. It contains a navigation bar which contains home, repository, Analytics tabs.

- **home page**: Search any article.Auto complete jQuery feature will help you out in filling up the article name.no need to give entire name of the article.you can select the name from the drop-down of listed articles. Then our model will recommend related articles based on cosine similarity and also retrieves articles under that supervisor.connect to VPN and click on the retrieved recommended articles link to read the full thesis.
- **Repository tab**: Filtered department-wise. Click on the respective department to see total thesis repository of that branch.we have not displayed entire article .connect to VPN to access the total paper.
- **Analytics page**: Contains statistical analysis of departments.

All this plots are dynamic and interactive. Below are some screenshot from the web application.



High Gain Cylindrical Dielectric resonator Antenna Loaded with Near Zero Index Metamaterial

Flow Transition for a Zero Pressure Gradient Boundary layer by Wide-Band Wall Excitation

Degradation of Endosulfan and its Metabolites in Aqueous and Soil Slurry Systems by Natural Attenuation Processes and Nano-s Addition

Zero Net Energy House Cluster: Protection and Energy Management

Poincare Non-Recurrent Points form an Effective Measure Zero Set

On Recurrence Times, Large Deviations and Zero Entropy- With Application to Sliding Window Lempel-Ziv Algorithm

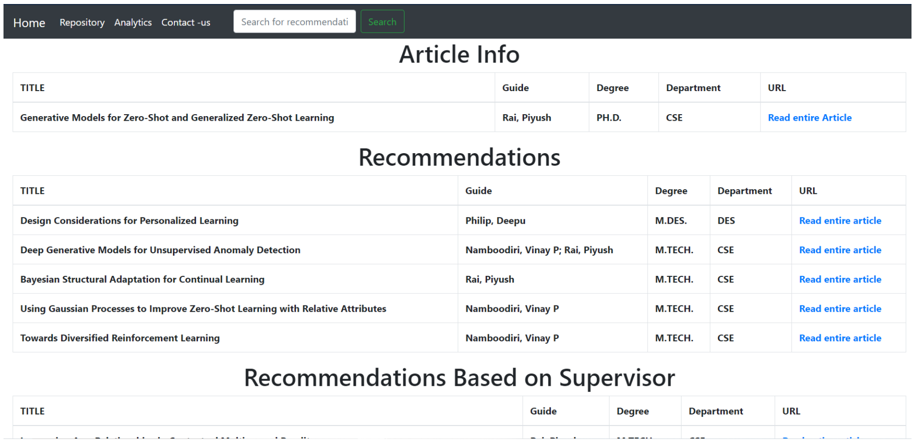
Stability of Two Dimensional Smooth Convex Bodies with General Polygonal Cross-sections in Zero Gravity Situation

FLOW CONTROL IN S-DUCT DIFFUSER WITH ZERO NET MASS FLOW

Experimental Investigation of the performance of a Quasi-Zero Stiffness Vibration Isolator

An efficient hybrid transform for video coding with early detection of zero quantized dct coefficients

Figure 4: Home Page, with auto complete dropdown for search bar



Article Info

TITLE	Guide	Degree	Department	URL
Generative Models for Zero-Shot and Generalized Zero-Shot Learning	Rai, Piyush	PH.D.	CSE	Read entire Article

Recommendations

TITLE	Guide	Degree	Department	URL
Design Considerations for Personalized Learning	Philip, Deepu	M.DES.	DES	Read entire article
Deep Generative Models for Unsupervised Anomaly Detection	Namboodiri, Vinay P; Rai, Piyush	M.TECH.	CSE	Read entire article
Bayesian Structural Adaptation for Continual Learning	Rai, Piyush	M.TECH.	CSE	Read entire article
Using Gaussian Processes to Improve Zero-Shot Learning with Relative Attributes	Namboodiri, Vinay P	M.TECH.	CSE	Read entire article
Towards Diversified Reinforcement Learning	Namboodiri, Vinay P	M.TECH.	CSE	Read entire article

Recommendations Based on Supervisor

TITLE	Guide	Degree	Department	URL

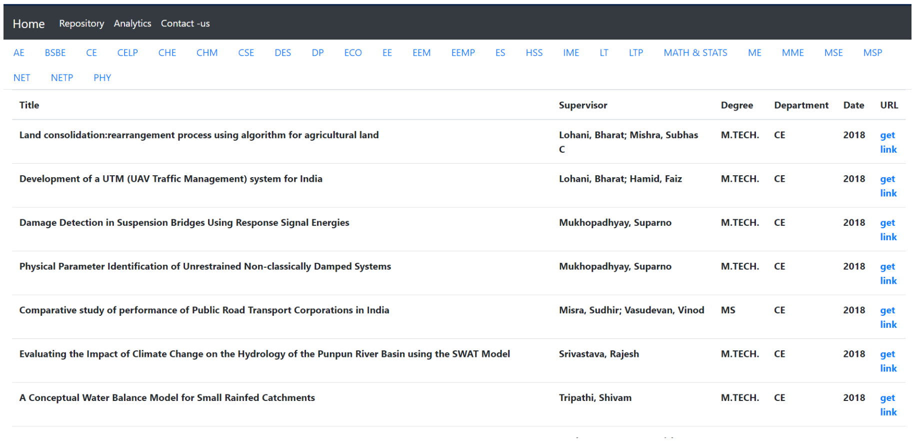
Figure 5: Recommendation results for a article searched

Results

WordCloud - The ‘CSE’ department word cloud is shown below. Other department word clouds can be found on the analytics page of the project web app.

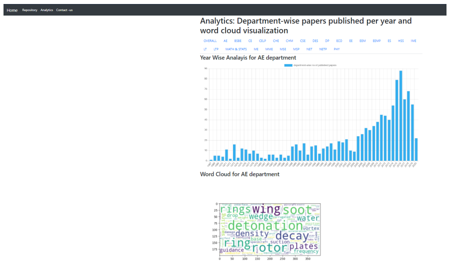
Word cloud for the supervisor ‘Rai, Piyush’ is also shown in the figure below.

Clustering - After feature extraction step, Clustering is implemented. Clustering is tried with different no.of clusters say 25,30,50 etc., and also with different number of features. The initial idea for number of clusters is taken from a reference, in which they were clustering some 20000 articles into 50 clusters. The observations that are drawn from clustering are, there is lot of difference in the clusters between clustering after dimensionality reduction and clustering with features extracted from feature extraction. The clustering with dimensionality



Title	Supervisor	Degree	Department	Date	URL
Land consolidation:rearrangement process using algorithm for agricultural land	Lohani, Bharat; Mishra, Subhas C	M.TECH.	CE	2018	get link
Development of a UTM (UAV Traffic Management) system for India	Lohani, Bharat; Hamid, Faiz	M.TECH.	CE	2018	get link
Damage Detection in Suspension Bridges Using Response Signal Energies	Mukhopadhyay, Suparno	M.TECH.	CE	2018	get link
Physical Parameter Identification of Unrestrained Non-classically Damped Systems	Mukhopadhyay, Suparno	M.TECH.	CE	2018	get link
Comparative study of performance of Public Road Transport Corporations in India	Misra, Sudhir; Vasudevan, Vinod	MS	CE	2018	get link
Evaluating the Impact of Climate Change on the Hydrology of the Punpun River Basin using the SWAT Model	Srivastava, Rajesh	M.TECH.	CE	2018	get link
A Conceptual Water Balance Model for Small Rainfed Catchments	Tripathi, Shivam	M.TECH.	CE	2018	get link

Figure 6: Department wise thesis repository with article link



Analytics: Department-wise papers published per year and word cloud visualization

Year Wise Analayis for AE department

Word Cloud for AE department

Figure 7: Department wise analytics and wordclouds

reduction is lot better. Also not all dimensionlaity reduction techniques were good. Only TSNE gave better reduced features, other techniques like TSVD and UMAP gave some outliers. So for clustering TSNE is used. Lets see some results with graphs.

As mentioned the clustering results, with total features was not good. So other techniques like TSVD and UMAP were also used and those did not give good results.

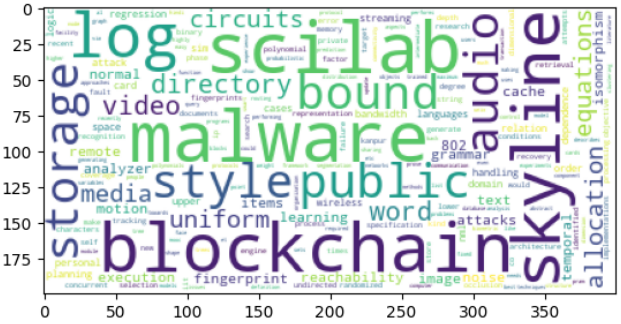


Figure 8: CSE department wordcloud

Statistical Analysis - Some of the results from statistical analysis are as follows

- **Total no of publications in IITK repository**:17311
- **Number of departments**:27

Refer plots from the website for detailed analysis of number publications department,year-wise. An example plot is attached as Figure 4.

Figure 9: Supervisor wordcloud

Discussion

In this project, there are many hyperparameters like number features to be extracted after feature extraction. Also hyperparameters like no of clusters in clustering. What dimensionality reduction technique should be used etc., For these things we had choosen the hyperparameters by randomly selecting a value and fine tuning it. There are no ground truth values for those and the results that we got are not the exactly correct.

Also for the recommendation, the recommendations that we got are also the ground truth recommendation. There can be much better results for this case and the technique that we used, the cosine similarity, may not be good for other applications. Also there were no real metrics for judging the recommendation. We judged the recommendation based on intuition.

Also the clusters, that we got doesnot have any label, so it is difficult to know what category can be inferred from each cluster. Again this should be done on intuition itself.

These are some areas where this project can be improved upon.

Future Direction

Currently in this project, data is only extracted from IITK repository. This can be extended to other repositories like Sodhganga, Springer etc.,. Also the current webpage is just in its raw stage, there is lot of room for improvement there. The recommendations and features extracted from text mining can also be refined.

Another thing is, currently the webpage is semi-dynamic, meaning it won't be taking data from the repositories regularly. So we need to make it dynamic so that when ever the repository gets a new article the database of the webapp should also get updated.

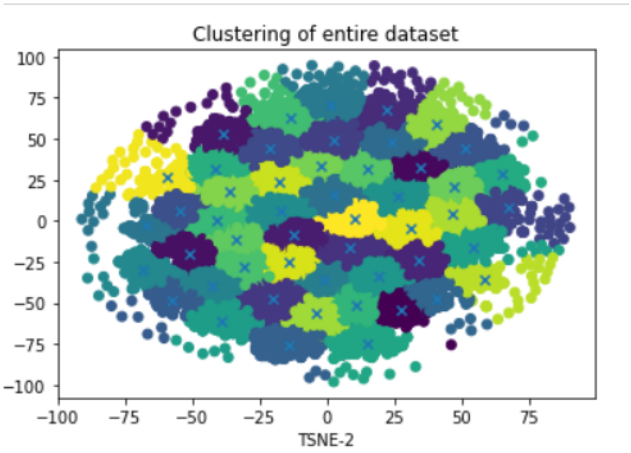


Figure 10: Cluster visualization of clustering performed on reduced feature space. TSNE is used for dimensionality reduction

Currently, we did not use any metrics for verifying the recommendation results. We just used intuition for verifying. There are many metrics to do this. Implementation of this will refine the recommendations.

Also, currently we are not using clustering results for recommendation system. This makes the runtime for the recommendation high. If we can use the clustering results and compute the similarity only for those data points that are present in that particular cluster then the runtime would reduce drastically.

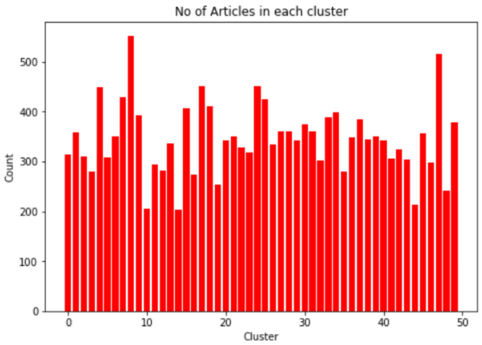


Figure 11: No of articles in each cluster for 50 clusters. Clustering is done on TSNE reduced feature space

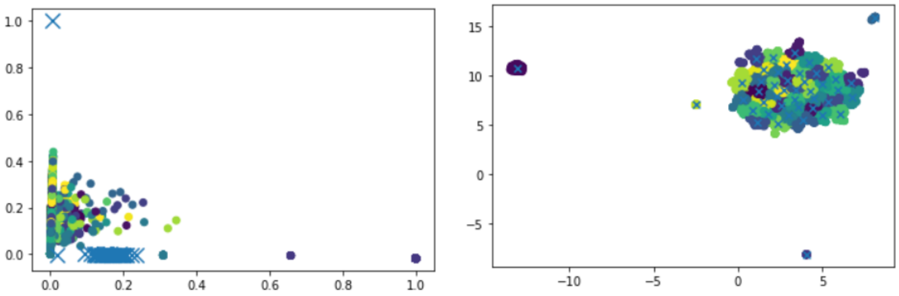


Figure 12: Left fig - Clustering using UMAP reduced features and Right fig - Clustering using TSVD reduced features