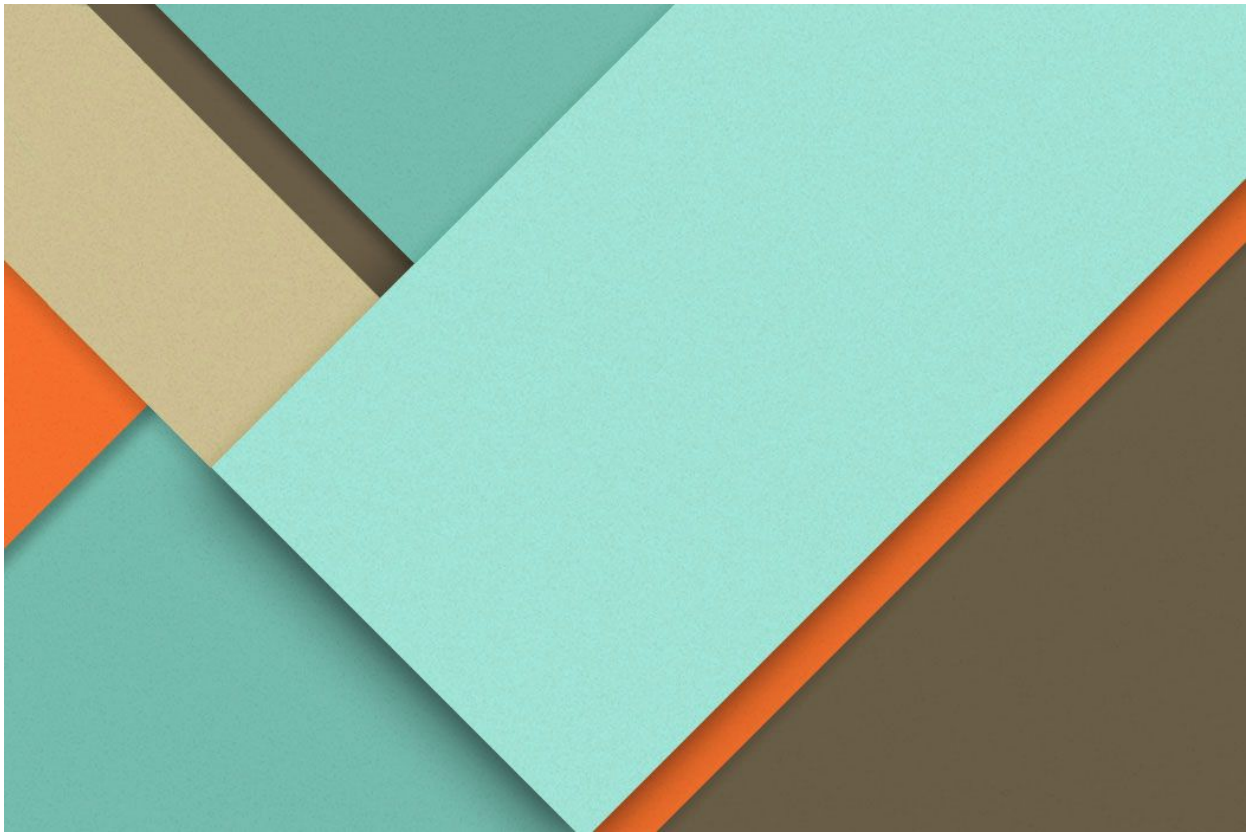


CSE564: Final Project Proposal

04.16.2019

Group members:
Ananya Palit (ID: 112077303), Souradeep Chakraborty (ID: 112281492)
Computer Science Department
Stony Brook University

Overview

The Digital Bibliography and Library Project (DBLP) is a well-known computer science bibliography website hosted at the University of Trier in Germany. It currently contains information of 3.66 million computer science publications along with the authors and the category such as journal, conference, book etc. Considering the huge amount of information in this database, it is difficult to browse through the textual data manually to find insights and correlations in it, particularly the time-varying ones. Burch et al. [1] presented certain interactive visualizations on the DBLP dataset that helps analyze the frequencies of various keywords. In this work, however, we attempt to go much further and present a more detailed analysis of the keywords and also link the keywords with the area of research and the country where the research has been conducted.

Objective

In this project, we aim to build interactive software to visualize insights about key-terms in publication titles. Specifically, we would like to explore the following questions:

1) Co-occurrence among these keywords - which terms occur together most often
2) Correlation of keywords with other aspects of articles - such as the research-domain of the article, or even the author's location/university, etc.
3) Term-Document clustering - what type of terms occurs most frequently in what type of documents
4) Evolution of research headline terms - a timeline of usage of research terms in titles

Our project would consist of three separate components, namely data-preparation, data analytics, and data visualization. We believe this analysis would help us to consider the possible correlated scientific words that may hint at related papers. We hope that our visualization would enable users to achieve typical literature search tasks faster.

Hypotheses and Testing

Our hypotheses or expectations are the following:

1) Certain terms are more likely to co-occur - like neural and backpropagation, or cluster and parallel, or cluster and K-Means, etc. This can be tested once we build and visualize the co-occurrence matrix of terms.
2) Certain terms tend to dominate in certain research areas more than certain others - such as 'index' and 'queries' in database-related articles, or 'neural' in Artificial Intelligence related articles. Yet others like 'data' may be common to many. This can

be tested after we perform our Latent Semantic Analysis to cluster terms and documents in a common space, and visualize it in a 2D scatter-plot.

3) It might be that there is a trend in the research related to certain domains, in the last 10 years - some topics may have faded out (say POS tagging) and some others have rejuvenated (like Neural Networks and Deep Learning). This can be tested by visualizing a trend-line graph for each research topic (so found in earlier steps).

Design Plan

The following are the general steps we'd like to follow in this project:

1. **Data Collection** - we have acquired the DBLP dataset [2], but might wish to augment more data with this one to draw out more interesting perspectives. We plan to do this using ORCID API.
2. **Data Preparation** - we might need preliminary NLP steps (such as stemming, stop-word removal, etc.)
3. **Data Analysis** - Several analyses may be required to be performed for each perspective we want to explore. Our primary focus will be to find co-occurrence of terms, documents, find the clusters they form in Low dimension concept space, etc.
4. **Data Visualization** - We need to visualize our results to clearly understand our analysis as well as make it interactive and aesthetically appealing for viewers.

We may reiterate these steps several times for each different perspective in our analysis.

Implementation

1. **Data Acquisition:** First, we gather the publications data from the DBLP dataset [2]. The dataset contains information about publications belonging to the following categories: article, book, thesis, proceedings. For this project, we plan to use the article's data as this file contains sufficient information for analyzing the keywords used in the computer science literature. After initial processing, we found a total of 14,955,220 keywords (without pre-processing) in all the *articles* data.

   Also, we plan to obtain the ORCID data [3] for the authors of the publications. We aim to leverage this data and find how the country with which the author is affiliated and the area of interest correlates with the publication keywords. This could help us understand the relationship between keywords and research area and also how the keywords and hence the research area related to the country at which the research has been conducted.

2. **Data Preprocessing:** We plan to first pre-process the data in the DBLP database to obtain a list of keywords from the titles of the publications. This would involve removing stopwords, punctuations and other commonly used words not relevant in the context of the computer science literature.

3. **Data Analysis:** Once the set of keywords has been obtained, we plan to:
   a) *Histogram*: Create a histogram of words and visualize the word frequencies in a particular time range and also overall.

   b) *LSA on term-document co-occurrence matrix*: In order to gain further insights into how the different key-words are grouped, we could perform LSA on the TF-IDF matrix. To compute this, we first compute the term occurrence matrix with rows being the key-words and columns being the titles of articles, and cell containing the term frequency in the title. Next, the TF-IDF matrix is computed on this matrix. We can perform LSA on this matrix to obtain the dominant clusters of the key-words and try to obtain a semantic understanding based on the different fields in computer science such as systems, artificial intelligence, multi-disciplinary areas, etc.

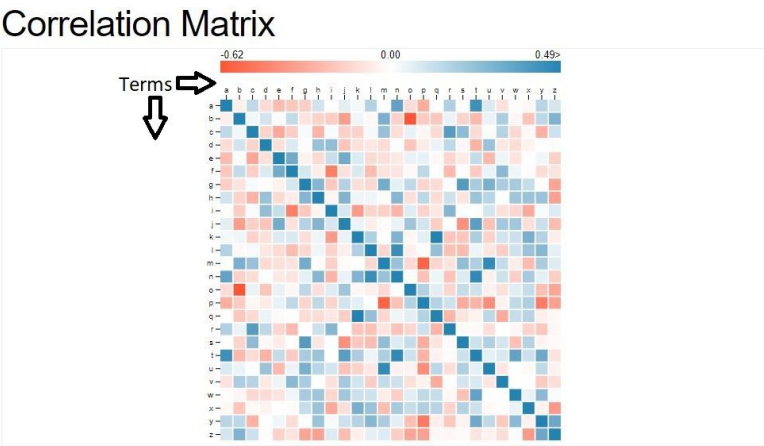
   c) Split the DBLP dataset into smaller subsets for different time ranges (e.g. a period of 5 years). Next, we can perform analytics on the individual time-based splits and observe how the keyword usage evolves with time.

4. **Interactive Visualization:**

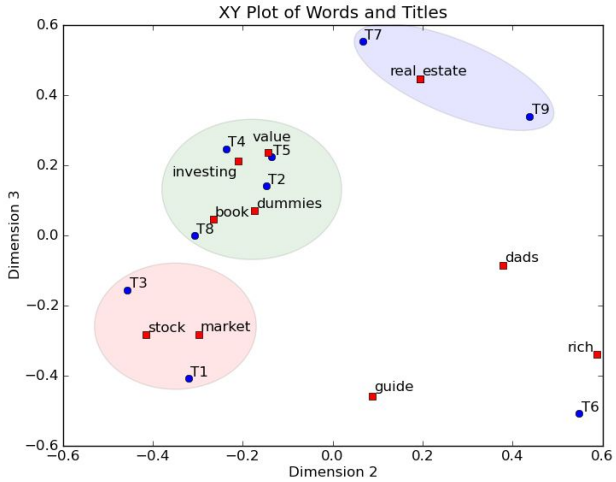
   Various interactive visualizations need to be prepared to show different the different perspectives mentioned above. As of now, we plan the following:

   a) **Word-cloud view**: a word-cloud visualization for term-frequencies

b) **Correlation Matrix**: a matrix visualization of the correlations between different keywords.



c) **Term-Document cluster visualization**: a scatter-based visualization of terms and documents clustered after LSA to visualize the k-D concept space.



d) **Term-similarity view**: Based on our results, we may create an interactive dynamic visualization of terms or domains similar to an input term.

e) **Term-usage trend view**: We can plot trend lines to visualize the term usage in research article headlines over ranges of time-intervals. We can make this interactive by allowing users to filter which terms they would like to see.

Estimated Timeline:

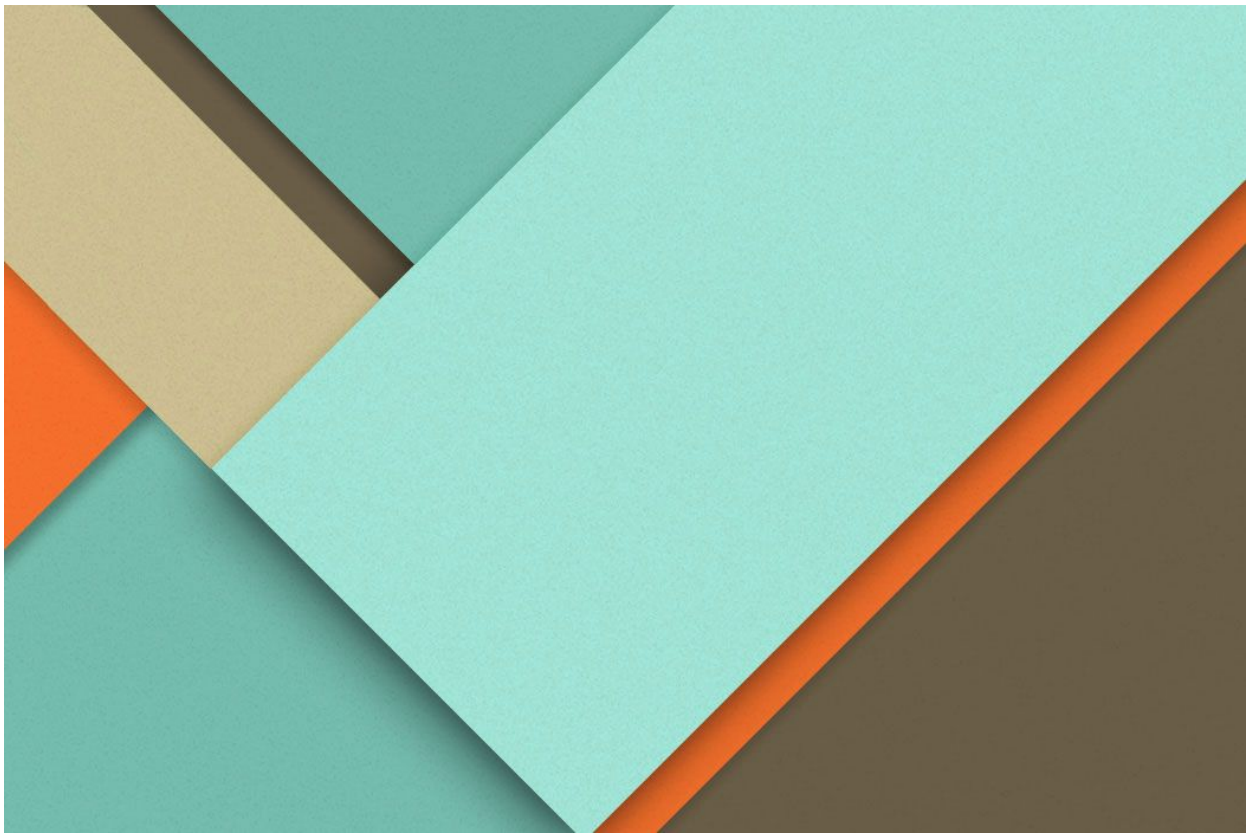
Task	Estimated time
Data Preparation Data Acquisition (including Augmentation) From other public datasets From public APIs Data Preprocessing Data cleaning, cross-linking, etc.	7-10 days
Developing and Tuning Analytical models Developing Charts for Visualization	10-15 days
Tuning and Enhancing the Visual UI Preparing Final Presentations Wrap-up	5 days

References:

1. Michael Burch, Daniel Pompe, Daniel Weiskopf, “An Analysis and Visualization Tool for DBLP Data”, 19th International Conference on Information Visualisation, 2015.

2. DBLP dataset: https://dblp.uni-trier.de/xml/

3. ORCID API: https://orcid.org/organizations/integrators/API



CSE564: Project Prelim Report

04.30.2019

Group members:
Ananya Palit (ID: 112077303), Souradeep Chakraborty (ID: 112281492)
Computer Science Department
Stony Brook University

Data Preprocessing

Our DBLP dataset has 2,008,467 articles (and that many titles) with around 15 million words. We processed our data to generate titles and keywords leveraging the following:

1. **Parsing** - the original files were in XML format. We parsed these files using an open-source parser [1] to generate CSV files.
2. **Tokenization** - We used NLTK's [2] tokenizer to tokenize each title.
3. **Lemmatization** - We used the *Wordnet Lemmatizer* from NLTK for lemmatizing words (like 'calculi' is lemmatized to 'calculus'). This helps to derive word-roots and is particularly helpful for clustering as in our case (Eg: 'system', 'systems', 'systemic' are all reduced to 'system').
4. **No stemming** - However, **we opted not to use stemming** because we found that both the *PorterStemmer* as well as the *SnowballStemmer* were not sufficiently accurate to stem words. They work well with simple plurals (like 'cats' and 'classes'), but they crudely chop off suffixes in general (like 'bushfires' becomes 'bushfir' and 'simulation' becomes 'simul') as they do not guarantee that the words will be valid dictionary words. **Since we are visualizing the words, such invalid chopped-off words appearing in the charts would be awkward**.
5. **Stopword and Punctuation removal** - We removed common punctuations before tokenization, and we filtered stopwords (common words like 'using', 'will', etc.) from our dataset using [3] *Gensim STOPWORDS*.

After the initial pre-processing, we arrived at the following statistics:

No. of unique Titles	1,952,578
No. of unique Terms	448,771
Most popular terms	'system', 'network', 'model', 'analysis', 'algorithm', 'data'

As an output of preprocessing, the following files were generated for further analysis:

Keyword-counts	A map of each keyword and its frequency
Keyword-years	A map of each keyword and the list of years they were used in a title (in order of their usage)
Title-year-author	A map of each title and its year of publication and author's orc-id.
Title-keyword-map	A map of each title and the keywords it contains

Term Co-occurrence Analysis

After preprocessing the data, we did a co-occurrence analysis for the 100 most frequently occurring keywords (interchangeably used with 'term'). We implemented it as follows:

1. Using the 'Keyword-counts' dictionary we filtered the 100 most frequent terms
2. Then, we created a map of all possible pairs of distinct keywords from this set
3. Likewise, from the 'Title-keyword-map', for each title
    a. we created all possible pairs of distinct keywords
    b. If they were in the pair-map created above, their count was incremented

    (Co-occurrence of a keyword with itself represents no. of titles they occurred multiple times)

The final result of this was a 100x100 matrix of the co-occurrence counts of the 100 most popular keywords. A sample of it is shown below:

	system	network	model	analysis	algorithm	data	method	approach	problem	application	...
system	3717	7639	8918	11606	6256	5962	6327	8270	2660	7208	...
network	7639	4050	8339	10373	9344	7429	3653	6834	3027	4820	...
model	8918	8339	1740	6362	3483	5616	3313	3598	1924	4392	...
analysis	11606	10373	6362	853	4365	8615	5359	4075	2039	4288	...
algorithm	6256	9344	3483	4365	765	3888	2032	1662	10158	3554	...
data	5962	7429	5616	8615	3888	2833	3717	4230	888	3532	...
method	6327	3653	3313	5359	2032	3717	824	1123	6722	3514	...
approach	8270	6834	3598	4075	1662	4230	1123	96	3882	2055	...
problem	2660	3027	1924	2039	10158	888	6722	3882	769	2552	...
application	7208	4820	4392	4288	3554	3532	3514	2055	2552	197	...
control	19215	7751	4597	2405	2314	1239	2198	3145	2297	2797	...
learning	4118	5255	3844	1663	3056	2849	1872	3397	960	1765	...
design	12121	5505	2532	4292	2551	1282	1910	3203	1199	2865	...
image	1951	2748	3009	3409	3569	1665	3561	2306	236	2331	...
information	11386	3321	2962	2203	814	2005	1329	2204	599	1455	...

This was saved for later use. Also, there are a few outliers, the matrix of the logarithm of these values was also saved.

Topic Modeling

A major part of our analytics was **topic-modeling - clustering terms and titles into topics**. We began by loading the preprocessed 'keyword-counts' and 'title-keyword-map'. We performed two kinds of data models to build the corpus for our analytics:

1. Bag of Words (BOW)

We used Gensim's *Dictionary* to form a set of keywords from the title-keyword-map, and filtered extremes like words occurring in fewer than 15 or more than 50% of the titles. Then, we used Gensim's *doc2Bow* function on each title-keyword-set to create the Bag-of-words (BOW) for it and built the **BOW-corpus** using all the titles.

2. TF-IDF

We used Gensim's *TfidfModel* on the BOW-corpus to form **TF-IDF-corpus** for all the titles. Unlike a bag of words, TF-IDF models represent the words by their TF-IDF scores.

Latent Dirichlet Allocation (LDA)

We then proceeded to the topic-clustering on each of these corpora using the **Latent Dirichlet Allocation (LDA)**. We chose LDA over LSA (Latent Semantic Analysis), as the latter cannot handle various issues like Polysemy (***our data has such polysemy in words like 'graph'***), while the former can not only handle these but also assign probabilities for each term and title to belong to a certain topic (or cluster).

We used Gensim's *LdaMulticore* model on each corpus with 10 topics to generate each case.

Current Results and Testing

Following is the view of topics generated by the LDA model on the BOW-corpus:

```
Topic: 0
Words: 0.060*"graph" + 0.022*"set" + 0.021*"recognition" + 0.016*"function" + 0.015*"space" + 0.014*"number" + 0.012*"pattern"
+ 0.010*"algorithm" + 0.009*"theorem" + 0.009*"group"
Topic: 1
Words: 0.055*"network" + 0.025*"system" + 0.020*"wireless" + 0.020*"sensor" + 0.017*"scheme" + 0.015*"based" + 0.014*"channel"
+ 0.014*"communication" + 0.013*"performance" + 0.013*"power"
Topic: 2
Words: 0.047*"image" + 0.021*"based" + 0.021*"classification" + 0.019*"feature" + 0.016*"selection" + 0.014*"method" + 0.013*"a
utomatic" + 0.012*"point" + 0.011*"3d" + 0.011*"detection"
Topic: 3
Words: 0.020*"system" + 0.017*"design" + 0.017*"issue" + 0.016*"special" + 0.015*"test" + 0.013*"analysis" + 0.013*"computer" +
0.012*"noise" + 0.012*"measurement" + 0.011*"simulation"
Topic: 4
Words: 0.062*"system" + 0.038*"control" + 0.021*"nonlinear" + 0.020*"linear" + 0.019*"method" + 0.017*"solution" + 0.017*"clas
s" + 0.013*"equation" + 0.013*"problem" + 0.013*"order"
Topic: 5
Words: 0.049*"algorithm" + 0.036*"problem" + 0.030*"optimization" + 0.024*"network" + 0.022*"time" + 0.019*"model" + 0.018*"rob
ot" + 0.017*"dynamic" + 0.015*"neural" + 0.012*"approach"
Topic: 6
Words: 0.023*"code" + 0.019*"algorithm" + 0.019*"tree" + 0.016*"search" + 0.016*"complexity" + 0.014*"machine" + 0.014*"random"
+ 0.013*"property" + 0.012*"polynomial" + 0.012*"vector"
Topic: 7
Words: 0.030*"learning" + 0.019*"service" + 0.018*"system" + 0.017*"environment" + 0.016*"mobile" + 0.015*"virtual" + 0.013*"ne
twork" + 0.012*"user" + 0.012*"game" + 0.011*"web"
Topic: 8
Words: 0.044*"system" + 0.037*"data" + 0.020*"analysis" + 0.020*"software" + 0.018*"model" + 0.017*"approach" + 0.016*"process"
+ 0.015*"distributed" + 0.012*"design" + 0.012*"language"
Topic: 9
Words: 0.031*"study" + 0.019*"information" + 0.016*"technology" + 0.016*"case" + 0.016*"research" + 0.011*"use" + 0.010*"socia
l" + 0.009*"review" + 0.009*"science" + 0.009*"impact"
```

A similar view was obtained for the LDA model on the TF-IDF corpus as well. **The semantic coherence is better for the BOW-corpus than the TF-IDF corpus as of now.**

Since there are no particular scores for an unsupervised model such as this, ***we manually tested the predictions of the model using some titles (both seen and unseen)***. An example is shown below for the title 'TensorFlow - A system for large-scale machine learning':

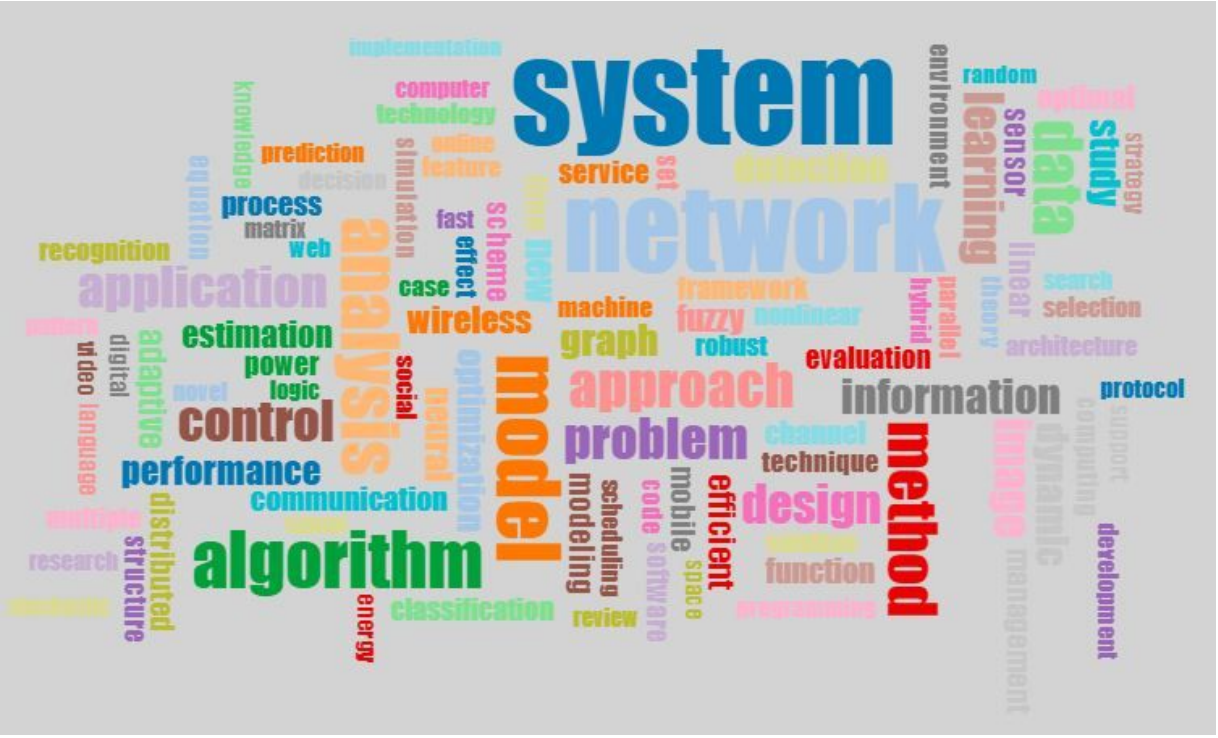
```
Score: 0.533555805683136
Topic: 0.049*"algorithm" + 0.036*"problem" + 0.030*"optimization" + 0.024*"network" + 0.022*"time" + 0.019*"model" + 0.018*"rob
ot" + 0.017*"dynamic" + 0.015*"neural" + 0.012*"approach"

Score: 0.19465483725070953
Topic: 0.060*"graph" + 0.022*"set" + 0.021*"recognition" + 0.016*"function" + 0.015*"space" + 0.014*"number" + 0.012*"pattern"
+ 0.010*"algorithm" + 0.009*"theorem" + 0.009*"group"

Score: 0.19399428367614746
Topic: 0.023*"code" + 0.019*"algorithm" + 0.019*"tree" + 0.016*"search" + 0.016*"complexity" + 0.014*"machine" + 0.014*"random"
+ 0.013*"property" + 0.012*"polynomial" + 0.012*"vector"

Score: 0.011116203851997852
Topic: 0.020*"system" + 0.017*"design" + 0.017*"issue" + 0.016*"special" + 0.015*"test" + 0.013*"analysis" + 0.013*"computer" +
0.012*"noise" + 0.012*"measurement" + 0.011*"simulation"

Score: 0.011115114204585552
Topic: 0.047*"image" + 0.021*"based" + 0.021*"classification" + 0.019*"feature" + 0.016*"selection" + 0.014*"method" + 0.013*"a
utomatic" + 0.012*"point" + 0.011*"3d" + 0.011*"detection"

Score: 0.011114641092717648
Topic: 0.044*"system" + 0.037*"data" + 0.020*"analysis" + 0.020*"software" + 0.018*"model" + 0.017*"approach" + 0.016*"process"
+ 0.015*"distributed" + 0.012*"design" + 0.012*"language"

Score: 0.011112382635474205
Topic: 0.030*"learning" + 0.019*"service" + 0.018*"system" + 0.017*"environment" + 0.016*"mobile" + 0.015*"virtual" + 0.013*"ne
twork" + 0.012*"user" + 0.012*"game" + 0.011*"web"
```

The most likely topic for this is the one with terms like ‘algorithm’, ‘optimization’, ‘network’, ‘model’, ‘neural’, etc. which seems to be a decent prediction.

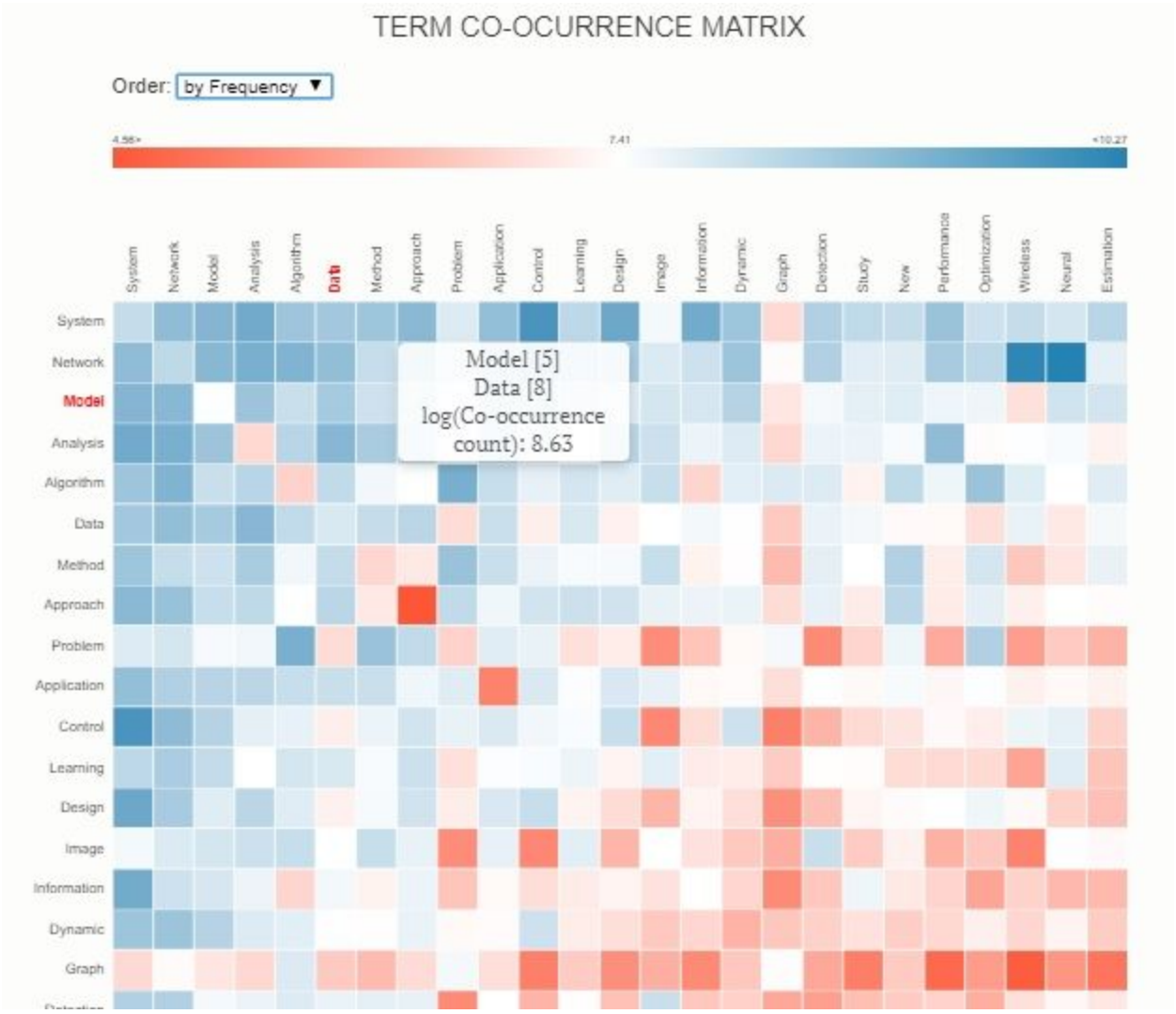
Visualizations

We have so far prepared two d3.js charts for visualization. Both these are planned to be part of a UI dashboard.

1. **Word-Cloud View** - the size of words is proportional to their frequency



2. **Co-occurrence Matrix** - a matrix showing the co-occurrence of the 25 most frequently occurring terms.
   - Since these values are skewed, they are transformed to logarithmic scales.
   - The color of a cell encodes this value scaled from the lowest (red) to highest (blue).
   - Hovering on any cell shows this value for the pair of terms as well the topic-number that they belong to (in square brackets).
   - The drop-down menu on the top-left allows users to sort this chart in 3 orders: 'name' of the term (alphabetically), 'frequency' (the view shown here), and 'topic' (the topic-group they belong to, showing the clusters). The reordering is animated in d3.



We plan to build a third chart - a scatter-plot to show the topic clusters and build a dashboard showing all 3 charts together.

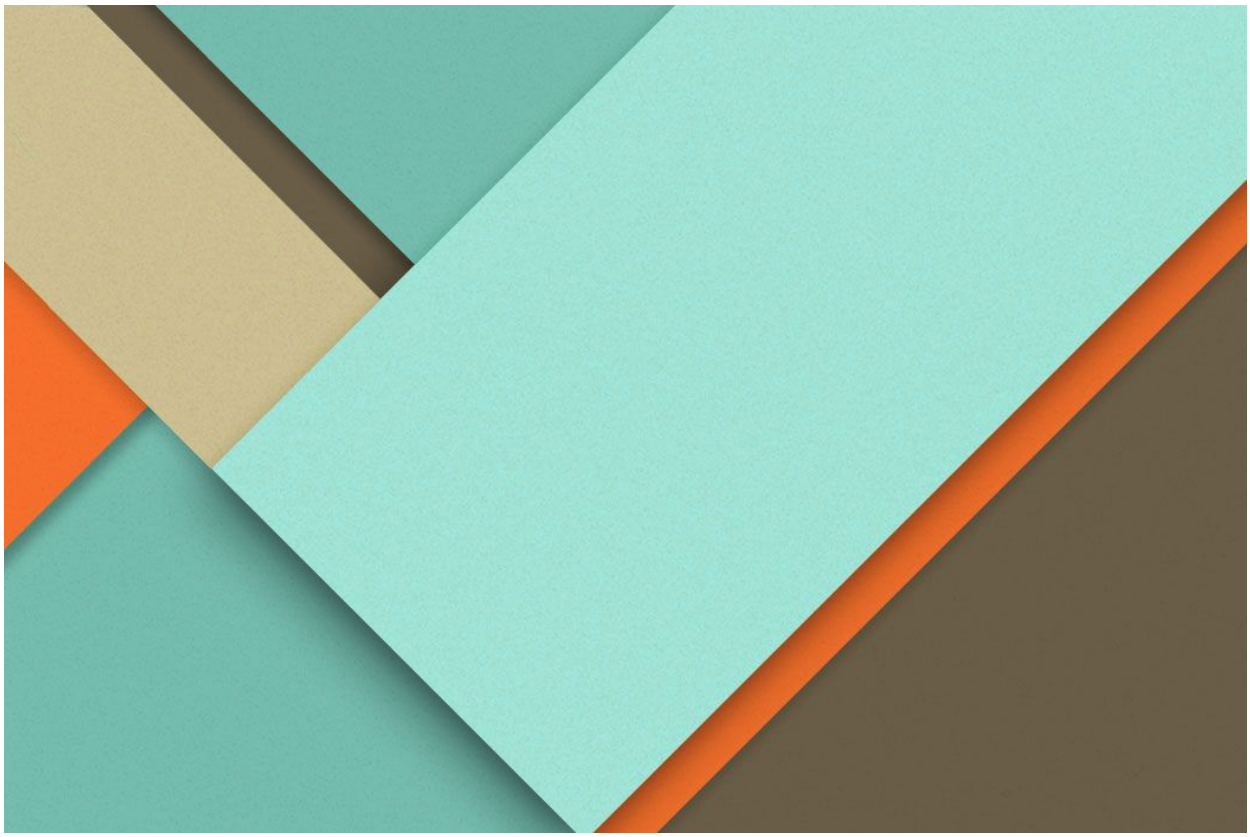
Remaining Work and Plan

Our main focus so far has been data-preparation and analytics, especially topic-modeling. For the remainder, our tentative plan and work items are as follows:

1. Time-series modeling - as per the proposal, we would compute the popularity of titles and keywords across time.
2. Enhancing the existing charts and building new ones - we plan to improve parts of our existing charts and build new ones (eg: a chart for Topic-cluster and one for Time-series modeling).
3. Creating a dashboard view for topic-modeling - along with the existing 2 charts we plan to incorporate the topic-cluster scatter-plot chart into a dashboard view allowing cross-linking and cross-filtering.
4. [Optionally] We are working to get authors' data from the ORCIDs. If sufficient data is available, we may present an analytics on authors' publications.
5. [Optionally] Enhancement of topic models - we may try to enhance the existing models for better semantic coherence.

References:

1. DBLP Parser https://github.com/ThomHurks/dblp-to-csv
2. NLTK: http://www.nltk.org/api/nltk.html
3. Gensim: https://radimrehurek.com/gensim/
4. Topic Modeling in Python https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24
5. Word-cloud Visualization: https://bl.ocks.org/jyucsiro/767539a876836e920e38bc80d2031ba7
6. Correlation Matrix visualization: https://bl.ocks.org/HarryStevens/302d078a089caf5aeb13e480b86fdaeb
7. Co-occurrence Matrix visualization: https://bost.ocks.org/mike/miserables/



CSE564: Final Project Report

05.22.2019

Group members:

Ananya Palit (ID: 112077303), Souradeep Chakraborty (ID: 112281492)

Computer Science Department

Stony Brook University

Focus of analysis

Our aim for the project has been to explore the evolution of Computer Science research topics and keyword usage. Hence our project title **"It's all in the keywords"**. Our principal hypotheses are:

1. Certain keywords should be very dominant in the CS literature, with certain pairs also being very common to co-occur.

2. Keywords usage in the titles represent the main topic of the research in the paper. Also the topics that keywords are associated with change with time due to their usage patterns.

3. The overall volume of publications have increased exponentially over the years. Also certain topics of research have grown or reduced in popularity.

4. Additionally, we increased our scope to explore the correlation in research publications of SBU CS faculty, to validate our premise that researchers with similar areas of interest publish titles using similar keywords representative of that topic.

We gathered and analysed data from DBLP dataset (as mentioned earlier) containing about 2 million article titles (and other metadata) with about 0.5 million keywords (found in preprocessing). Our analysis details and visualization are detailed in the following sections.

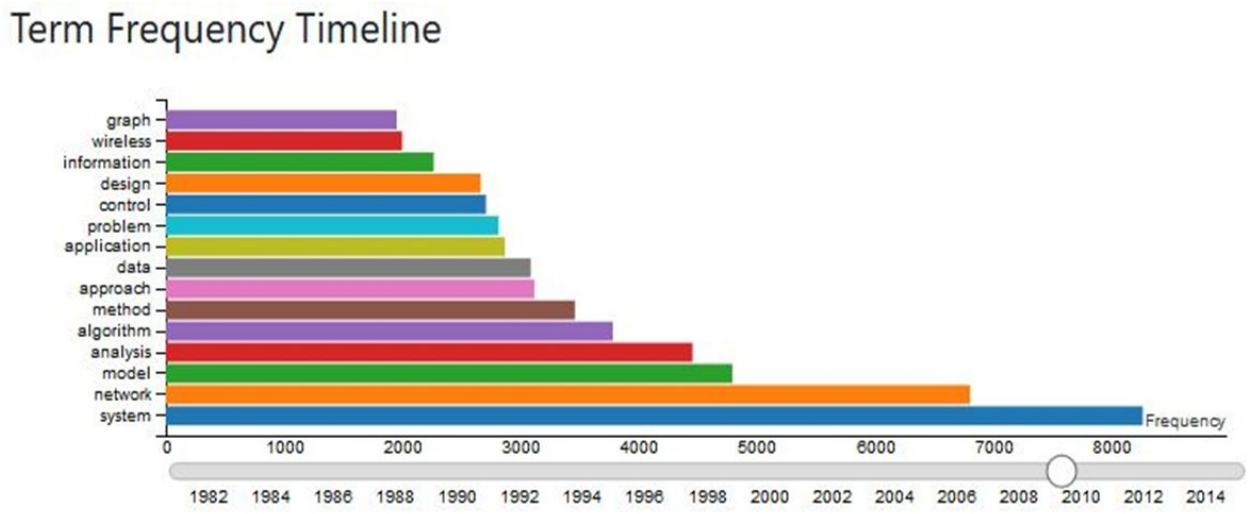
Term Frequency Analysis

1. **Most dominant terms overall**

   We presented a **word-cloud chart** to visualize which are the most popular keywords used in titles overall. The top-25 is shown in the chart, with the size of words proportionate with their frequencies.

2. Evolution of Term usage

A **horizontal bar-chart** visualizes the time-varying popularity of keywords in article titles. The bar-length shows the frequency of usage of a term, and the slider at the bottom can be used to check the top 10 popular terms for any year. This helped us figure out the evolution of terms usage and that certain terms have grown much more popular in article headlines over the years.

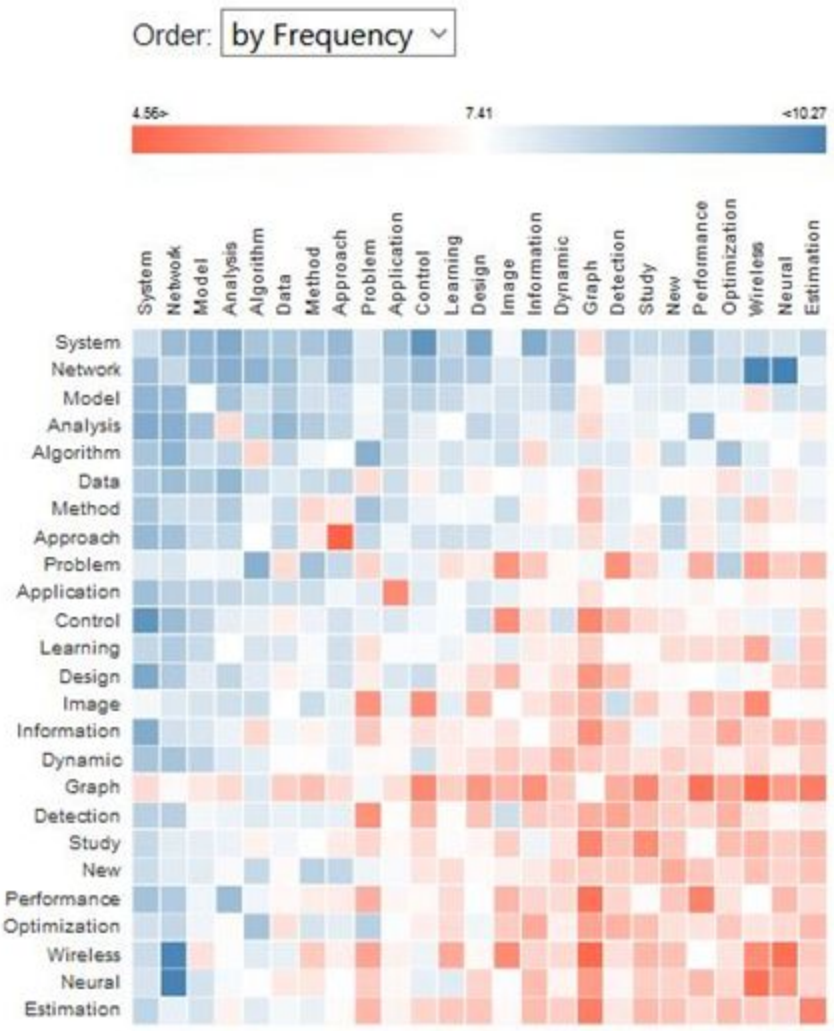


3. Co-occurrence

We show a **co-occurrence matrix** - a matrix showing the co-occurrence of the 25 most frequently occurring terms.

- Since these values are skewed, they are transformed to logarithmic scales.
- The color of cells encode this value scaled from the lowest (red) to highest (blue).
- Hovering on any cell shows this value for the pair of terms as well the topic-number that they belong to (in square brackets).

- The drop-down menu on the top-left allows users to sort this chart in 3 orders: ‘name’ of the term (alphabetically), ‘frequency’, and ‘topic’ (the topic-group they belong to, showing the clusters). The reordering is animated in d3.



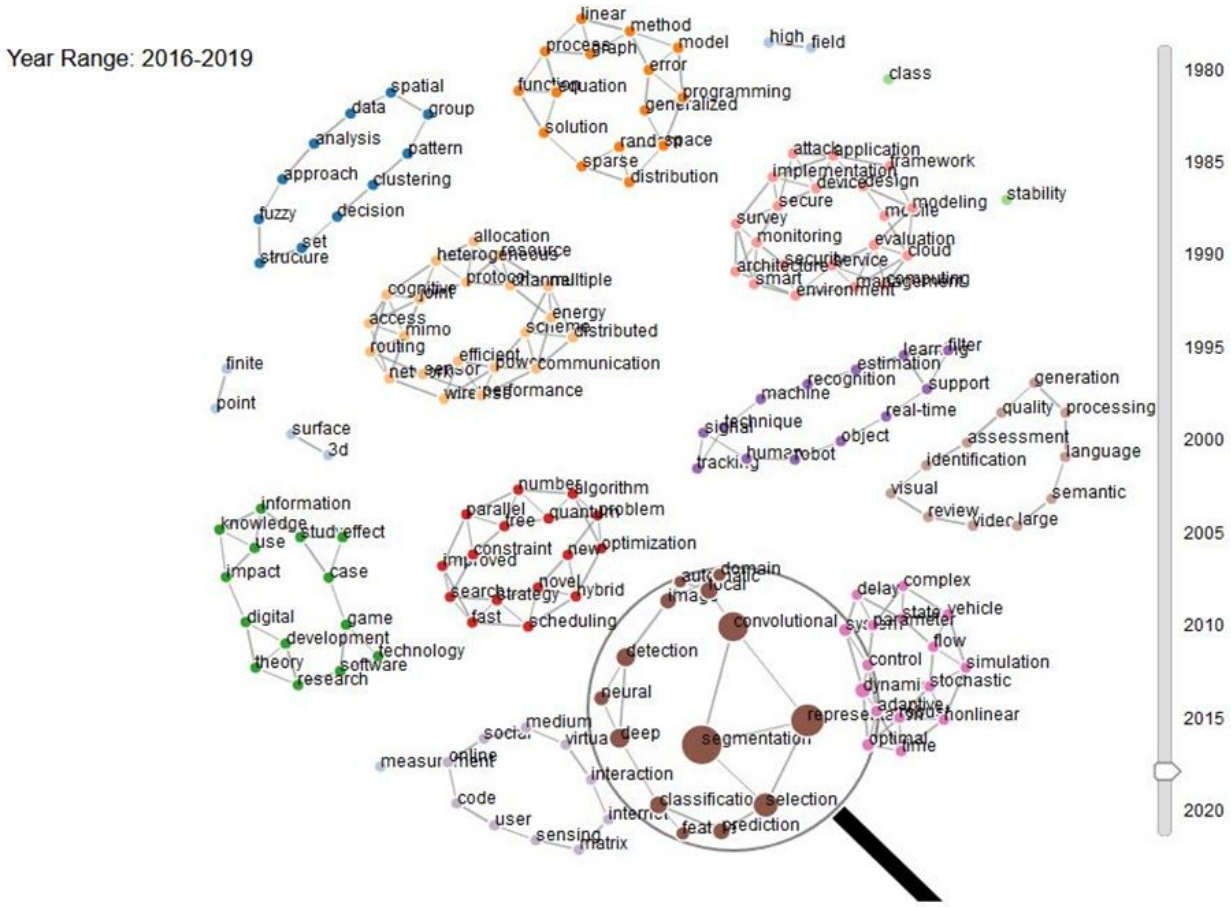
Topic Modeling

1. **Methodology**

   As mentioned before, we have tried a number of word-representations like Bag-of-words model and the TF-IDF model as an input to the **Latent Dirichlet Allocation (LDA)** model (using the LDAMulticore API of gensim library). The Bag-of-words yielding better results was our final choice for word-representation. We trained the model for 10 topics and displayed the top-10 terms for each topic. From these terms, we figured out the most likely area of research that each cluster represents; (we also use this to label the data in the charts of the dashboard view).

2. Visualizing clusters

We chose a **Graph with Fisheye lens distortion** to best represent the clusters of keywords. The keywords are colored as per the clusters they represent at a time. The magnifying lens can be moved across the chart to see the keywords bigger.



3. Changing clusters over time

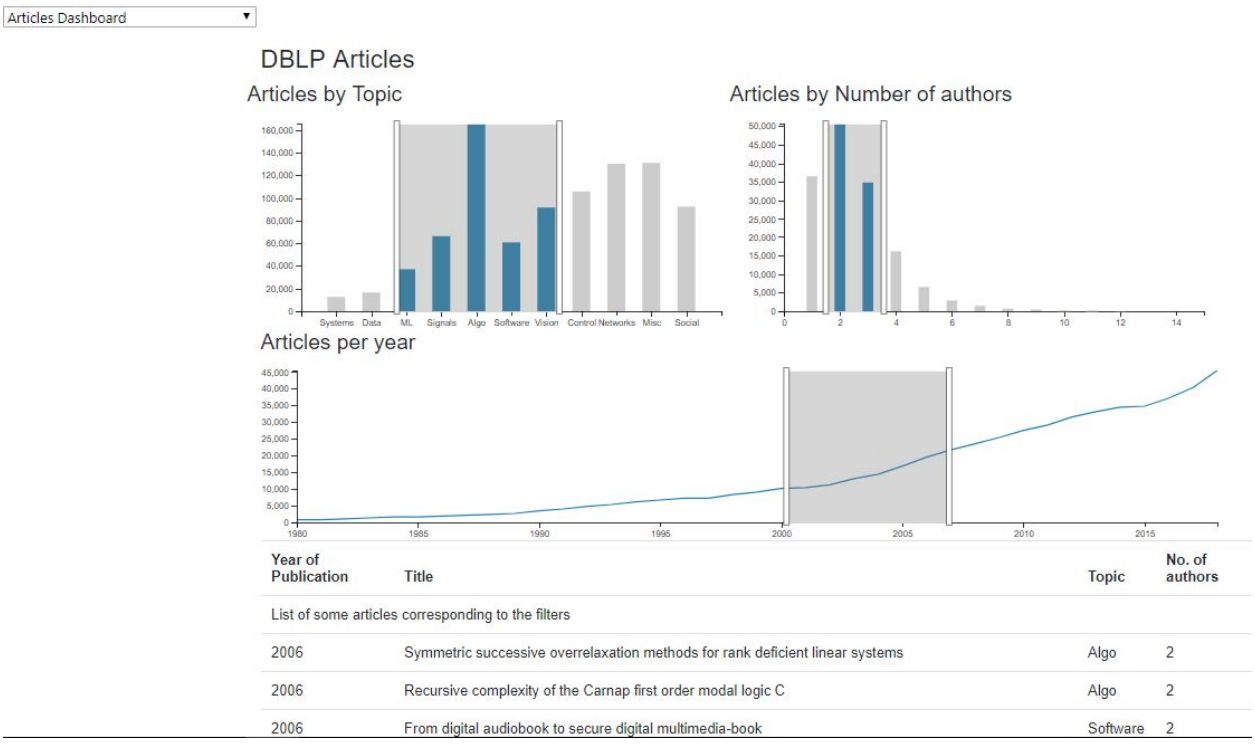
One thing we noted that these clusters of keywords that represent a topic can change with time. An interesting insight would be to see which keywords most prominently represent a topic and also whether keywords change in correlation with more different topics across time. We ran the topic-modeling for each 5-year span from 1980 till 2019, and provided a vertical slider in this chart to explore this.

Evolution of Topics - Trend analysis

The Dashboard view

We have prepared a visual dashboard to explore the trends in the topic-wise publications - which ones have grown more popular and which ones have reduced over the years. Also, we explored the trend in number of authors per article.

Since this is a co-ordinated view, users can cross-filter among the charts on the dashboard selecting subsets of data of their interest and visually querying the data to discover trends and unusualities. For example, a user can select a single topic (like Computer Vision) in the articles-per-topic chart, select 1 in the authors-per-article chart and the period between say, 2010 till the end in the timeline chart to visually query the no. of single-author articles in the field of Computer Vision since 2010. A data-table at the bottom shows the latest 10 article titles from this set for reference.



Year of Publication	Title	Topic	No. of authors
List of some articles corresponding to the filters			
2006	Symmetric successive overrelaxation methods for rank deficient linear systems	Algo	2
2006	Recursive complexity of the Carnap first order modal logic C	Algo	2
2006	From digital audiobook to secure digital multimedia-book	Software	2

Correlations in publications among SBU CS Faculty

Data and Methodology

In order to obtain the relationships among the SBU CS faculties in terms of their publications, we first listed the names of the faculty members (our dear professors!) from the SBU CS website [10]. Thereafter, we collected their publication data from their corresponding Google Scholar profile webpages. We considered Google Scholar for our analysis as the publication data for every faculty member could be obtained very conveniently using a python based API, Scholarly [11] for obtaining Google Scholar data. In total, we could collect data for 42 faculty members. The data for some of the members could not be located in Google scholar and hence, we have not considered their publication data in our analysis.

After the initial filtering step, as discussed above, we obtained the keywords from the titles of the publications listed in the profile pages for every faculty. Next, we lemmatized the keywords obtained and thus formed lists of keywords for every faculty member. Let's denote the list of keyword for every faculty member i by L_i. Following this step, we computed the pairwise similarity in the publications for every member pair, *S(i,j)* by computing the number of keyword co-occurrences in the lists (L_i) obtained in the previous step. We divided the total number of co-occurrences, *C(i,j)* in the lists by the mean value of publications of the two faculties involved in the particular pair. We can express the similarity score as follows:

$$S(i,j) = C(i,j) \ / ((N(i) + N(j))/2)$$

Next, we normalized the similarity scores for every pair by dividing the score by the row summation of the scores corresponding to a faculty. The method used for normalization is as follows:

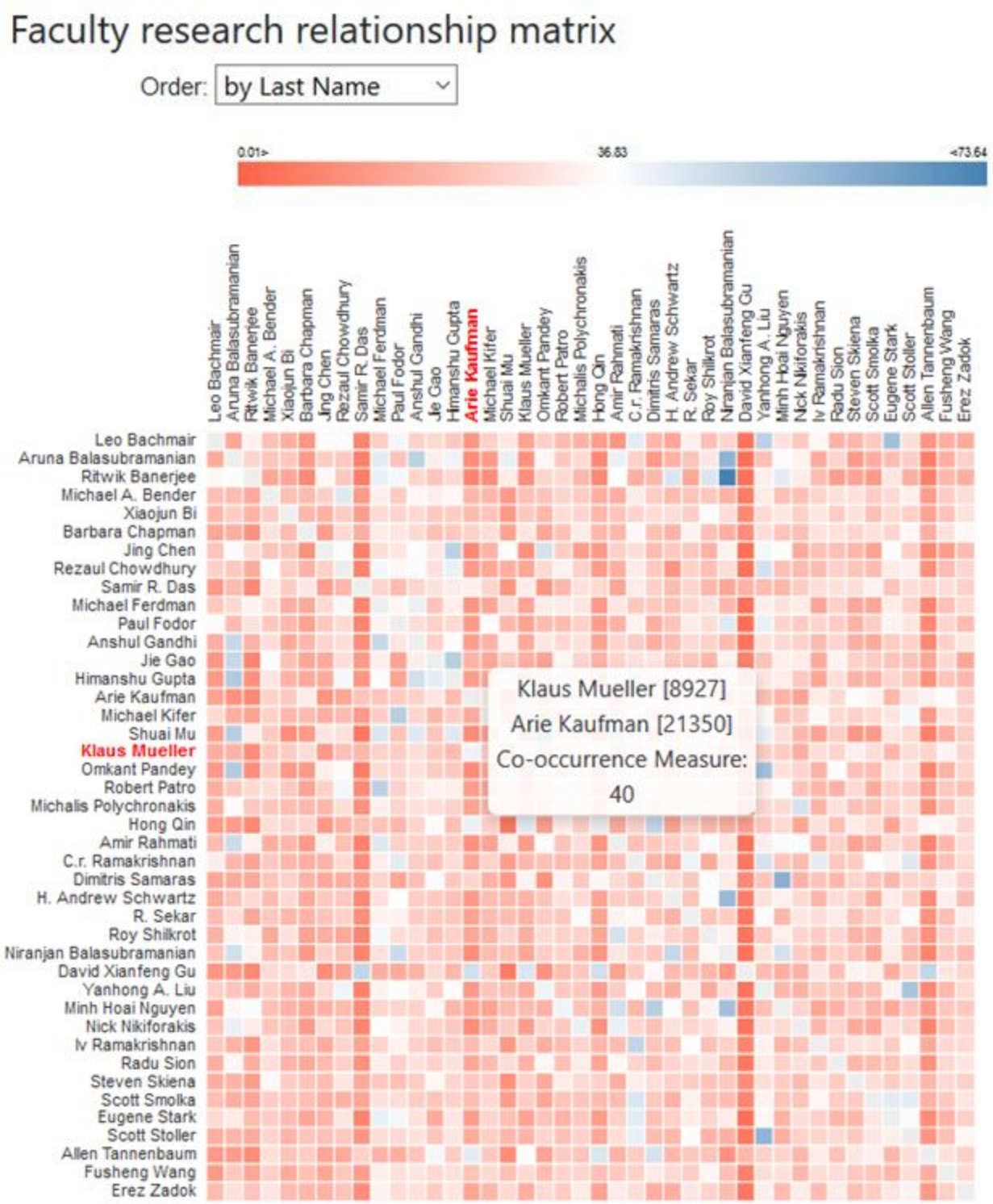
$$S'(i,j) = S(i,j) \ / \sum S(i,j), \text{ where } \sum S(i,j) \text{ indicates summation over all } j$$

The normalized scores, *S'(i,j)* were used since the variation in the number of publications, N_i of every faculty was huge (often involving a big difference, e.g. N_i = 23 vs. N_j=2310) and the similarity metric would only provide us a meaningful insight if an adequate normalization technique is applied to it. Thus, from the row wise analysis of scores, for every faculty, we can gain insight about the correlation of their works with other faculties.

After obtaining the normalized pairwise similarity scores, we plotted the scores using the keyword co-occurrence matrix, which is some sense, could be considered as being the research correlation among the faculty members. In the next sections, we discuss the insights we obtained from the above computations.

Correlation Matrix

The similar chart as used in the first view to represent the term co-occurrence. However, this time, we show the citation count of each faculty in brackets and allow sort by name and citation-count.



Sample results

Some of the correlations we explored are as follows:

Prof. Klaus Mueller's publication titles are most correlated with those of Prof. Arie Kaufman and Prof. Hong Qin.

Similar correlations were found between - Prof. Dimitris Samaras and Prof. Minh Hoai Nguyen; Prof. Steven Skiena, Prof. Michael Bender and Prof. Jie Gao; Prof. Ritwik Banerjee and Prof. Niranjan Balasubramanian, and so on for many others.

Demonstration video link: https://youtu.be/AgQBmTDIjJo

Key Insights and Conclusions

1. Some keywords like 'system' and 'network' have been dominant keywords in titles throughout time. The word-cloud shows this.

2. Some keywords like 'data' and 'learning' have risen in popularity in the last decade. The term-frequency timeline chart reveals this.

3. Some keywords frequently co-occur with some others - like 'data' and 'analysis', or 'network' with 'neural', or 'wireless', and so on. The co-occurrence matrix shows this.

4. Title Keywords alone can often represent topics of research. For example, the words such as 'image', 'detection', 'feature', 'object', 'video', etc. all cluster together (as seen in the cluster-graph). This most likely represents the topic of computer vision.

5. The cluster graph also helps to visualize the change in keywords usage in different topics over time. For example, the words 'segmentation', 'convolutional', and 'neural' begin to appear in the 'vision' topic-cluster in the last 5-year period. This reveals the new kinds of techniques used in this topic nowadays.

6. Vision and Machine Learning related research has indeed risen dramatically since 2015. This was found out from a visual-query on the dashboard by filtering only the

these topics in the topic-chart and seeing the trend-line graph below which shows exponential growth in number of publications in this area.

7. Overall, 'Algorithm' related research was however the most dominant throughout. This can be seen easily in the topics-chart on the dashboard.

8. The dashboard also reveals that single-author papers predominate areas like 'Social computing' and 'Miscellaneous' literature (like reviews and case-studies).

9. From the correlation matrix of SBU CS faculty, we already mentioned a few examples earlier, which prove that there may be some correlation in titles of articles published by faculty of common areas of interest.

These insights and revelations prove not only all the hypotheses that we had postulated in the beginning, but also reveal more than we had imagined, such as single-author papers dominating the 'Social computing' area. Leveraging the power of visual analytics, we have found these, and we are sure that even more insights can be explored with further lookup.

References:

1. DBLP Parser https://github.com/ThomHurks/dblp-to-csv
2. NLTK: http://www.nltk.org/api/nltk.html
3. Gensim: https://radimrehurek.com/gensim/
4. Topic Modeling in Python https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24
5. Word-cloud Visualization: https://bl.ocks.org/jyucsiro/767539a876836e920e38bc80d2031ba7
6. Correlation Matrix visualization: https://bl.ocks.org/HarryStevens/302d078a089caf5aeb13e480b86fdaeb
7. Co-occurrence Matrix visualization: https://bost.ocks.org/mike/miserables/
8. Fisheye Lens distortion on graph: http://bl.ocks.org/fernoftheandes/8637581
9. Dashboard using d3.js, dc.js, crossfilter.js http://bl.ocks.org/d3noob/6077996
10. SBU CS Faculty members https://www.cs.stonybrook.edu/people/faculty
11. Scholarly Python API https://pypi.org/project/scholarly/