Project Name: PapeRank

Needle in a Data Haystack Final Project

Team Members:

Roy Zohar – roy.zohar.40@gmail.com – 209896174, roy_zohar

Yoav Tamir – yoavtamir00@gmail.com – 322291519, yoav.tamir

Overview:

Our Project's goal was to develop a tool that helps researchers find relevant and influential papers. On the practical side, we wanted to take a huge amount of academic papers, and be able to search and categorize them, in order to find important data.

Data:

We used the ***Semantic Scholar*** database, which contains a whopping 40 million academic papers. Each academic paper contains lots of raw data regarding the paper. The total size of the database is 87GBs. This database is open-source, and we downloaded it off the semantic scholar website: https://www.semanticscholar.org/

Our solution:

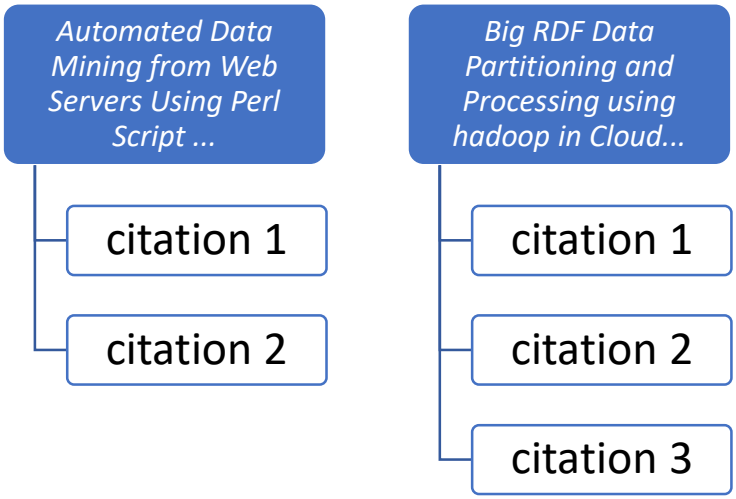
The main stages of our project were:

1. We processed the entire database, and organized it in databases that are convenient to use.
2. We ran a PageRank algorithm on the papers, where instead of edges we used citations.
3. We clustered the most important papers according to keywords from the abstract of each paper.
4. We implemented a heuristic search through the database, similar to Google Search, that takes into account both importance and matchness.
5. Content Based Research partner recommender system.

1. Processing the Database:

The raw data we used was organized in 40 huge JSON files. Obviously, our data cannot be kept all at once in the RAM. Therefore, we first off built 2 SQL databases, that act as dictionaries, that map each paper to a unique index identifier and vice versa.

We noticed that the SQL databases weren't running in reasonable enough time. After trying a few tools and running some tests, we found that numpy's memmap tool was the most efficient option. We created a few memmap arrays, that represent the edge matrix of citations. Since (40*10^6)^2 was more memory than we could allow ourselves, we couldn't save a 2-dimensional array in memory as is. Our solution



here was to use the sparsity of the matrix, and store the edges in a sort of array of linked lists.

We created many more databases throughout our work, that are built on the same principle. We found the experience of dealing with "Big Data" challenging but interesting.

2. Running Page Rank

Next, we ran the PageRank algorithm on all the academic papers. The idea behind using PageRank, is that a paper is more important in our opinion if has been heavily

cited, especially by other important papers. The algorithm is identical to the one we saw in class.

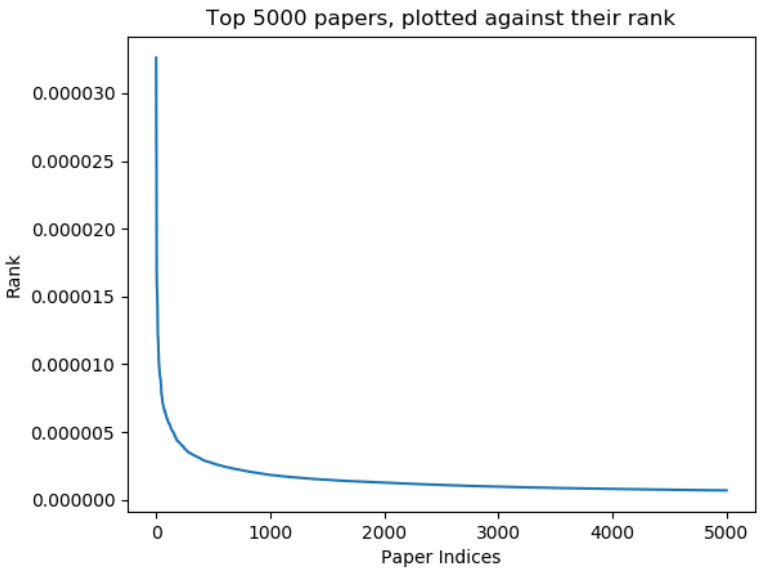
As a sanity check, we looked at the most important paper to see if it seems like an important article. The algorithm outputted:

Computers and Intractability: A Guide to the Theory of NP-Completeness

M. R. Garey, David S. Johnson • Published 1979

Which makes perfect sense, since it has been tagged on Semantic Scholar as “highly influential”, and has been cited a whopping 41475 times.

We also plotted the top 5000 papers’ ranks, and got the following figure:

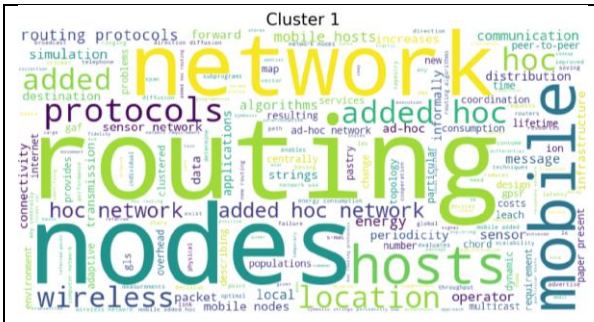


What can we learn from this visualization? There is a small group of papers that are extremely significant by a few orders of magnitude than most of the papers. This fits our understanding of the academic world – there are tons of papers but only a select few are truly groundbreaking.

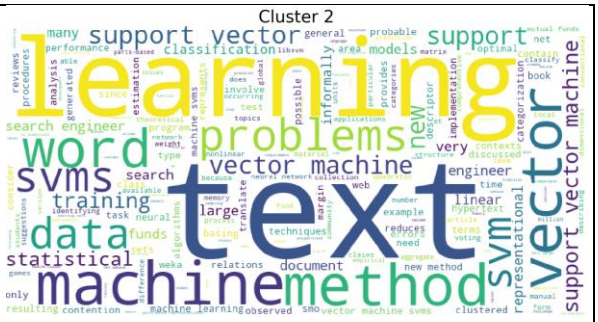
We also noticed that a lot of papers got the same, minimal ranking. This also makes sense, since they are probably papers with no in-citations (“end nodes”)

3. Top Article Clustering

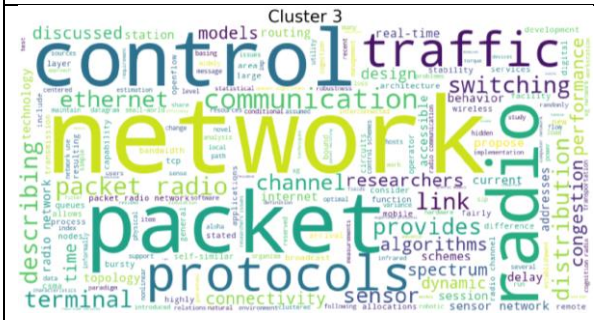
In order to understand our academic article space better, we decided to cluster the most important papers (based on their PageRank) by their abstract introduction. This way, we can get a sense of the topics in our dataset. We clustered them with K-means, using a cosine similarity and the Bag of Words model. We then visualized each cluster by creating a word tag cloud for each cluster, in order to get a sense of the topic of the cluster. For example, when running the algorithm with K=20, we got the following clusters:



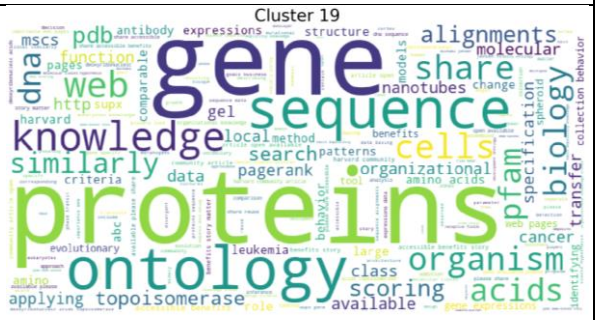
Cluster of articles regarding **communication networks**



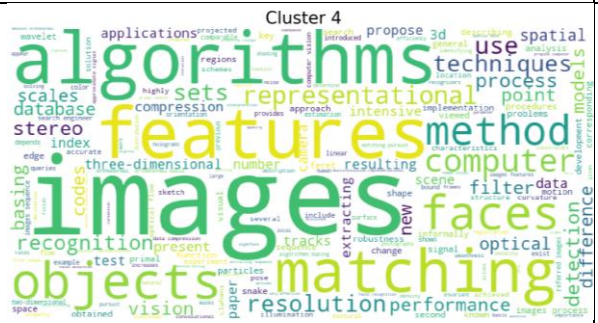
Cluster of articles regarding **learning algorithms for NLP**



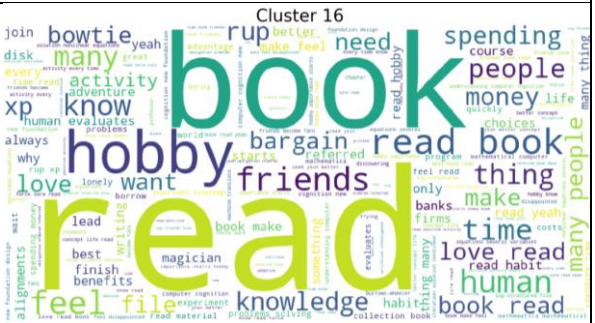
A slightly different cluster of articles regarding **communication networks**



Cluster of articles regarding **Biology**



Cluster of articles regarding **Computer Vision**



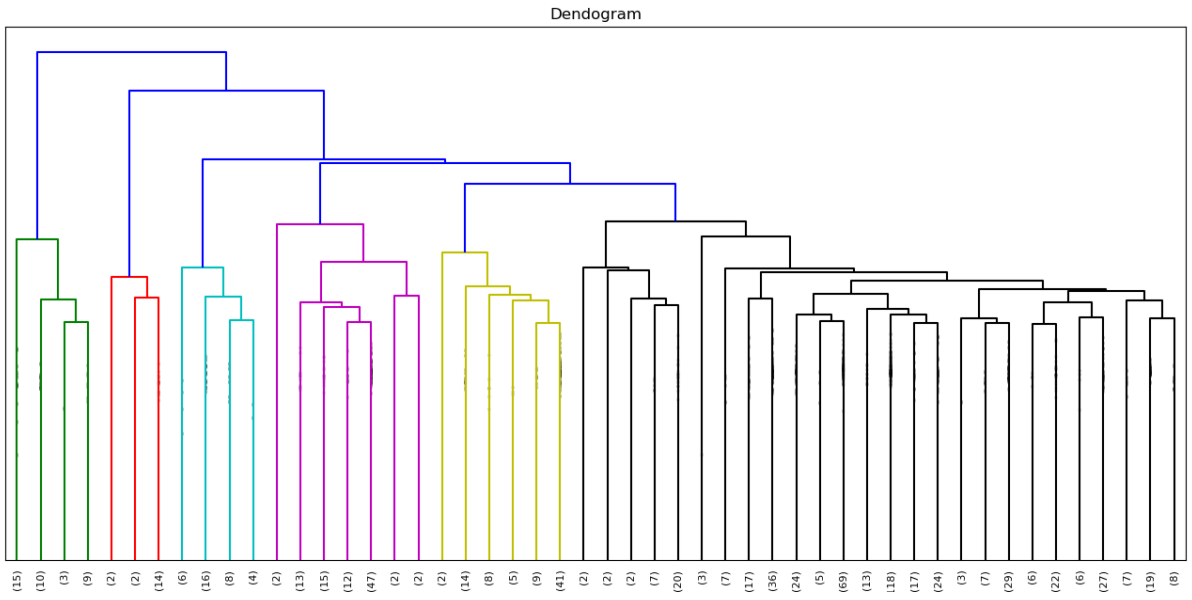
Cluster of articles regarding **Psychology and Book Reading**

From this clustering, we found that most of the articles were related to Computer Science, but also a large portion were related to Biology and Psychology.

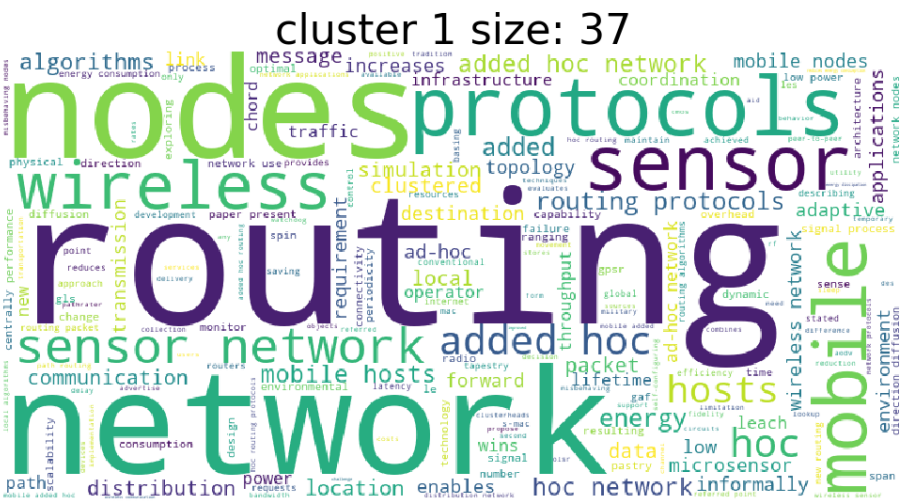
We also added a feature that finds the cluster that is most similar cluster a new academic paper. This feature can be very useful to researchers, who wish to find closely related topics and **previous substantial** work in the area. For example, a researcher that wants to enter the Computer Vision area, can use our clustering in order to find the main works that have been done in the area.

We also implemented a **Hierarichal Clustering algorithm**. We chose to do this so we can recursively explore our academic topics and subtopics. For instance, a researcher interested in “Computer Science” articles, can then find the “Computer Vision” cluster, which contains the “Feature extraction” for example, which contains the “Facial feature extraction” and so on… This way, we get another perspective at the academic world, this time looking at the structure from a “depth” point of view.

The dendogram we created for the top 1000 articles:



For example, one isolated cluster we found using this method was again the communications cluster, whose word cloud is:



4. Academic Paper Search

We wanted to implement a search, similar to “Google Search”. The search takes into account both the matchness of the query (How often does “Big Data” appear), and the importance of the paper, resulting in the most relevant and important articles. We cleaned the query using techniques from class: stemming the query using Porter-Stemmer, and removing stopwords and such. For example, when searching for “Big Data”:

Results for ' Big Data ':

1. Privacy and Security of Big Data: Current Challenges and Future Research Perspectives (Match Score: 0.60987)(194582):

Authors: Alfredo Cuzzocrea

Abstract: Privacy and security of Big Data is gaining momentum in the research community, also due to emerging technologies like Cloud Computing, analytics engi...

Link: https://semanticscholar.org/paper/b6b3bdfd3fc4036e68ecae7c9700a659255e724a

2. The National Institutes of Health's Big Data to Knowledge (BD2K) initiative: capitalizing on biomedical big data (Match Score: 0.60973)(958400):

Authors: Ronald Margolis, Leslie Derr, Michelle Dunn, Michael F. Huerta, Jennie Larkin, Jerry Sheehan, Mark Guyer, Eric D. Green

Abstract: Biomedical research has and will continue to generate large amounts of data (termed 'big data') in many formats and at all levels. Consequently, there...

Link: https://semanticscholar.org/paper/0deea670bf0da44ef4c8376b7bd4a5832a0a61e0

3. Investigation into Big Data Impact on Digital Marketing (Match Score: 0.60421)(618050):

Authors: K. Grishikashvili

Abstract: The increased accessibility of digitally sourced data and advance technology to analyse it drives many industries to digital change. Many global busin...

Link: https://semanticscholar.org/paper/ca1a13fd36904ea2d0f5a4644c599d15a49018c7

4. Techniques for Graph Analytics on Big Data (Match Score: 0.60354)(596121):

Authors: M. Usman Nisar, Arash Fard, John A. Miller

Abstract: Graphs enjoy profound importance because of their versatility and expressivity. They can be effectively used to represent social networks, web search ...

Link: https://semanticscholar.org/paper/8b35736f536f13daf049b472fdf24fcd4e0207a3

.....

5. Partner recommender system

Finally, we implemented a content-based recommender system, that given a researcher, finds the top 10 closest researchers. For every major researcher in the database, we extracted the keywords using “nltk” from every abstract of every article he/she ever wrote. We defined closest by using the bag of words model and the cosine similarity.

When Querying “Danny Keren”, a Computer Vision and Image Processing professor at Haifa University, we got the following results:

1 : C. Richard Johnson
2 : William A. Sethares
3 : Andrew G. Klein
4 : Patrice Abry
5 : Ming-Hung Lin
6 : Sz-Yu Chiou
7 : Yi-You Hou
8 : A. P. McHale
9 : Danny Crookes
10 : N. Beney

Note that this list isn’t perfect at all, however it does match some of Prof. Keren’s fields. For instance, #1 C. Richard Johnson, released an influential paper called “Image Processing for artist identification “: https://www.semanticscholar.org/paper/Image-processing-for-artist-identification-Johnson-Hendriks/160882b239795f6ed919d2a74cec58dd0aadd74e

Thoughts for the Future:

- Improving the Recommender System with more features: Recent Publications, Better choice and ranking of keywords.
- Improving the entire system with language translation – an English researcher cannot be paired for instance with a German researcher.
- Improving the running times for everything – better data structures, parallel computing.
- Ranking each cluster separately, to get a better understanding of fundamental articles in specific fields.
- Ranking between clusters and ignoring in-citations that occur inside the cluster (a kind of super-graph) in order to better understand connections between research fields.

Conclusion:

There is a whole lot to be learnt from looking at the academic world as a huge connected network. By observing the academic world in this manner, we were able to find important articles and researchers, extract useful information, get a much better understanding of different fields by clustering (topic analysis), and were even able to take a first step in recommending research partners.