Bridging the world of Citation Networks

ANKITA BIHANI- ankitab@stanford.edu -Stanford ID : 06118817, Stanford University
ANUPRIYA GAGNEJA - anupriya@stanford.edu - Stanford ID: 06117184, Stanford University
GARRETT GUTIERREZ - garrettg@stanford.edu - Stanford ID : 05844582, Stanford University

In the last few years, there has been a burgeoning interest in collaboration network analysis to help understand and visualize the dynamics of the academic social network. The growing interest in this domain can be ascribed to the significant improvements in the availability and accessibility of digital bibliographic data and relevant computer technologies.The analysis of collaboration networks plays a key role not only in describing the evolution of researchers and their research communities, but also in the generation of knowledge. In this project we aim to explore and analyze the collaboration network of our dataset and create meaningful results.

Categories and Subject Descriptors: [**1**]: *—Link prediction in Citation networks, Recommendation in Citation networks, Analysis of citation networks*
General Terms: collaboration networks, natural language processing, citation networks, clustering, recommendations, link prediction, collaboration distance, influence graph

1. INTRODUCTION

Researchers usually collaborate with other researchers or cite other authors while publishing their own research work. Collaboration network, therefore, can be represented by undirected graphs and if we observe closely, we can find some remarkable patterns that can be mined to get meaningful interpretations and results from these relationships. We solve the problem of link prediction in collaboration networks and predict collaborations which can be useful in guiding future collaborations of a researcher. Using Natural Language Processing techniques, we arrive at research interests profiles of each of the authors, which is used to cluster them into separate communities.We also built a recommendation system that can help guide research work by suggesting the authors to read relevant research works that they might be interested in. We also compute the collaboration distance between a pair of authors in the dataset to present to the author, the number of hops required to collaborate with a particular researcher (potentially a domain expert). Besides, we also present the influence graph for every researcher. Most of the above functionalities are available on the web-portal we developed for our project.

2. MOTIVATION

The primary motivation behind this project is to bridge the world of citation networks, thereby helping research scholars create more impact and build better and more impactful collaborations.

3. LITERATURE REVIEW

3.1 Natural Language Processing

In the paper **Automatic Extraction of Keywords from Abstracts** [7], the author has proposed a model to automatically extract keywords from abstracts. He does this by building a list of words and collocations sorted in descending order, according to their frequency, while filtering general terms. Although this seems to be a good strategy to extract the keywords from the abstract in our project, the failure rate for this model was approximately 38.25% which has been improved in our implementation.

3.2 Link prediction

Liben-Nowell and Kleinberg [3] investigated and analyzed a host of link prediction techniques and classified them into three categories:
1.Node Neighborhood Based Methods that included common neighbors, preferential attachment, Jaccard coefficient, Adamic-Adar.
2. Paths Based Methodologies like PageRank, SimRank
3. Higher Level Approaches like low-rank approximation, unseen bigrams and clustering.

In [1],Allali et al proposed a new approach of internal link prediction for bipartite graphs. They defined concepts of induced link and internal links in bipartite graphs. Using induced links, they create a projection graph, assign neighborhood based weights to the edges in the projected graph, then identify internal links which are likely to appear in the future. Once the internal links are generated, their future appearance is predicted by comparing the assigned weight against a pre-defined threshold.

Chaoji, Salem, and Zaki (2006)[4] tested several supervised learning models (decision tree, k-nearest neighbor, multilayer perception, support vector machine [SVM], radial basis function [RBF] network) for link predictions and their analysis showed that using SVM outperformed all of the other techniques by a narrow margin in terms of all performance measures.

Hasan et al [5] summarized major categories of link prediction methods: network feature based classification model, probabilistic model and linear algebraic model. They investigated the characteristics of each type of link prediction methods and discussed the challenges frequently associated with link prediction and the methods to deal with these challenges.
The algorithm that we use for link prediction is also motivated from the above research papers.

3.3 Clustering

In [6] Kannan et al. establish two primary goals. The first is to quantify what constitutes a ”good” clustering in a given graph. The second is to provide a reasonably efficient algorithm for clustering a graph well with respect to this definition. Eventually, the quality of clustering is defined in terms of the graph’s conductance and the ratio of the weight of inner cluster edges over the total weight of all edges. The goal of finding a good clustering, then, is defined as the goal of finding some clustering that maximizes the former while minimizing the latter. A reasonably efficient algorithm is provided that attempts to find a good clustering given this definition. Since finding such a clustering for any graph would be NP hard, an algorithm that converges upon an approximation is used. In this paper, the authors have proposed a reasonably efficient algorithm for arriving at ”good clustering”. However, they have not shown experimental evidence by evaluating the performance of this algorithm on real-world data.

4. DATASET

For this project, we used the ACM-Citation-network V8 which has 2,381,688 papers and 10,476,564 citation relationships. Each entry of the dataset contains a piece of publication record, which includes paper Title, name of the authors, year of publication, publication venue/ journal, index ID of the paper, the id of references of this paper and the abstract. Table 1 shows the graph statistics of the author-author collaboration graph that we create from our dataset.

4.1 Dataset cleaning and creating relevant graphs

We extract the entire data from citation-acm-v8 datatset and parse the data to arrive at relevant mappings.

We store the mapping between every author of a paper and the paper ID associated with that author. There is a separate row for each of the authors of a particular paper. This table is then loaded as a graph to give us the paper-author collaboration network where an edge from A to B denotes that paper A has one of the authors as B.

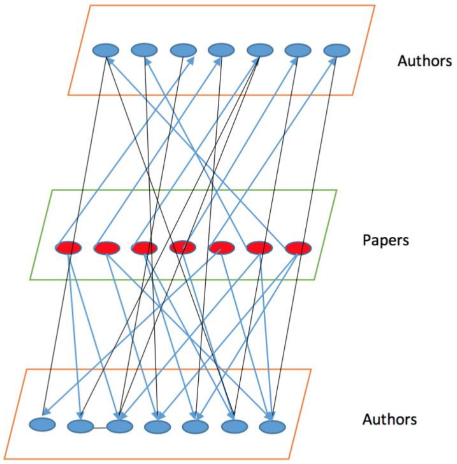


Fig. 1. Author-Author Collaboration Netowork.

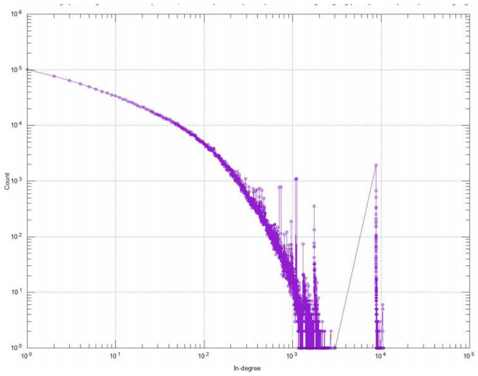


Fig. 2. Degree Distribution of the Author Author Collaboration Network

Table 1: Dataset Statistics

Author - Author collaboration network	
Number of nodes in the Graph	1517920
Number of edges in the Graph	4938957
Number of Non Zero Deg Nodes	1517920
Number of unique directed edges	9877914
Number of unique Bidirectional edges	4938957
Is the graph is weakly connected	False
Fraction of nodes in the largest strongly connected component	0.828966612206
Approx full diameter of the graph	23
Approx effective diameter of the graph	7.27126828482
Clustering coefficient of the graph is	0.613873473382
Number of Triads in the graph	8806483

Additionally, we store a mapping between every paper ID and its references. There is a separate row for each of the references of a particular paper. This table is then loaded as a graph to give us the citation network where an edge from X to Y represents that paper X cited paper Y.

Once we have the paper-author network we can get an author-author collaboration network, where an edge exists between two authors if they have a common paper.

Fig. 1 shows the author - author collaboration network representation. To analyze the author-author collaboration network in depth, we plot a few graphs. Fig. 2 shows the degree distribution of this author-author collaboration network in From the degree distribution plot, we see that maximum number of nodes in the network have degree 1, which implies that maximum number of authors have only one collaboration. Fig. 3 shows the plot of the K-core edge size distribution. Fig. 5. shows the distribution of sizes of strongly connected components of the graph. Fig. 6 shows the cumulative distribution of the shortest path lengths of the Author-Author collaboration Graph. Fig.7 shows the visualization of a subset of the authors in the author-author collaboration network.Fig.8 shows the distribution of clustering coefficient of the same.

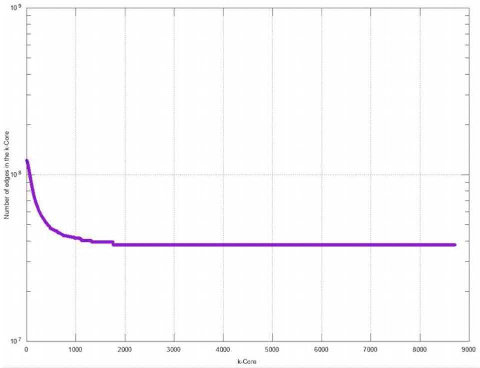
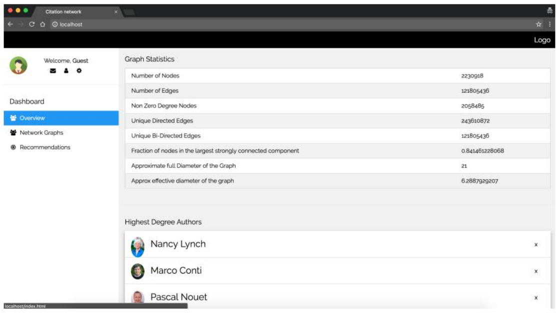


Fig. 3. Plot of k-core edge-size distribution i.e., core k vs. number of edges in k-core of the Author-Author Collaboration Graph.



Graph Statistics

Number of Nodes	2230918
Number of Edges	121805436
Non Zero Degree Nodes	2058485
Unique Directed Edges	243610872
Unique Bi-Directed Edges	121805436
Fraction of nodes in the largest strongly connected component	0.841461228068
Approximate full Diameter of the Graph	21
Approx effective diameter of the graph	6.2887929207

Highest Degree Authors

Nancy Lynch ×

Marco Conti ×

Pascal Nouet ×

Fig. 4. Snapshot of the home page of our web portal

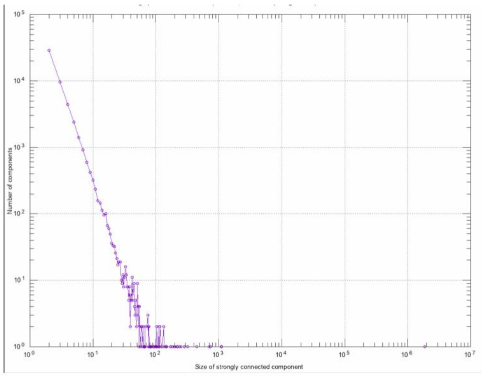


Fig. 5. Plot of Distribution of sizes of strongly connected components of the Author-Author Collaboration Graph.

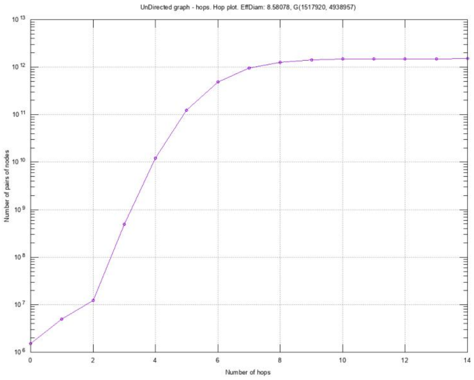


Fig. 6. The cumulative distribution of the shortest path lengths of the Author-Author collaboration Graph.

5. LINK PREDICTION FOR COLLABORATIONS

In our project, we predict future collaborations, i.e, predicting the potential future researchers, a researcher might collaborate with. We use the algorithm 1 below for solving the link prediction problem.

ALGORITHM 1: Algorithm for link prediction

```
We divide our dataset based on the year of publication.Our predictor uses links between authors in the time interval [t1, t2] and predicts the future collaborations in the time interval [t2,t3].
1. For each pair of authors (x,y):
(a) if (x,y) $\in$ Edges(G) for [t1,t2]: continue
(b) Otherwise, compute the similarity score between authors (x,y) based on the features listed in the section below for time interval [t2,t3]
(c) If the score for pair of authors (x,y) $<$ pre-defined threshold, assign score = 0 for (x,y)
(d) Otherwise, assign the score for pair of authors(x,y) = 1
2. Return all the pairs of authors with score = 1 as the future collaborations in the interval [t2,t3]
```

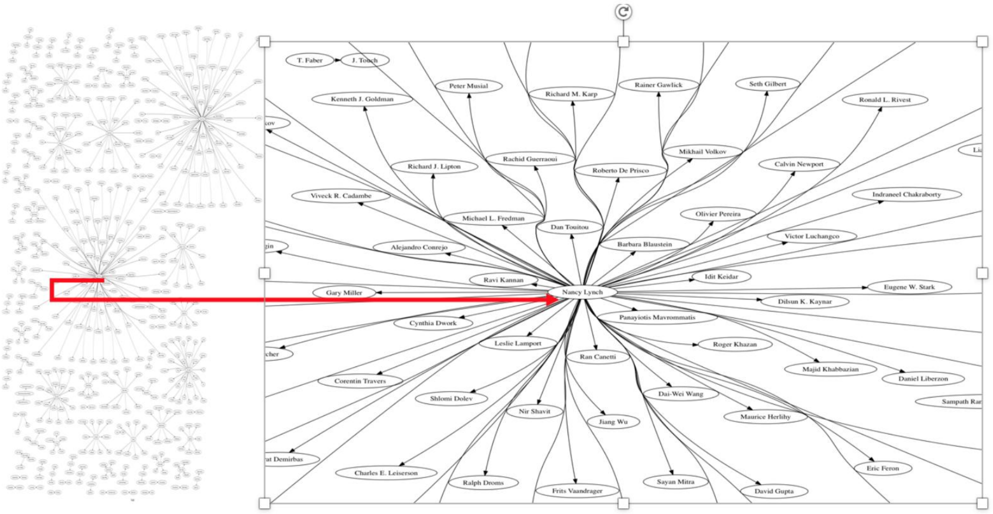


Fig. 7. Snapshot of the visualization of 2000 of the nodes from the Author-Author Collaboration network created from the acm-citation network. Alongside, this graph, we have the zoomed-in version of one of the highest degree node in this graph and the immediate neighbours of the node

The results of this problem can provide interesting results with meaningful insight. We can use this model to provide collaboration recommendations to researchers based on their similarity scores.

We used top 10 features based on their ranks in our model. ranked by the combined score given by Information Gain and Gain Ratio statistics to obtain a ranking for each of our features.

5.1 Feature Ranking Methods

5.1.1 *Information Gain.* Information Gain measures the information gained by choosing an attribute with respect to the class.It can also be expressed as mutual information. The mutual information $I(X_i; C)$ of two random variables X_i and C is defined as

$$I(X_i; C) = H(X_i)H(X_i|C),$$

Here, $H(X_i)$ is the entropy of X_i, and $H(X_i|C)$ is the conditional entropy of X_i conditioned on C.
Now the information gain $IG(X_i)$ of a feature X_i with respect to class label C is defined as

$$IG(X_i) = \sum I(X_i; C)$$

5.1.2 *Gain Ratio.* The gain ratio of a feature X_i with respect to the true label C is their mutual information normalized by the entropy of X_i

$$GR(X_i; C) = \frac{I(X_i;C)}{H(X_i)}$$

5.2 Features used in Link Prediction

5.2.1 *Number of common neighbours.* The score of the number of common neighbours between two authors (x, y) in the graph is defined as:

$$score(commonneighbours) = \frac{Neighbours(x) \cap Neighbours(y)}{Neighbours(x) \cup Neighbours(y)}$$

5.2.2 *Jaccard Similarity Coefficient of Paper Titles.* For each pair of authors, we compute the similarity in their paper titles based on the number of common words that their papers' titles share. For each pair of authors (x, y), we compute two vectors, $T_x = w_{x1}, w_{x2}, ...w_{xk}$, $T_y = w_{y1}, w_{y2}, ...w_{yl}$ where T_x contains the relative frequency of term i in the bag of words for titles of author x and T_y contains the relative frequency of term i in the bag of words for titles of author y. We now compute the title similarity by measuring the Jaccard similarity between vectors T_x and T_y as follows:

$$JaccardTitleSimilarity(x, y) = \frac{T_x.T_y}{(|T_x|+|T_y|-|T_x.T_y|)}$$

where $|T_x|$ is $\sqrt{w_{x}{}_{1}^{2} + w_{x}{}_{2}^{2} + ...w_{x}{}_{k}^{2}}$

5.2.3 *Cosine Similarity Coefficient of Paper Titles.* For each pair of authors, we compute the similarity in their paper titles based on the number of common words that their papers' titles share. For each pair of authors (x, y), we compute two vectors, $T_x = w_{x1}, w_{x2}, ...w_{xk}$, $T_y = w_{y1}, w_{y2}, ...w_{yl}$ where T_x contains the relative frequency of term i in the bag of words for titles of author x and T_y contains the relative frequency of term i in the bag of words for titles of author y. We now compute the title similarity in titles of authors x and y by measuring the cosine similarity between vectors T_x and T_y as follows:

$$CosineTitleSimilarity(x, y) = \frac{T_x.T_y}{(|T_x| \cdot |T_y|)}$$

where $|T_x|$ is $\sqrt{w_{x}{}_{1}^{2} + w_{x}{}_{2}^{2} + ...w_{x}{}_{k}^{2}}$

5.2.4 *Jaccard Similarity Coefficient of Paper Abstracts.* For each pair of authors (x,y), we compute the similarity in their paper abstracts based on the number of common words that their papers' abstracts share. After pre-processing,for each pair of authors (x, y), we compute two vectors, $T_x = w_{x1}, w_{x2}, ...w_{xk}$, $T_y = w_{y1}, w_{y2}, ...w_{yl}$ where T_x contains the relative frequency of term i in the bag of words for abstracts of author x and T_y contains the relative frequency of term i in the bag of words for abstracts of author y. We now compute the abstract similarity between authors x and y by measuring the Jaccard similarity between vectors T_x and T_y using the formula in section 5.2.2:

5.2.5 *Cosine Similarity Coefficient of Paper Abstracts.* For each pair of authors,(x,y), we compute the similarity in their paper abstracts based on the number of common words that their papers' abstracts share. For each pair of authors (x, y), we compute two vectors, $T_x = w_{x1}, w_{x2}, ...w_{xk}$, $T_y = w_{y1}, w_{y2}, ...w_{yl}$ where T_x contains the relative frequency of term i in the bag of words for abstracts of author x and T_y contains the relative frequency of term i in the bag of words for abstracts of author y. We now compute the abstracts similarity between authors x and y by measuring the cosine similarity between vectors T_x and T_y using the formula in section 5.2.3

5.2.6 *Belong to the same Cluster?.* Next we compute a score which tells us if two authors x and y belong to the same cluster. We use our clustering algorithm mentioned in section 9 to assign the authors into different clusters and then calculate a score as follows:

$$ClusterScore(x, y) = 1[xandyinthesamecluster]$$

5.2.7 *Number of hops between them in the shortest path.* We calculate the number of hops between two authors x and y and then output a score as follows:

$$NumberOfHops(x, y) = 0.1 * NumberOfHopsBetweenXAndY$$

5.2.8 *Difference in Degree Centrality.* The difference in Degree Centrality is defined as :

Difference in Degree Centrality = $abs(DegreeCentrality(x) - DegreeCentrality(y))$

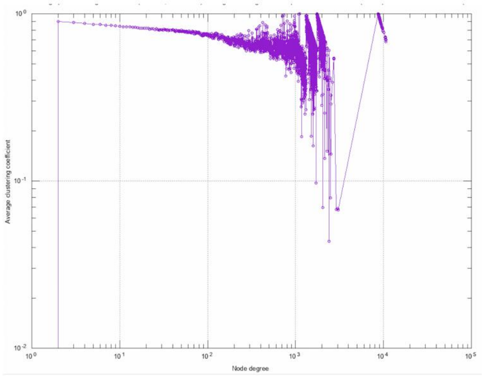


Fig. 8. The distribution of clustering coefficient of the Author-Author Graph.

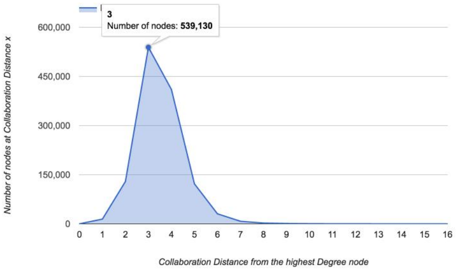


Fig. 9. The number of nodes at a distance 'x' from the node with the highest degree centrality.

5.2.9 *Have papers in common journal?*. The score is defined as :

$$CommonJournals = 1[IfXAndYHaveAtleastonePaperInaCommonJournal]$$

5.2.10 *Number of common references in their papers* . The score for the number of common references in papers is defined as:

$$NumberOfCommonReferences = \frac{count(References(x) \cap References(y))}{count(References(x) \cup References(y))}$$

6. SKILL-SET / RESEARCH INTERESTS PROFILING

One of the key challenges that we address in our system is to predict the skill-set/ research interests of a researcher by using Natural Language Processing techniques on the abstracts and titles of their published research papers.We created a map from authors to all their paper titles and abstracts. Then, for each of the authors we extract the keywords using tf-idf and rake python package. For any keywords to be considered as a potential research interest/ skill set of a researcher it has to have a weight ≥ 5. We show the results of our algorithm for 2 of the authors in the Results section.

7. PEOPLE WHO CITED X ALSO CITED Y

Recommendation engines have indeed revolutionized e-commerce and social networking, hence we wanted to build a recommendation system for our citation network, as well. Recommendations of similar authors and/or similar research papers using common co-authors could be useful in guiding and influencing research works by recommending relevant research works. For all the papers in our dataset, we extract the papers that appear in the References section and return the top 5 papers that were cited maximum number of times with this paper, in decreasing order of number of co-appearances.This can be useful to provide a researcher with relevant research works that can motivate the current research work. In Fig. 12, we demonstrate a snapshot of how our web-interface returns the most co-cited papers for any given paper.

8. COLLABORATION DISTANCE

We compute the shortest collaboration distance between all pairs of researchers that lie within 3 hops of each other in our dataset.This would be useful to see how many collaborations (hops) an individual needs to reach another researcher (potentially a domain expert). We also computed the most central node (based on degree centrality) in the Author-Author collaboration graph and calculated the shortest path length from this node to all the other reachable nodes. The plot in Figure 9 shows the number of nodes at a distance 'x' from the node

Table 3:Link prediction results using SVM Classifier

TP Rate	FP Rate	Precision	Class
0.71	0.09	0.8875	1
0.79	0.18	0.8144	0
0.75	0.15	0.8437	Average

with the highest degree centrality. We can see that maximum number of nodes in the graph have a collaboration distance of 4 from the highest degree node which implies that most of the authors can collaborate with the author having the highest degree centrality (who is potentially the most influential researcher) in 4 hops. We also observe that there are barely 4 nodes that have a collaboration distance of 16 (which is the maximum in this graph).In Fig. 13, we demonstrate a snapshot of how our web-interface returns the collaboration distance between any two authors given to it as input.

9. CLUSTER ANALYSIS IN CITATION NETWORKS

We cluster all the authors in the dataset based on their paper title and abstract similarity measures.Once we segregate the authors into different clusters, we use this information as a feature in order to solve our link prediction problem. We used multiple clustering techniques including the similarity measures and the k-means clustering algorithm.

10. INFLUENCE GRAPH

Influence graph is defined as a representation of the top ’n’ researchers who have influenced the work of a particular researcher. We created such an influence graph for every researcher in our dataset. The purpose of this idea is to help researchers understand the career trajectory of their collaborators in a better way. We created a mapping between authors in our dataset to the authors of papers listed in the references section of their papers. Using this mapping we compute the top ’n’ authors that have influenced the work of a given author.

11. RESULTS

11.1 Link Prediction

The table 3 above shows the results of the Link Prediction problem. We enumerate the True Positive Rates, False Positives Rates and Precision for the two classes where 1 represents that there is a high possibility of collaboration link between the two authors and 0 represents that there is a very low probability of the two authors collaborating in the future.We tried the Naive Bayes algorithm for our problem but we found that SVM outperformed Naive Bayes significantly.

11.2 Skill-set Profiling

Fig. 10 and 11 show the skill-set and research interests of Professor Jure Leskovec and Professor Daphne Koller respectively. As can be seen from Figure 10, the highest score for Professor Jure’s research interest is given to ’Social Networks’ by our algorithm which is fairly accurate in this case. For the evaluation of our algorithm we manually compared the keywords obtained as the result of our algorithm to skill sets and research interests of the respective researchers from their aminer.org and LinkedIn profiles (for 100 authors sampled randomly). The average accuracy of our prediction for this sample was close to 85%.

11.3 People who cited X also cited Y

Fig. 12 shows the results obtain from our web interface for the papers that co-appear with a given paper in the decreasing order of number of co-appearances. We compute and store the most co-cited papers in a database,

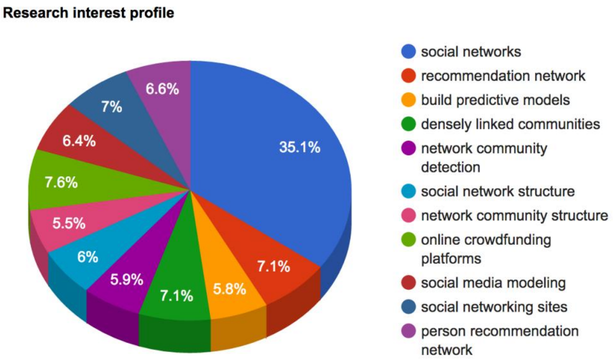


Fig. 10. Research interests of Professor Jure Leskovec

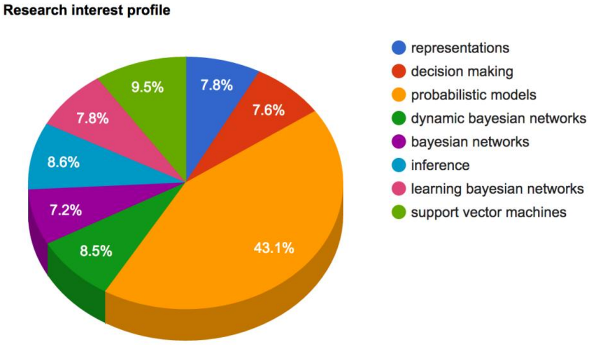
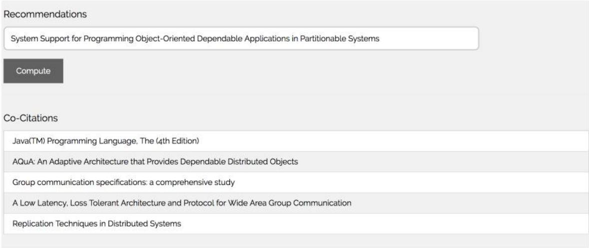


Fig. 11. Research interests for Professor Daphne Koller



Recommendations

System Support for Programming Object-Oriented Dependable Applications in Partitionable Systems

Co-Citations

Java(TM) Programming Language, The (4th Edition)

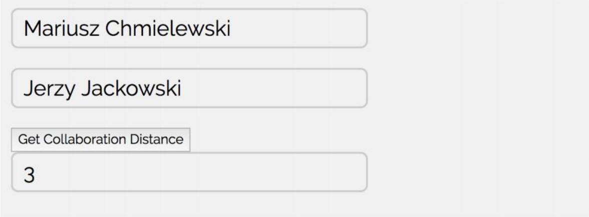
AQuA: An Adaptive Architecture that Provides Dependable Distributed Objects

Group communication specifications: a comprehensive study

A Low Latency, Loss Tolerant Architecture and Protocol for Wide Area Group Communication

Replication Techniques in Distributed Systems

Fig. 12. Snapshot from our webportal returning recommendation Results for a given paper



Mariusz Chmielewski

Jerzy Jackowski

3

Fig. 13. Snapshot from our webportal computing the shortest collaboration distance between two given authors

which can be retrieved to display the results on our portal.These results can be used to provide citation/reading recommendations to people based on the papers that they have already cited.

11.4 Collaboration Distance

Our dataset for author-author collaborations contained close to 1.5 million nodes. The challenge involved in calculating the collaboration distance between each pair of authors was the sheer volume of computations and space required to store the same. For every single author, we compute the collaboration distance with every other author in the dataset. When we first attempted to solve this problem, for ≈ 1 % of the authors in the dataset, we ended up having 121 million rows in a single database. This was computationally very inefficient. Hence, to efficiently store and retrieve the collaboration distance between two authors for our web portal, we logically sharded the collaboration distance data into 1000 shards and used multi-threading to parallely compute the collaboration distance to be stored in the database. Using parallel computations, we were able to reduce the time required for computing and storing the collaboration distance for every 500 authors from ≈ 8 minutes down to 50 seconds.

11.5 Clustering

Fig. 14 shows the Clustering results for a subset of the authors from our author-author collaboration network. We assign the authors into 5 clusters using the K-means algorithm and Similarity between their papers' titles and abstracts. On comparing the results of our clustering with the research interest profiles generated for each

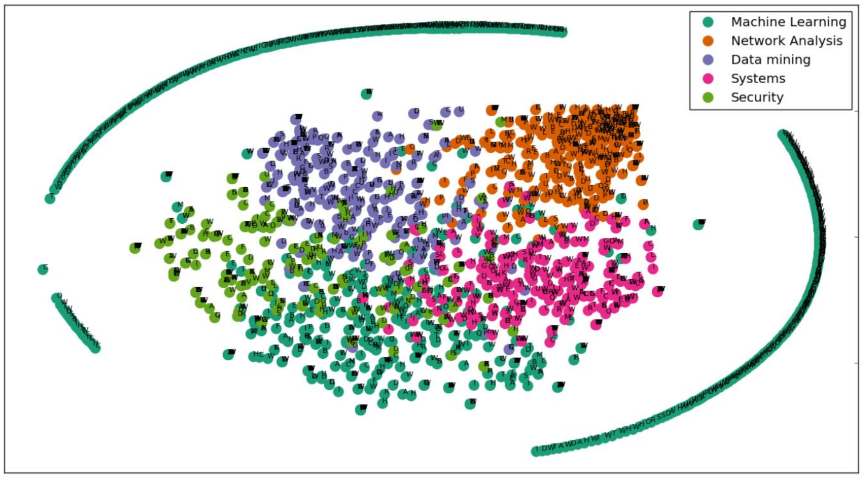


Fig. 14. The figure shows the results of K-means and similarity based clustering algorithms applied to a sample dataset from the author-author collaboration network. The authors have been classified into 5 clusters predefined by us.

of the authors, we find that close to 70% of the people with similar research interests were grouped together by our clustering algorithm.

12. CONTRIBUTIONS

Ankita Bihani & Anupriya Gagneja - Equally contributed to the implementation of the entire project.
Garrett Gutierrez - Literature review of the Clustering segment (section 3.3 of the report)

13. REFERENCES

[1] Oussama Allali, Clemence Magnien, and Matthieu Latapy. Link prediction in bipartite graphs using internal links and weighted projection. 2011
[2] Mohammad Al Hasan and Mohammed J. Zaki. A Survey of Link Prediction in Social Networks. 2011.
[3] David Liben-Nowell and Jon Kleinberg. The Link Prediction Problem for Social Networks. 2003.
[4] Hasan, M.A., Chaoji, V., Salem, S., Zaki, M. (2006). Link prediction using supervised learning. SIAM 2006 Workshop on Link Analysis, Counterterrorism and Security, Bethesda, MD.
[5] Mohammad Al Hasan and Mohammed J. Zaki. A Survey of Link Prediction in Social Networks. 2011.
[6] Kannan, R., S. Vempala, and A. Veta. (2000) ”On Clusterings-good, Bad and Spectral”, Journal of the ACM, Vol. 51, No. 3, May 2004, pp. 497-515.
[7] HaCohen-Kerner, Y. (2003). ”Automatic extraction of keywords from abstracts” In V. Palade, R. J. Howlett, & L. Jain (Eds.), Knowledge-based intelligent information and engineering systems: Proceedings of the 7th international conference, KES 2003, Oxford, UK, September 2003, Part I (pp. 843-849).