Visualizing Ranked Theta-Join Results

Zixuan Chen* Aamod Khatiwada

Figure 1: Our proposed visualizations visualize ranked theta-join results. Both single ranking view and multi-ranking view are provided. In the single ranking view, a scatter plot is shown alongside the ranking table, which presents the weight distribution of the results in the ranking. In the multi-ranking view, at most three rankings are juxtaposed to be compared. The parameters of a ranking function can be tuned via the slide bars above each ranking.

Abstract

Ranked enumeration over joins began to attract attention recently. A ranked enumeration algorithm uses a ranking function to rank join results. For change in ranking function, the ranking result also gets changed. Knowing the influence of a certain ranking function and why some results are ranked at the top is helpful for understanding the joins and rankings. We propose a visualization tool to assist data management researchers in having a deeper understanding of these rankings. Our visualization enables researchers to easily find out why certain results are ranked top and compare multiple rankings to get insights. Our github repository is `https://github.com/NEU-CS-7250-S21/project-group-02-ranked-theta-join`.

*E-mails: [chen.zixu | khatiwada.a]@northeastern.edu

Index Terms: Human-centered computing—Visualization

1 Introduction

Join is an essential operator of queries in databases. A theta-join joins two tables based on a relationship between two columns instead of direct equality. Because of the various setting of θ operator, theta-join is powerful and can be leveraged to do tasks like similarity join. Ranking the results obtained after the theta-join using a certain ranking function has recently become a very interesting topic and open problem [1]. This is known as Ranked enumeration of Theta-join and the ranking order is determined by a given ranking function.

In this paper, we focus on ranked enumeration of conjunctive queries containing theta-joins. Since conjunctive queries tend to be complicated, researchers are not able to easily identify why certain results are at the top of a ranking list, which hinders further research on the ranking results. Moreover, researchers usually tune the parameters of a ranking function to get different sets of results. However, without an adequate visualization tool, it is quite difficult

for them to compare those ranking result sets, which makes it hard for them to notice the influence of a ranking function as well as its parameters on the ranking results.

Therefore, we argue that it would be helpful to use the visualizations for explaining the rankings to researchers. For 4 months, we worked with domain experts to assess what visualization tools they need and how visualizations can help them understand the ranking results of joins better.

As the output, in this paper, we propose a visualization tool to help researchers understand the ranking results better and get insights more easily. Two visualizations are provided. The first one is a single ranking view which provides insights about a single ranking and aims to help users understand why certain results are at the top of the ranking. The second visualization juxtaposes multiple rankings to enable researchers to easily observe what will happen if the parameters of the ranking function is changed. Our demo is online at `https://github.com/NEU-CS-7250-S21/project-group-02-ranked-theta-join` with a demo video attached. Our slides are online at `https://docs.google.com/presentation/d/1gtVEIwBwWzoJ6WOdH_67MMYI6nd76qPTZ3BZ3XAglKM/edit#slide=id.p`.

2 Related Work

Three papers of our partner Professor Mirek Riedewald [9, 10, 13] provide detailed introduction about ranked enumeration of answers to conjunctive queries, and he thinks adding visualization to explain these results would be helpful. In his work, the top results are printed one by one, and solid algorithms are provided to conduct the rankings. Some nice and clear visualizations will for sure make the ranked enumeration more understandable.

There are also several papers [5, 6] which work on Visualization over queries including SQL queries and graph queries in recent years. They are showing a trend to utilize visualizations and help the database researchers.

For visual analysis of rankings, LineUp [2] is a great paper which provides visualizations which share the similar ideas with ours. It aims to rank data by custom attributes and weights. It also provides ranking comparisons. The main difference of our proposed visualization is that we provide more fine-grained analysis on every ranking result (row) and even every edge (cell) of the rankings. Moreover, our initial goal is to provide various ranking functions which supports more custom rankings.

RankBooster [7] provides several charts to visualize and predict rankings. RankExplorer [8] focuses on time series data. Podium [11] also provides an attribute-based ranking and is able to compare the current results with the previous one. SRVis [12] is similar to LineUp in terms of its ranking view but it integrates the spatial visualizations as well.

We had a rather clear aim at the beginning of the course so there is no big change compared to our plan. We planned to implement the single ranking and multi-ranking functions at the very beginning. The scatter plot was also in our sketches although it was of lower priority and we didn't decide whether to add it at the beginning.

3 Partner

Professor Mirek Riedewald from the Data Lab, Northeastern University, is our Service Learning Partner for this project. He is the author of two papers related to ranked enumeration over joins and thinks it will be helpful to have some visualizations on it. Moreover, he provided us an appointment for the interview and participated in our discussions. For the interview, we prepared seven high-level questions to make sure that our discussion does not deviate from our major goals.

Before the interview, we only considered the database community to be our prospective partners. But after interviewing Professor Riedewald, we were surprise to know the impact, width and the horizon of this work. This visualization will not only impact the database research community, but also will assist the analysts, researchers, and naive users from other areas like network security, social network analysis, animal movement analysis, and vehicle navigation monitoring. Furthermore, we came to know that there are no any existing visualizations to assist him and most of the time, his research team is dependent on the statistical analysis for explaining their results.

From this interview, we understood our Service Learning Partner has four basic requirements, which we recorded in user requirement section of our note. Apart from them, he is open to consider our suggestions on more visual representations that help explaining the results of his algorithm. Also, it will be interesting for him, if we substitute the primarily chosen bitcoin data with the similar graph data from other fields like network attack, bird observation, and vehicle navigation and also explain the performance of his algorithm on those domains.

This interview made us more clear about the end goal of our service learning partner and also realized the prospective scalability of our project.

4 Data

Our general task is to analyze how to use visualizations to help explain the rankings of theta-joins. More specifically, we convert the theta-join background into a graph problem. We are using different sets of nodes to represent different tables, edges to represent the joinability between rows of the tables and the edge weight between nodes to represent the theta-join strength between those rows.

For our visualizations, we use a real network, the Bitcoin OTC trust network provided by [3, 4]. In this network, users are linked by edges with weights representing the degree of trust from one user to another. The degree of trust can be either positive or negative. To control the input size, we only use a part of the data by extracting only relationships between users with ids smaller than a threshold. We use MapReduce to conduct self-joins of the data to get the 3-edge result set as the dataset to be ranked, which contains 4 nodes and 3 edges between them. Each set of nodes corresponds to a table. Here, we have 4 set of nodes and they represent four tables. Note that, our work can be scaled for the larger number of tables. The nodes in each set are the tuples in a table. For example, the edge weights between first two columns represent the join strength of the rows from table1 with that from table2.

5 Task Analysis

Table 1 is the Task table of our work. Our partner raised two questions to us, which are the major tasks we are trying to do, "Can you use visualizations to help me understand why certain results are at the top of a ranking?" and "Can you use visualizations to show the influence of different ranking functions on the ranked results?".

For the first task, the high-level task is to discover the reasons of why these results are ranked top and present nice visualizations to help researchers understand that. We need to allow users to look up the appearances of each edge in these results. With more charts along side the ranking, we can finish the low-level task of identifying one result and summarize the distribution.

For the second task, the high-level task is to discover the influence of different ranking functions and present the comparisons between them. The users should be able to look up the appearances of certain results in different rankings and compare them.

The consumers of our visualizations can be various, including Database researchers and anyone who wants to know the reasons of some specific complicated rankings. For instance, these visualizations may be used in social network analysis, route planning, etc.

Task ID #	Domain Task	Analytic Task (low-level, "query")	Search Task (mid-level)	Analyze Task (high-level)
1	Show why some certain results are at the top	Identify and Summarize	Lookup	Discover and Present
2	Showing the influence of different ranking functions	Compare	Lookup	Discover and Present

Table 1: Domain task and abstract tasks.

6 VISUALIZATION DESIGN

In this section, the detailed visualization design is presented. We designed an interactive visualization tool to help database researchers explore the enumeration ranking results of theta-joins better. Both visualizations on single view and multiple views are provided. Our designs are based on the task abstractions in the previous section.

Fig. 1 shows all our visualization components including a single ranking view visualization and a multi-ranking view visualization. In the following subsections, we introduce the ranking table, the scatter plot and the multi-ranking view respectively. The static encoding, the interactions and the linking views are included in these sections. Our design juxtaposes multiple views to show the details and comparisons. The used channels include color, luminance, line width and point position to provide visualizations for various attributes.

6.1 Ranking Table

This ranking table provides insights about a single ranking. It aims to help users understand why certain results are at the top of the ranking.

The single ranking is used to show the ranking results of a theta-join. It is in the form of a table but it is more powerful and illustrative. The ranking shows top-K results in terms of the ranking function f. The odd columns show the node id (corresponding to table in the theta-join background) and the even columns show the edge connecting two nodes where its width is used to represent the edge weight (corresponding to link strength between tables in the theta-join background), and the luminance of its color is used to show the importance of this column (e.g., α, β, γ parameters in the linear ranking function).

In the ranking table, we want to encode the importance of the column and the edge weight at the same time so we use the luminance of the color and the width at the same time. Combining the two channels, we can get the importance of one edge. And sum the importance values of all edges in one row to get the total weight, which is the ranking score. Therefore, from the ranking table, we can easily figure out why certain results are at the top. The top results tend to have wider edge for darker color, which means large weight for important edges.

The ranking table has the following interactions:

- Change the number of K to set the row count in the ranking (this K corresponds to the top-K results we want to show in the ranking).
- Change the order to choose whether the lightest path(ascending) or the heaviest path(descending). In ascending order, the color of edges is red while in descending order, the color of the edges is blue.
- Use the slide bars to set the parameters of the ranking function. When a slide bar is changed, the ranking as well as the ranking function changes correspondingly.
- Hover over one row to show the detailed weight calculation (the number above the edge is the weight * parameter) and click on one row to highlight. We define the number above the edge to be the *edge importance*. We name the number shown in the last column to be the *total weight* of this row. The total weight equals the sum of all the edge importance values in one row.
- Double click on one edge to show all appearances of it in the ranking. This can help researchers find some critical or frequently appeared edges/paths.
- Brush some consecutive rows to highlight them all.

The single ranking table is linked to the scatter plot on the right side. The table and the scatter plot share the same highlighted elements. If the slide bars of the ranking table is tuned, corresponding changes show in the scatter plot.

6.2 Scatter Plot

Alongside the ranking table, a scatter plot is shown. This scatter plot shows the total weight distribution against the weight of edges in each column and aims to help researchers better understand the single ranking.

In the scatter plot, every node refers to one row in the ranking table. The y axis shows the total weight of this row, and the x axis shows the weight of a certain edge in this row. We can use the selection box to change the selected edge. The y axis is set to be from the minimum total weight to the maximum weight of the ranking and change according to the selection of parameters in the ranking function.

We use the scatter plot because it is a simple way to show the distribution clearly.

The scatter plot has the following interactions:

- Change the number of K to set the node count in the scatter plot.
- Select an edge in the selection box to choose what exactly the x axis means. If Edge1 is selected, then the scatter plot shows the total weight distribution against the weight of edges in the column between node1 and node2. Interestingly, we found out that if one column is of low importance, the nodes distribute evenly comparatively, and if one column is of high importance, the nodes tend to gather together to one side of the scatter plot.
- Brush a rectangle to highlight all the nodes inside it.

As introduced in the previous subsection, the scatter plot is linked to the single ranking table so all the highlighted elements are linked.

6.3 Multi-ranking View

The multi-ranking visualization provides insights about multiple rankings. We juxtapose at most three ranking tables. It is easy the observe what will happen if the parameters of the ranking function is changed. In the meanwhile, it is a good tool to compare multiple rankings at the same time. The static designs of one ranking remain the same as the ranking table we introduced before.

Ranking tables are juxtaposed so that it is very intuitive to compare the positions in different rankings where a certain result appears. In this way, researchers can easily understand the influence of a certain ranking function as well as its parameters.

The multiple rankings have the following interactions:

- Create new rankings (at most 3).

- Delete a ranking by using the deletion button at the top of the ranking.
- Change the number of K to set the row count in the ranking. This interaction applies to all the rankings at the same time.
- Use the slide bars to set the parameters of the ranking function. This interaction only applied to one certain ranking since every ranking has the slide bar box. By setting different sets of parameters for different rankings, the aim of showing the influence of ranking functions can be achieved.
- Hover over one row to show the detailed weight calculation. Click on one row to highlight, not only the appearance in this ranking but also the appearances in other rankings. All rankings are linked.
- Brush some consecutive rows in one ranking to highlight them all. At the same time, if these rows appear in other rankings, the appearances in other rankings are linked to be highlighted as well.

7 Discussion

In this section, we discuss about the takeaways, limitations and the future work of our visualizations.

7.1 Takeaways

First of all, after our efforts, we successfully implemented all the visualizations with their encodings and interactions in our plan. We did a lot of work but the most important lesson we learned is the whole process of how to develop this visualization. We thought about the problem, the tasks, the encodings, the potential interactions beforehand, and made several sketches both in paper and electronically. Finally, we coded to implement the sketches. That's an amzaing tour instead of simply developing a web page. The initial abstract problem, the task analysis and the design are all crucial. And finally we understand a visualization can indeed provide much information because we have a feeling that we can always add some more functions into our visualizations. Moreover, although both of us had some experience of web development, we still found it interesting to know more about the power of HTML, JavaScript and especially D3 library.

7.2 Usability Testing

From the usability testing, we got some helpful feedback and suggestions.The most important point that we needed to improve was that the explanation of the visualizations were quite weak. Afterwards, we added detailed explanations and instructions on how to use the tool. Besides that, several detailed technical problems were proposed from the testing, and it created a great chance for us to look back and reflect on our visualizations. We've been trying to fix all issues we had.

7.3 Future Work

There are plenty of potential future work we could do to improve the visualizations and include more functionalities. We list several in the following:

- Include different types of functions. At present, our visualizations can only deal with linear functions now. Actually, there are various types of ranking functions. For example, the ranking score can be the logarithmic weighted sum, or we can simply use lexicographical order to rank the results.
- Add more interactions between multiple rankings like lines connecting the tuples to show the changes. Also, marking one result and tracking its moving trend (up or down) in these rankings will be very helpful. The use of native HTML table obstructed us from implementing these with ease. We are also considering constructing all rankings using SVG such that it will be easy for us to add the interactions but harder for us to build and manage the rankings.
- Try to think about other applications. Now we only focus on theta-joins but it is potential that these visualizations can be used for other applications, maybe just after small modifications.

8 Conclusion

In this project, we propose a visualization tool to help researchers understand their ranking results of theta-joins better including why certain results appear at the top of the ranking and how one ranking function and its parameters influence the ranking. We present a single ranking view to show a single ranking table with a scatter plot showing its distribution and a multi-ranking view to compare different rankings. The visualizations haven't been put into use yet, and we may try to improve them by adding more ranking functions, adding more interactions between rankings, etc.

Acknowledgements

We appreciate all the things Professor Cody Dunne did for us in this project including the initial suggestions, the feedback and all the comments. Thanks for the good class! We also appreciate the help from all the TAs and thank the students of this course, CS7250 for testing our visualizations and providing feedback. Finally, we thank our partner Mirek Riedewald and also his collaborator Nikolaos Tziavelis for discussing about this project with us.

References

[1] E. Boros, B. Kimelfeld, R. Pichler, and N. Schweikardt. Enumeration in data management (dagstuhl seminar 19211). *Dagstuhl Reports*, 9(5):89–109, 2019. doi: 10.4230/DagRep.9.5.89

[2] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. Lineup: Visual analysis of multi-attribute rankings. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2277–2286, 2013. doi: 10.1109/TVCG.2013.173

[3] S. Kumar, B. Hooi, D. Makhija, M. Kumar, C. Faloutsos, and V. Subrahmanian. Rev2: Fraudulent user prediction in rating platforms. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 333–341. ACM, 2018.

[4] S. Kumar, F. Spezzano, V. Subrahmanian, and C. Faloutsos. Edge weight prediction in weighted signed networks. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pp. 221–230. IEEE, 2016.

[5] A. Leventidis, J. Zhang, C. Dunne, W. Gatterbauer, H. Jagadish, and M. Riedewald. Queryvis: Logic-based diagrams help users understand complicated sql queries faster. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, SIGMOD '20, p. 2303–2318. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3318464.3389767

[6] R. Pienta, F. Hohman, A. Endert, A. Tamersoy, K. Roundy, C. Gates, S. Navathe, and D. H. Chau. Vigor: Interactive visual exploration of graph query results. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):215–225, 2018. doi: 10.1109/TVCG.2017.2744898

[7] A. Puri, B. K. Ku, Y. Wang, and H. Qu. RankBooster: Visual Analysis of Ranking Predictions. In A. Kerren, C. Garth, and G. E. Marai, eds., *EuroVis 2020 - Short Papers*. The Eurographics Association, 2020. doi: 10.2312/evs.20201068

[8] C. Shi, W. Cui, S. Liu, P. Xu, W. Chen, and H. Qu. Rankexplorer: Visualization of ranking changes in large time series data. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2669–2678, 2012. doi: 10.1109/TVCG.2012.253

[9] N. Tziavelis, D. Ajwani, W. Gatterbauer, M. Riedewald, and X. Yang. Optimal algorithms for ranked enumeration of answers to full conjunctive queries. *Proc. VLDB Endow.*, 13(9):1582–1597, May 2020. doi: 10.14778/3397230.3397250

[10] N. Tziavelis, W. Gatterbauer, and M. Riedewald. Optimal join algorithms meet top-k. SIGMOD '20. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3318464.3383132

[11] E. Wall, S. Das, R. Chawla, B. Kalidindi, E. T. Brown, and A. Endert. Podium: Ranking data using mixed-initiative visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):288–297, 2018. doi: 10.1109/TVCG.2017.2745078

[12] D. Weng, R. Chen, Z. Deng, F. Wu, J. Chen, and Y. Wu. Srvis: Towards better spatial integration in ranking visualization. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):459–469, 2019. doi: 10.1109/TVCG.2018.2865126

[13] X. Yang, M. Riedewald, R. Li, and W. Gatterbauer. Any-k algorithms for exploratory analysis with conjunctive queries. In *Proceedings of the 5th International Workshop on Exploratory Search in Databases and the Web*, ExploreDB 2018. Association for Computing Machinery, New York, NY, USA, 2018. doi: 10.1145/3214708.3214711

9 GROUP CHARTER

Group Purpose: Mainly, we considered two reasons for this group formation. First, both of the group members are PhD Students from the Data Lab, Northeastern University. We have been working together on the different aspects and algorithm related to the data Integration, data discovery, and optimization. Being from the same research group, it becomes easier for us to understand the ongoing works, sync up with the ideas, and divide the responsibilities efficiently. Second, it is very important for us to understand the requirement of our Service-Learning Partner. With both of us doing our research on the same field that Professor Riedewald, our service-Learning Partner, is working, it becomes easier to understand his requirements and design the visualization accordingly.
Apart from our Service-Learning Partner, the researchers from the entire database community will be benefited by this work. As Ranked enumeration over join is itself a novice concept, our main purpose is to explain the obtained ranked results with visual representation. This helps to enhance the trust over the ranking algorithm and hopefully, will open further research directions.
Group Goals: Our initial goal will be to implement the visualization using obvious techniques as discussed in section 1. After that, we will focus on integrating more encoding as per the expectation and requirements of our partner. Furthermore, we also want to dive deep to learn the visualization techniques and rules that can aid us in our future research as well. Both of us are very interested in this project and we are aiming for a collective grade of A in the class.
Group Member Roles/ Responsibilites and Ground Rules: We have divided our responsibilities on the following basis: Technical Responsibilities and Organizational Responsibilities. Both members are provided with the sub-responsibilities under these two main responsibilities. On technical part, we have to deal with algorithm implementation, front-end design, and a little bit of back-end work with flask, if necessary. There will also be further granularity on front-end design. We will divide these pieces of works, where Zixuan will focus on algorithm and parts of front-end design and Aamod will focus on the use of flask and remaining front-end works. In organizational part, we have documentation and communication. Zixuan will lead this project as a Communication Director whereas, Aamod will handle the documentation and paper works. During interview with the learning partner, Zixuan will focus on preparing and asking the questions whereas, it will be Aamod's responsibility to log the meeting details. Each week, a portion of work will be assigned to each of the member. For accountability and progress evaluation, we will have two meetings every week. One will be a regular group meeting between two of us. Next, we will have combined meeting with our Service-Learning Partner to update the weekly progress and to receive his feedback.
Potential barriers and coping strategies: With semester going on remotely, we considered the lack of in-person meeting opportunities as the major barrier on this project. However, we will use the available platforms to coordinate as far as possible.
Updates: So far, we are clear that we do not need the backend computations for our project. However, there is an added responsibility of writing the MapReduce program to perform the initial self-joins. Also, if the implementation becomes slower while considering all linear and higher order functions, we need to perform more pre-computation and generate the ranking results for some specific functions. We intend to replace the backend work related to flask with this pre-computation job using MapReduce. The backend job was initially assigned to Aamod. Therefore, to keep our work division balanced, it will be Aamod's responsibility to perform these pre-computations. This is the only major update from our group charter. Apart from this, we are able to follow our plans and rules properly without problems. Furthermore, we are having a regular meeting between us every week and also with our Service Partner, whenever required. Both of us are comfortable with our group roles and the task division is working well for us.

APPENDIX

A INTERVIEW

Fig. 2 lists our interview questions. Fig. 3, Fig. 4 and Fig. 5 are the interview notes.

1. Could you please explain your high-level requirement for one more time?
2. Who are the targeted users? We know they can be Database researchers because we provide some new technique. But, do we have more potential users?
3. (If we have more potential users,) what can be their mission? In another way, what we can provide to them and how can we benefit them?
4. As discussed earlier, about the data we are going to use, we think the Twitter data will be fine. Do you think it's adequate?
5. Could you maybe explain more about why users want to analyze these data? The user motives.
6. What do you think are the users looking for in the data? That will be what we are going to try to provide.
7. Are you already using some visualizations?

Figure 2: High-level Interview Questions

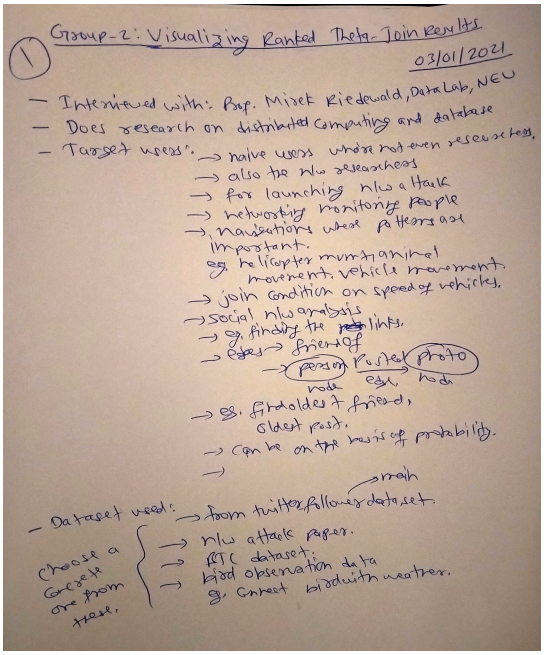

Figure 3: Interview Note (First Page)

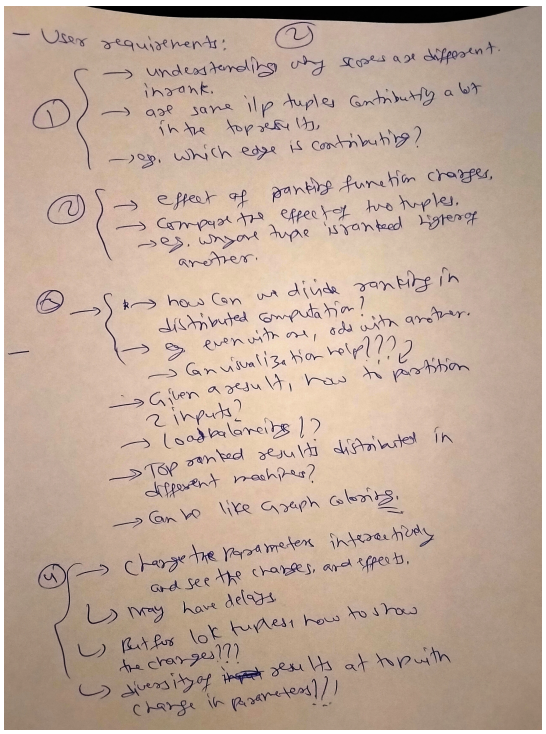

Figure 4: Interview Note (Second Page)

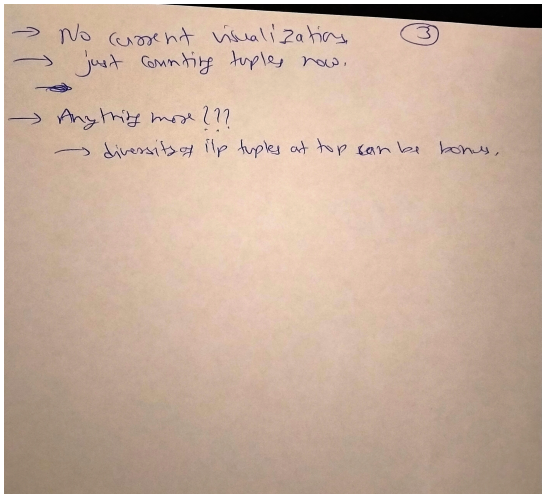

Figure 5: Interview Note (Third Page)

B Data Exploration

We are working with the Bitcoin who-trust-whom dataset for the visualization project. This is a graph dataset containing the ids of the users and the trust-rating they provided to other users. All these users trade using bitcoin on Bitcoin OTC plaform. The dataset has a total of four columns where, the first column is Source user ID, the second column is Target user ID, the third column is rating score that the source user gave to the target user. It ranges from -10 (total distrust) to +10 (total trust). The last column is the rating submission timestamp. However, this dataset is not appropriate for the visualization. Therefore, we wrote a reducer-side join program using MapReduce to perform a self-join for two times to get the length-3 path dataset. After self-join, the schema of our dataset is: (node1, node2, node3, node4, score1, score2, score3). Henceforth, the term dataset refers to the joined length-3 path dataset that we generated using the Map reduce program.

Data Types

Our dataset contains a total of 7 columns and 83,074,108 rows. Their data types are explained below:

- The first four columns refer to the first node, second node, third node, and fourth node in the graph respectively. All of these columns are categorical columns.
- The fifth column is the weight for first edge i.e. from node1 to node2. It is quantitative data.
- The sixth column is the weight for second edge i.e. from node2 to node3. It is quantitative data.
- The last column is the weight for third edge i.e. from node3 to node4. It is also quantitative data.

Potential Issues

Since our dataset is very large containing more than 83 million rows, we are concerned about the interactivity and latency of our visualization. Apart from this, our dataset is well-structured, well-normalized and does not contain null values.

Insights

The original dataset contains 35,592 edges, which increases to over 83 millions after join. Therefore, it is not suitable to use the original dataset for the visualization, as we need to perform joins interactively, which is very time-consuming. Therefore, we pre-computed the length-3 paths using a MapReduce program to obtain the dataset that we can use for our visualization. To explore the data, we use Tableau and get some insights. First, we are very happy that Tableau can't cover all the ideas we want to implement. Some visualizations generated by Tableau do provide some useful figures. Most of the ratings are either 1-5 or 10/-10, and most of the rating sum of a path also tends to be small, showing a trend of low rating between nodes. Also, the distribution is quite even, which means there does not exist some node in the graph which is a neighbor of all other nodes, which is really good for visualization and further analysis.

The raw data are not adequate for our tasks so we are using a processed dataset now. As mentioned earlier, we do not see any serious issues with our data other than its huge size.

Screenshots

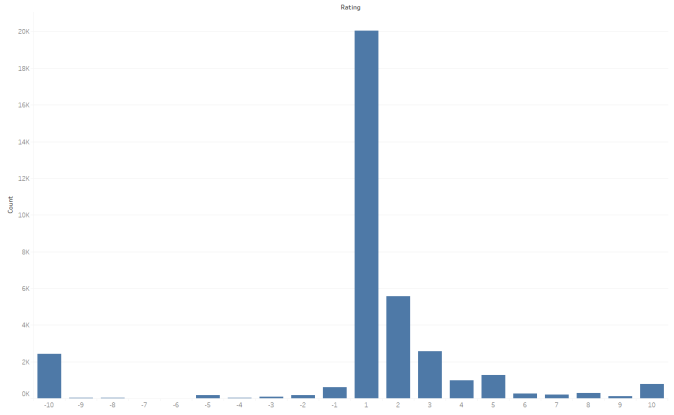

Figure 6: We're using raw data of the bitcoin dataset, showing for each rating from -10 to 10, how many edges have it. Bar charts are used because in this way the comparisons of 20 ratings can be showed clearly. We found that the ratings accumulate from 1 to 5 as well as 10 and -10. These are ratings usually used in the graph.

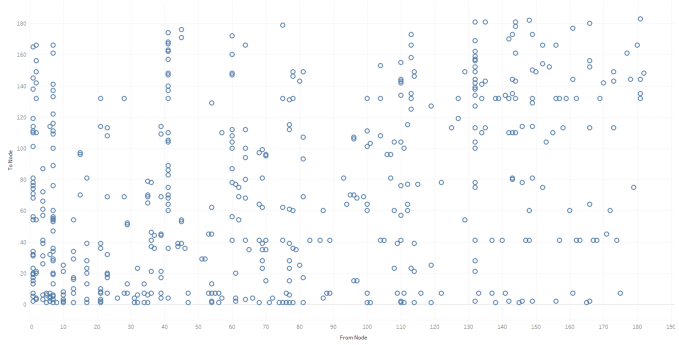

Figure 7: We're using a subset of the bitcoin dataset which contains 500 edges. Since there are not excessive edges, we use a scatter plot where each point represents an edge in the graph. Scatter plots is very clear for us to see the distribution and even which two nodes share an edge. In this figure, we can see the edges are distributed quite evenly, which is very good for our visualization and further analysis.

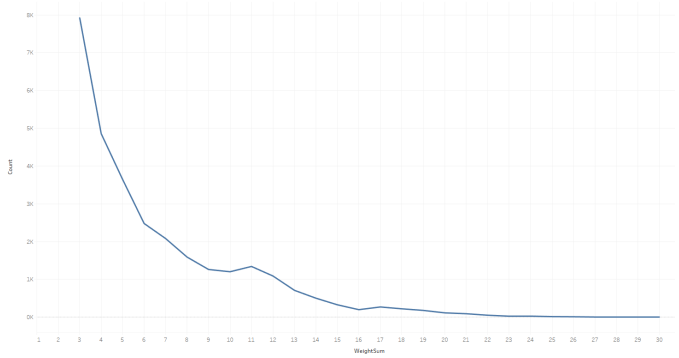

Figure 8: We're using the dataset after processing. We define the weight sum to be the sum of all edge weights in one path. In this visualizaiton, a line chart is used to show for each weight sum, there exist how many paths. By a line cahrt, we can clearly see the comparisons and the dropping trend from small weight to big in this figure.

C Design Sketches

We are choosing the Sketch 1,2 of Zixuan Chen and the Sketch 2 of Aamod Khatiwada as our favorite sketches. As we discussed with Professor Cody Dunne and got some exciting ideas before beginning our sketches, our sketches are driven with the common concepts.

The Sketch 1 of Zixuan Chen is a simple and clear sketch for Task 1, and it can still borrow the idea of Aamod to use another hue for negative weights. By using both luminance and width as channels, this visualization perfectly satisfy the requirements in Task 1. We can easily compare the edge weight and how much the ranking function depends on that edge to get the reason why some results rank top. By using diverging color set, we can even directly see which edges are positive and which edges are negative.

The Sketch 2 of Zixuan and the Sketch 2 of Aamod are quite similar, and we think they can be merged together for Task 2. The single view could be the same as for Task 1 but in this visualization we are doing some comparisons between multiple rankings. In this visualization, we can easily compare a couple of ranking functions by some highlights or links and tune the parameters to create (and for sure remove) ranking views. This visualization can be interactive by tracking the positional move of a certain path in several rankings.

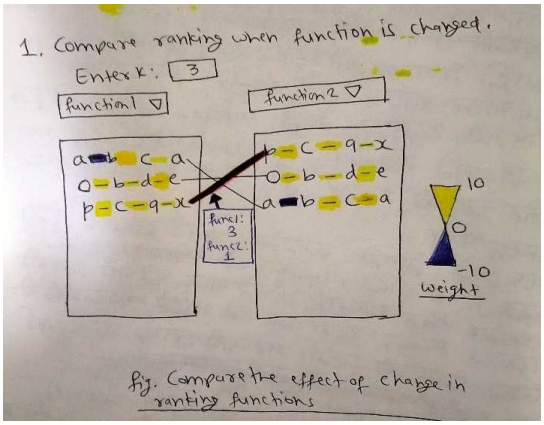

Figure 9: We need to encode the edge weight and the rank of a path with different functions for entered k. So, I encoded the weights of the edges by the size channel together with color. The green color with largest width indicates a weight of 10 whereas, the blue color with largest width indicates -10. For tracking the rank of a path with different ranking function, I used an edge to connect them and with mouseover on the edge, we can view the details about ranking and position shift. It is related to Task 2. By: Aamod Khatiwada

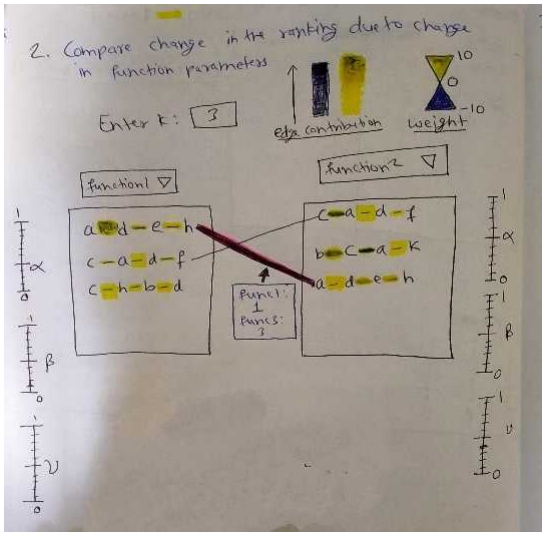

Figure 10: **Favorite**The diverging pair of green and blue colors together with size encodes weight of each edge. The luminance channel encodes the importance of each edge in the ranking for the given function. In function 1, the thick edges are important whereas the function 2 prefers thin edges. The edges with high luminance are the important one. The user can tune the parameters α, β, and γ for both functions and see the change in ranks. This visualization is related to Task 1. By: Aamod Khatiwada

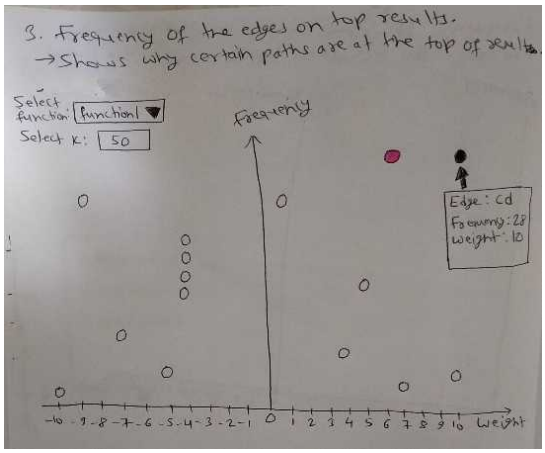

Figure 11: I used position encoding for this plot. I also used hue encoding to highlight the most-frequent edges in the top-results (pink-colored). The user can change function and the value of k in an interactive manner. This visualization is related to Task 3 and looking at the repetition of edges in the top-results, we can distribute the important edges evenly in the parallel computations. Also, for the given number of machines, we would use more hues to encode edges such that the edges with same hues are provided to the same machine. By: Aamod Khatiwada

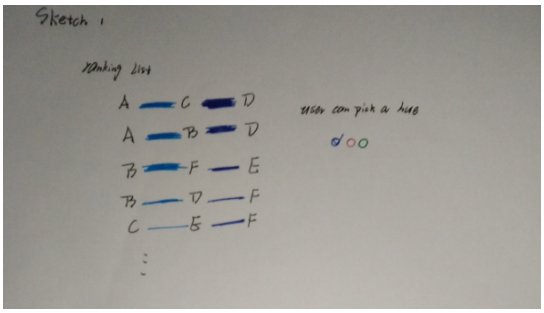

Figure 12: **Favorite**This sketch is by Zixuan Chen. I am using a ranking table for this visualization with line width and luminance as channels. This visualization is for Task 1. First, some preliminary knowledge about our ranking is that we're using a ranking function with some parameters combining with the edge weight to get the ranks. Since we want to show why certain results rank top, we need to show two things in the visualization, the edge weight and how much the ranking function depends on that edge. Thus the line width represents the edge weight and the luminance represents how much the ranking function depends on that edge.

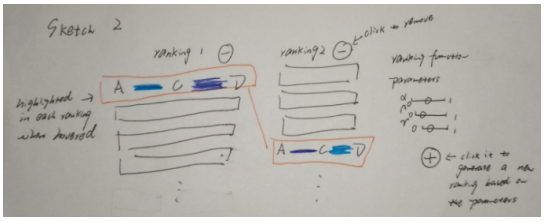

Figure 13: **Favorite** This sketch is by Zixuan Chen. I am using multiple ranking tables for this visualization with line width and luminance as channels. This visualization is for Task 2. For single view, it is the same as my sketch 1. The difference here is in this visualization, we want to highlight the difference when we are changing the parameters of a ranking function. On the right, there is a parameter tuning panel. A user can tune the parameters and create a new ranking. If one row in a ranking is hovered or clicked, its positions in all the rankings will be highlighted. In this way, we know how the changes in a ranking function affect the ranking results.

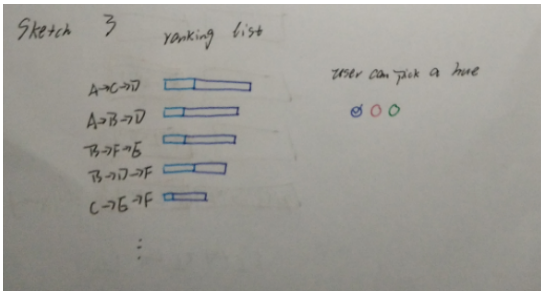

Figure 14: This sketch is by Zixuan Chen. I am using a stacked bar chart for this visualization with bar height and luminance as channels. This visualization is for Task 1. Since we want to show why certain results rank top, we need to show two things in the visualization, the edge weight and how much the ranking function depends on that edge. The stacked bars can show both the rank score of one edge and the rank score of this path, and the luminance represents how much the ranking function depends on that edge.

D Digital Sketches

In the following, we present our digital sketches. We are focusing on Task 1 (Single Ranking) and Task 2 (Multiple Rankings). We didn't add new tasks. Actually, for Task 1 and Task 2, we think they are of the same importance because Task 1 is sort of one part of Task 2. We need to understand what a single ranking function is doing to further understand how different ranking functions affect the ranking results. For single ranking, we're using width to represent edge weight and luminance to represent how much the ranking function relies on this edge (to be more specific, parameters like α). Fig. 15 shows our single ranking mode. In this mode, a user can use a mouseover event to show the appearances of a certain edge. We think by means of this visualization, a user can understand how the results are ranked. Fig. 16 and Fig. 17 show our vths by clicking and brushing on paths and use that to think about the influence of different fuisualizations for multiple rankings. Users can clearly observe the movement of certain panctions. There is no noteworthy change in our plan so far.

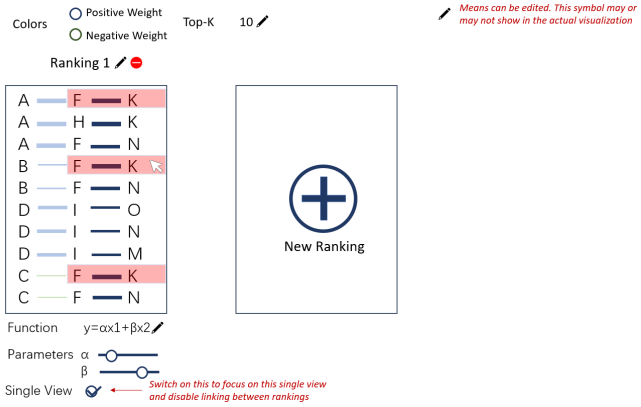

Figure 15: Digital Sketch1 - Single Ranking

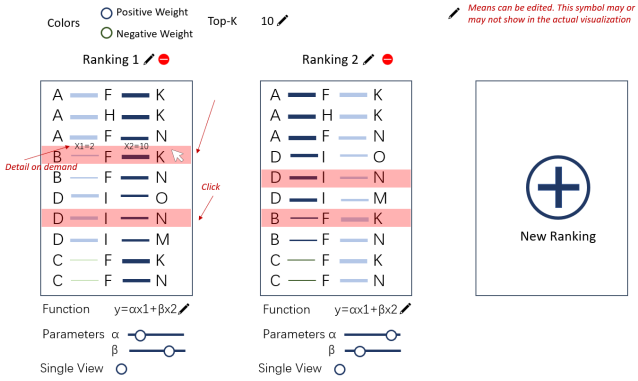

Figure 16: Digital Sketch2 - Multiple Rankings

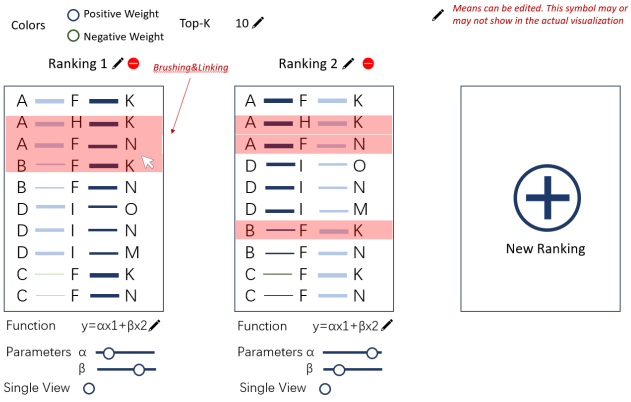

Figure 17: Digital Sketch3 - Multiple Rankings (Brushing)

E REFLECTIONS

Aamod: We had two major goals at the beginning of this project. First, we wanted to visually explain the top ranked join results given by a ranking function. Second, we wanted to compare the outcomes of different ranking functions and see how it changes with the change in function parameters. To build a visualization framework for these tasks, we started with a graph dataset containing the edges and their weights. We self-joined the graph for two times to obtain the three hop paths having four nodes and three edges. Each edge has its own weight. Now, given a ranking function f(x,y,z), it uses the edge weights as the roots (i.e. x,y and z) and user-defined parameters α, β and γ as the coefficients. For a given value of positive integer k, the ranking function outputs top-k paths in either ascending or descending order. Our visualization was found to be very effective in understanding the ranking . We designed two separate visualizations for two separate tasks. First, we used a table visualization together with scatterplot which explained the ranking results of a ranking function. Looking at the weight distribution on scatterplot, one can easily catch the outliers and understand why they are present in the result. In most of the cases, the outliers received supports from the neighbouring edges to exist in the top ranks. Also, by changing the function, the user can easily see changes on the results and decide what could be the best ranking function that can eliminate the outlier edges as far as possible. Furthermore, our second visualization was also found to be effective in comparing the ranking results of different functions at the same time.

As we were working with a huge dataset, it was challenging for us to select the right subset of data that explains the significance of each visual encodings we used. Furthermore, the latency of the system was another concern. However, we were able to select the proper subset of dataset that offered reasonable interaction speed as well as covered all the visual concepts.

I was able to learn the use of D3 library and also sharpen my visualization skills like selecting good marks, using proper encoding and adding the interactive elements.

Initially, I was expecting the visual design to be the challenging part of data visualization. However, since different users can have different perception towards the same visual encoding, I realized that designing a good visualization is not enough. Proper story telling is equally important to make it useful for the target users.

Our group was the perfect combination for this project and I really enjoyed working with my teammate. The obvious future step for us after this project is to append more diverse ranking functions and expand the top-k ranking concept to any-k ranking. Also, we want to implement an effective sorting algorithm for huge dataset to increase the interaction speed. Most importantly, I learned the importance of storytelling together with visual design and I would take this into account whenever I get an opportunity to participate in the visualization projects.

Zixuan Chen: First I met our Partner Prof. Mirek Riedewald and talked about his visualization needs. He got some exciting work about ranked enumerations of theta-join and felt that it would be interesting to see some visualizations. I liked his idea and thought visualizing for rankings would be useful and related to my research focus since I am in the Data Lab. Aamod was interested as well and joined me so I got a reliable teammate. Our initial aim was to use visualizations to help explain theta-join ranking results to database researchers. We used the Bitcoin network data where we used the nodes to represent tables, the edges to represent the join relationship and the edge weights to represent the theta-join strength in our theta-join background. Then we consulted Prof. Cody Dunne for his advice and understood we could use channels like line width and luminance. Finally, there goes the visualization page as we present now. For two major questions we wanted to answer, which are "How to use a visualization to explain why some results are at the top of a certain ranking?" and "How to user a visualization to show the influence of a ranking function?", we present the single ranking view and multi-ranking view respectively.

Actually, I had some background about D3 before the course. Still, after the project, I think I know more now. For the first time, I know the brushing and linking function, which is very useful and we applied it. One interesting class is that you can always learn from others. When we finished a part of the functions, we always felt that it was good now but it turned out that you can always make the visualization better. Just go to ask the opinions of someone else, then you get inspired again. Aamod and I are in the same lab, and we have experience working together so it is not so hard for us to collaborate. I am leading the team and sometimes I need to make a decision where I need to accept some opinions of others but sometimes also decline some suggestions.

From the project and the course, I learned a lot about visualization. I'd be happy to join other visualization projects or use visualization tools to help my research in the future. About the project, I want to make it connected to the algorithms provided by our partner and maybe make it one part of his work to increase efficiency.

F SLIDES

The link to our slides: https://docs.google.com/presentation/d/1gtVEIwBwWzoJ6WOdH_67MMYI6nd76qPTZ3BZ3XAglKM/edit?usp=sharing.