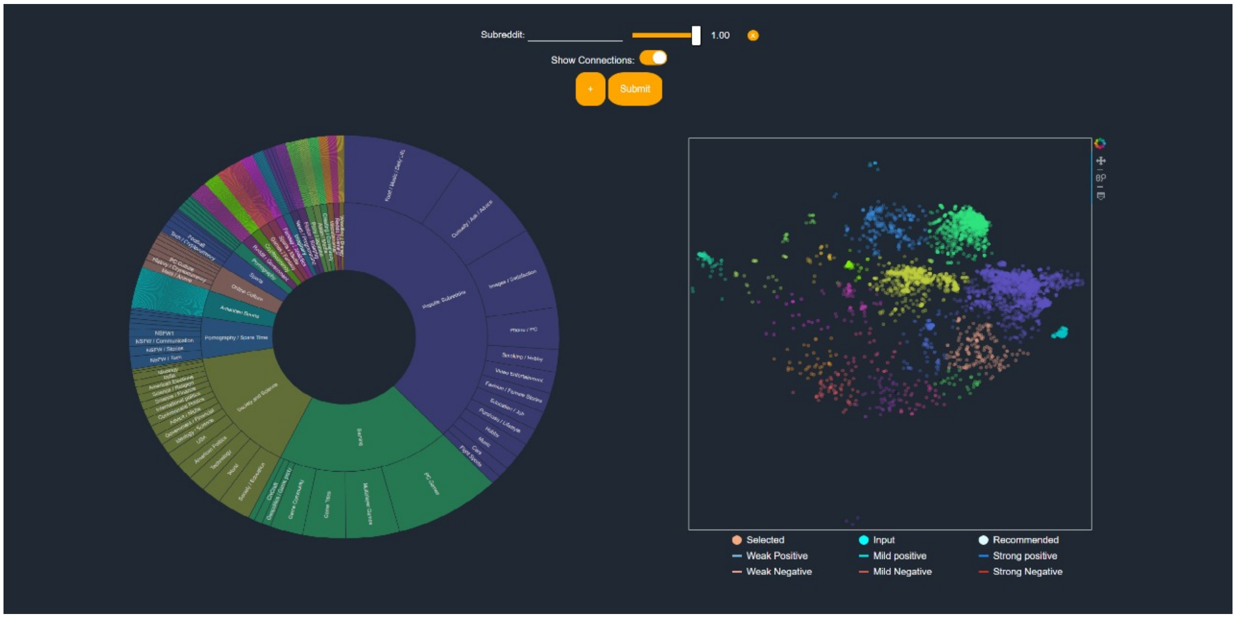
Visualising Subreddits Recommendation System

Hinrik Gudmundsson 12675326
Albert Harkema 12854794
Eui Yeon Jang 12661635
Christiaan van der Vlist 12876658
Zi Long Zhu 10654860



ABSTRACT

In this paper we present a visual analytics system that allows users to navigate through and gain insight from a data set of mobilisations between different internet communities. The system consists of a network graph and sunburst for navigations, and a recommender system and community clustering to see how the communities relate to each other. We evaluate the system through a questionnaire that test users filled in after exploring the system. These results were varied, with some users finding the system decent and others finding it poor. This variation seems to stem from the background of the test users, the nature of the communities given as input, and the limitation of the data set used. Taking these into account to further improve the system may enhance the user experience as well as the insight provided by the system.

1 INTRODUCTION

In the current age, online communities, such as fora and social media groups, serve as popular platforms for users to interact with one another, expressing and sharing information, ideas and opinions. These communities allow users to engage in interactions with others within their community as well as across different communities. In order to better understand the mechanism of these interactions, Kumar et al. [3] have collected and analysed a data set that captures the explicit sentiment of interactions between different communities on Reddit. The authors of the paper focus on mobilisations – instances where a post in a source community hyperlinks to a post in another, target community mobilises users to comment in the target post with a certain sentiment.

Unfortunately, the magnitude of the information contained in the data set makes it difficult for humans to make intuitive sense of the data, gain insight from it, or reason with it. In this paper, we seek to develop a tool that aids users in better understanding the relations between these communities, called subreddits. To do this, we must answer the question of how to clearly visualise both the data and the insight the tool provides.

In this paper, we take the approach of developing a visual analytics system for the aforementioned data set in the form of a recommender system. A visual analytics system "...uses data to draw conclusions about the world, a process, or an application field" [7]. Our system recommends subreddits to users based on an input of subreddits and respective weights that represent the importance of the source subreddit. We stipulate that the input subreddits should have been the source of a mobilisation. The system uses a network graph to trace the flow of sentiment between subreddits starting from the input subreddits, and recommends the subreddits that receive the highest aggregated sentiment. Visually, the system consists of two major components. The first is a network graph that represents the data on both a granular level and a more global level through the use of clusters. The second is a sunburst that allows users to browse or explore clusters of subreddits. These two components interact, ensuring that the network graph always shows the most relevant subreddits. When an input is given, the resulting subgraph network that was responsible for finding the recommendations is also displayed. The weights for each subreddit can be updated after the recommender system has found the recommendations, causing it to rapidly reevaluate and update the network if necessary. This functionality allows users to explore what-if scenarios where a source subreddit that in reality has positive sentiment towards a target subreddits now has negative sentiment, or vice versa.

Furthermore, the clusters allow users to easily view what subreddits are similar to each other. In this way, a user can see how sentiment flows between similar subreddits, how it flows between different clusters, and how the input subreddits related to the recommended subreddits, i.e. the ones that are viewed most positively by the input subreddits. All in all, this allows users to gain insights into the interactions and relationships between different subreddits.

The corresponding code for the system can be found in the author's github repository. [1]

The paper is structured in the following manner: Section 2 discusses related work. Section 3 describes the data set in more detail and explains the methods of developing our visual analytics system. It also discuss the scheme for qualitative evaluation of our system and discuss the results of the evaluation in Section 4. Finally, Section 5 contains the conclusion of this paper and suggests further work that can be done.

2 Related Work

The paper [3] that the dataset was originally presented with focused on finding instances of negative mobilisation and the subsequent influence these had on the targeted communities. They found that such negative mobilisations indeed have lasting negative effects whereby ill-behaved users would come to dominate the targeted subreddits. This highlights the importance of being able to predict and gain insight into mobilisations and their consequences.

Belák et al. [2] introduce a framework that can be used to find which the level of influence and dependence that communities have on others.

The work of von Landesberger et al. [9] gives an overview numerous techniques, alongside their advantages and disadvantages, that can be used to create intuitive and aesthetically pleasing representations of graph data. Moreover, they provide papers that have actually applied these techniques and discuss the reasoning behind them in those specific cases.

3 Methods

To allow for the best user experience we have designed our system in the following way. Firstly, we provide the user with input fields and a submit button. The user can increase or decrease the number of subreddits the user wishes to input by adjusting the number of input fields. Each input field has a corresponding slider, with which the user can adjust the weight the user wishes to attach to the specific subreddit. Once the user is ready to train a model and generate recommendations, the user can press the submit button. Secondly, we have a sunburst graph as shown in Figure 1. The sunburst graph contains named clusters of subreddits, allowing the user to more easily navigate all the available subreddits. Moreover, we provide a network graph, displayed in Figure 2 that shows all of the subreddits as nodes in their respective clusters in a network. These cluster colours correspond to the cluster colours in the sunburst graph, allowing for easier interaction between the two graphs.

3.1 Data Set

Kumar et al. [3] provide a subreddit hyperlink network data set extracted from publicly available Reddit data from January 2014 to April 2017 [2]. The network was extracted from posts that create hyperlinks from one subreddit to another, referred to as the source and target subreddits, respectively. The data set contains hyperlinks extracted from titles and bodies of the posts. This data set contains 55,863 unique subreddits and 858,490 hyperlinks. We use a subset of this data set, focusing on the hyperlinks extracted from bodies of the posts and we further narrow it down by considering the top 5,000 most occurring target subreddits and their respective hyperlinks. Our final data set contains 5,000 unique subreddits and 59,692 hyperlinks. Each hyperlink is annotated with a timestamp, a binary sentiment label of the source post towards the target post, and a property vector containing various features of the source post. Instead of the binary sentiment label, we make use of the real-valued VADER sentiment scores given in the property vector. The authors also provide 300-dimensional embeddings for each subreddits [3], represent underlying latent representations of the subreddits [4, 5].

3.2 Architecture

The architecture of the recommender system follows a standard client-server approach. The server implements a RESTful API that provides communication between the server and the client over HTTP. We use Flask as it is a light weight framework that provides all the HTTP functionality required for our RESTful API. It is a design choice that the server be responsible for all data processing and computationally heavy tasks. By setting up a server that is responsible for heavy computation, we keep the client as light as possible and improve the performance on the user end.

The client has the main purpose of providing the visualisations and the interaction functionality to the user for visual analytics. The client is also responsible for managing user input that is later passed to the server for processing. The client then receives the results from the server and displays it to the user. The React framework is used for providing the client side of the project for flexibility, scalability and ability to provide dynamic user interfaces that can change data without reloading the page.

3.3 Recommender System

For our recommender system we had a couple of key points that were paramount to its success in a visualisation application. Firstly, and most importantly, it should provide the user with accurate recommendations. Secondly, it should be quick to update, because the user should not wait a long time for an update in the visualisation. Thirdly, to tie the visualisation into the broader scope of visual analytics the recommender should retrieve the path in the network that led to its prediction. Moreover, the user should be able to interact with the system and data by adjusting the weights that control parts of the recommendation.

3.3.1 BFSuggest

BFSuggest is the main part of our recommendation algorithm. It is an alteration of the Breadth First Search algorithm to perform a graph search. BFSuggest takes as input the subreddits that the user selects and starts a search. For each input subreddit, it expands five nodes that have the highest outgoing sentiment, and appends these to the queue. At each level, it continues the search of the top five subreddits with highest sentiment scores until a predefined depth cutoff is reached.

During this process, it builds a dataframe keeping track of the total aggregated sentiment score for each subreddit BFSuggest has expanded. It saves these scores per input subreddit as an entry in the dataframe. To provide the user with recommendations, we sum over the scores related to the different input subreddits and show the top five final scores. Saving the scores in a dataframe in this way allows for quick interaction. Once the user adjusts the weight for a specific input subreddit, we merely have to multiply the weights of that original subreddit's column, add up the scores and update the top five recommendation. Doing a complete update of the recommendations or changing recommendations based on a slider change only takes about 200 milliseconds.

3.3.2 DLSuggest

However, BFSuggest expands numerous edges, some of which do not contribute to the recommendations because their edges is not on the path leading to the recommendations. To tie in more to the visual analytics framework, we also wrote the DLSuggest algorithm. DLSuggest performs an altered version of a Depth Limited Search

[1] https://github.com/Sasafrass/infoviz10

[2] https://snap.stanford.edu/data/soc-RedditHyperlinks.html

[3] http://snap.stanford.edu/data/web-RedditEmbeddings.html

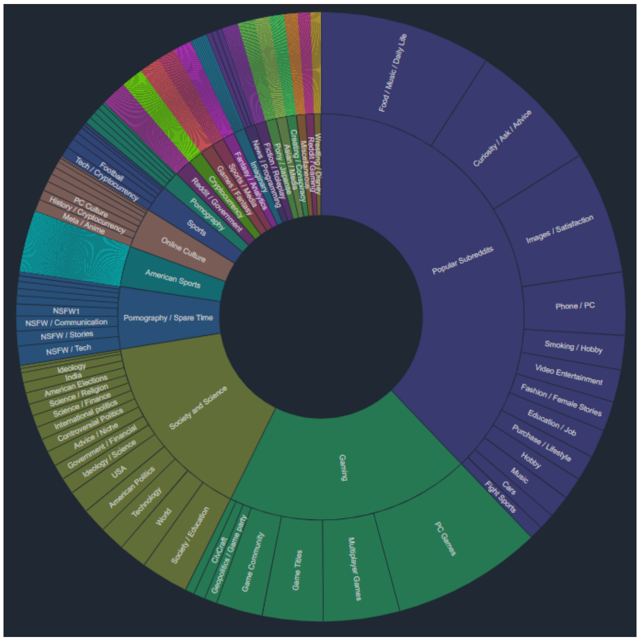


Figure 1: The sunburst graph. The different colours correspond to different clusters.

on all the edges that were expanded by BFSuggest, and retrieves the ones that ultimately contributed to the recommendation. This allows the user to see the sentiment links between a large number of subreddits, allowing for further analysis of the data.

3.4 Clustering

To have a better understanding of the subreddits and the associations and relations, we perform clustering. We use t-distributed stochastic neighbour embedding (t-SNE) [6] on the subreddit embeddings data set for dimensionality reduction, to project 300-dimensional subreddit embeddings onto a 2-dimensional plane for visualisation.[4]. The clustering was performed multiple times for large clusters and their subclusters.

We tune the *perplexity* hyperparameter for t-SNE which is a measure of how many close neighbours a point has. As a rule of thumb, a perplexity of 50 is used to cluster larger data sets and perplexity values of single digit range for smaller data sets [6]. After analysis of what gave us the best results, we decided to use a perplexity of 50 for the main clusters and a perplexity of 3 for the smallest clusters. Furthermore, we have used an early exaggeration of 25, which controls how much space is between the clusters. And a learning rate of 200 and a maximum number of iterations of 100,000.

We perform k-means for cluster assignments after the projections led to seemingly interpretable clusters. K-means works by randomly initialising a set number of clusters and iteratively assigning all points that are closest to this cluster and updating the location of the cluster based on the average location of its corresponding points [1]. The number of centroids used for k-means is were picked by hand. Usually a more sophisticated heuristics is used, e.g. the 'elbow method', however these methods require labelled data to measure the quality of the clusters. But we do not posses such data.

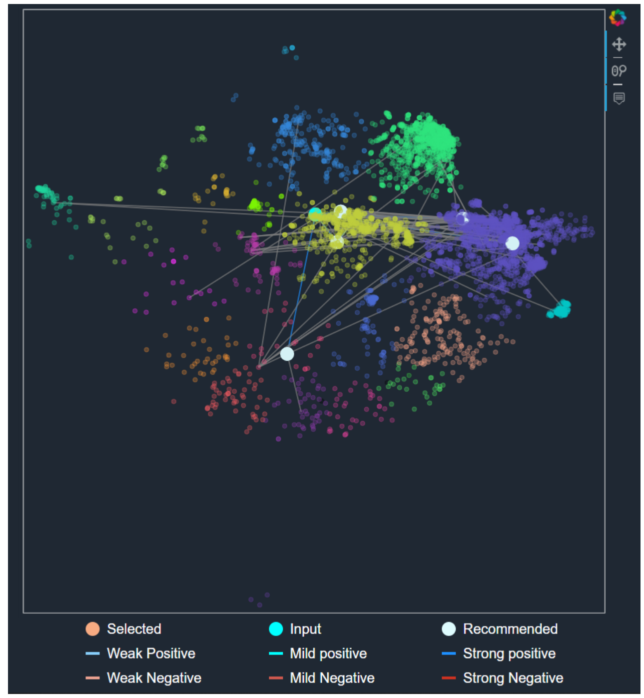


Figure 2: The network graph, with nodes coloured based on their respective cluster.

3.5 Sunburst Graph

The purpose of the sunburst graph is to give users a better overview of the available subreddits in the data set and allow users to explore the different associations between subreddits. It is also used as a selection tool, to select the input subreddits to submit to the recommender system.

Using the clustering, we generate a sunburst graph that visualises the hierarchical structure of cluster groupings. Showing the sub-clustering provides the users with a fine-grained means of selecting subreddits. Javascript and D3 is used to generate the graph, which allow for flexible animations, accessible interaction with the data and straightforward integration of the clustering data. For the user, this allows for intuitive interaction with the data which contributes to more understanding of the data and their structure. It can provide more insight as to why certain predictions are made. Moreover, it enables the user to uncover patterns for themselves in the clusters, potentially leading to finding other interesting subreddits in their respective clusters.

The sunburst graph colours the clusters and subreddits according to their top-level clusters, each top-level cluster a distinct colour. The colours were chosen to be bright to have contrast against the dark background of system.

The sunburst graph is tight-knit with the network graph. The graphs display the same clusterings from a different perspective and interactions with the sunburst graph is reflected in the network graph in highlighting of the node in the network graph of the selected subreddit.

3.6 Network Graph

In the network graph, subreddits are visualised as nodes and the edges represent that a mobilisation happened between the two subreddits, either unilaterally or bilaterally. The nodes are positioned using coordinates provided by the clustering algorithm, and they are coloured accordingly to their top-level cluster, as in the sunburst graph. Initially, the graph conveys the clustering of the subreddits without any edges between the nodes. This provides a different view on the same clustering alongside of the sunburst graph. As the user navigates through the different clusters on the sunburst graph, the nodes of the selected clusters will be highlighted in the network graph.

Once the user submits the input subreddits to the recommender system either using the sunburst graph or manually, the recommended subreddits are retrieved and shown in the network graph. The input and recommended subreddits are made recognisable by increasing the side of the node glyphs and changing their colours. Furthermore, the recommender system returns the path (edges) travelled to find the recommended subreddits. This path is also displayed on the network graph where each edge is coloured with blue, red, or grey to denote the positive, negative, and neutral sentiment of the links, respectively. The edge colours are also graded based on the magnitude of the sentiment. The different colours and their meanings are presented in a legend below the network graph.

The user can hover over the nodes to display the names of the subreddits the nodes represent. They can also zoom in and out to further explore the different edges and nodes in the visualised path.

3.7 Qualitative Evaluation

In order to evaluate our visual analytics system, we devise a short qualitative questionnaire using the review scheme introduced in [8]. A few test users are asked to answer the following questions using a 1-5 scale, where 1 is the lowest and 5 the highest:

Q1 Rate your overall satisfaction with the recommendations.

Q2 Rate your overall satisfaction with the system (application).

Q3 Rate the usefulness, efficiency, and intuitiveness of the visualisations.

Q4 Rate the usefulness, efficiency, and intuitiveness of the interaction.

Q5 Did you learn something new about the different relations between the subreddits?

Q6 How easy was it to understand the visualisation?

Q7 How easy was it to understand what an interaction did?

We define the terms 'usefulness', 'efficiency', and 'intuitiveness' for the users as follows:

- *Usefulness* : the system and the analytics process was useful in arriving at an answer and understanding of the situation.
- *Efficiency*: the system does not require any unnecessary nor repetitive steps.
- *Intuitiveness*: what step to take next and the visualisations are easy to understand.

[4]We have reached out to the authors of the data set from Stanford University to ask which hyperparameters they used for their t-SNE to increase reproducibility. However, we were not able to receive a meaningful response and therefore tuned our own hyperparameters.

The first question addresses the main question of our research, namely the recommender. Because the used subreddit embeddings data set [5] does not contain any ground truth recommendations, we set out to do a qualitative analysis to assess how well our recommender performs. Moreover, with the second question we wish to assess how well our system as a whole performs. Hereby, we aim to explore whether the visualisation responds swiftly to inputs, performs smoothly and is quick to update recommendations. The third and fourth questions target the usability of our visualisation and the interaction; in other words, whether it is intuitively easy the visualisation and the interactions are to understand.

In addition to the questions pertaining to satisfaction and usability, we assess whether our product helped people understand the data and whether it has shone a new light on the data for them. Finally, the last questions deal with the question of whether it was easy to understand the visualisation and how easy it was to understand the interaction. The answers to these questions can be summarised to form a qualitative assessment of the product as a whole. The results of this questionnaire are discussed in Section 4.

4 Results

15 test users were asked to evaluate our system. To perform the evaluation of the recommender system, the users navigated the system via screen sharing. Each evaluation session was conducted by one member of the group and who was also available to field any questions the user had. At the end, we would give the user some time to fill out the questionnaire and give an additional remark with respect to the system as a whole. As our questionnaire makes use of the Likert-scale, we treat each result as an ordinal variable. There was no set duration or default inputs for the users to try. They were free to explore the system and stop as they saw fit. The tallies of the scores given by the users are summarised in Table 1.

	Likert-scale				
	1	2	3	4	5
Q1	0	5	5	5	0
Q2	0	0	5	7	3
Q3	0	4	6	1	4
Q4	0	2	5	7	1
Q5	3	1	2	6	3
Q6	0	2	7	3	3
Q7	0	0	2	10	3

Table 1: Tally of scores given by test users per question

The scores seem to indicate that the performance of the system varies, at times working well and other times working fairly poorly. General remarks made by some of the participants also touched upon this subject and showed a similar range of assessments for the recommender system. While one user commented that "some of the recommendations seemed quite spot-on ", another mentioned "recommendations were a bit all over the place". A user noted that the "quality of recommendations varied. For my most visited subreddits they were way off, for some other ones they were okay.".

The scores are slightly higher regarding the overall satisfaction with the system (Question 2), compared to that of the recommender system considered individually. Some general remarks include "Flow felt a bit weird, first going down to the circle thing and then up again to submit." addressing the flow of the entire system. Another user remarked "Interesting visualisation. Very cool to see a global overview of how subreddits are related to each other and to what category each subreddit belongs. I'm impressed."

The somewhat varied response with regards to the general performance of the recommender can perhaps be attributed to the range of different inputs each user tested. Based on the results, we can distinguish two types of users - one type of user who chooses quite general subreddits such as 'iama', news, and 'politics', and a second type who tends to head towards highly specific subreddits such as 'greenbaypackers'. The former group tends to get worse recommendations than the latter. This can be explained by the inner workings of the recommender, which bases its recommendations on a graph search over sentiment scores. Because the sentiment scores linking to other subreddits for general subreddits are relatively scattered, the combination of their scores may not lead to very coherent recommendations. Contrarily, the sentiment links going out from very specific subreddits tend to stay closer to home. This may lead to more coherent recommendations. Ultimately, this leads us to believe that it is possible to provide recommendations based on sentiment but it seems more capable of doing so for subreddits found in more coherent clusters. The user's view of the system's performance depends on whether they tried on the inputs, the number of inputs, the time they spent exploring the system, and so on. The system as a whole receives a higher range of scores most likely due to the added consideration usability and aesthetics of the system.

The assessment usefulness, efficiency and intuitiveness of both the visualisation and the interaction (Questions 3 and 4) seem to receive similarly mediocre scores. A user remarked that the sunburst graph was difficult to navigate and another user commented that the visualisations and interactions were not intuitive and required more information with regards to what the scores for the recommendation means and what the edges in the network graph meant.

Responses with regards to newly gained insight (Question 5) showed the widest range of scores. This could be attributed to the the users' familiarity with Reddit. Some users who had not used Reddit before commented that this was an interesting look into the platform - "Very cool to see a global overview of how subreddits are related to each other and to what category each subreddit belongs". Comments from experienced Reddit users ranged from being confused about the connections between subreddits to being able to find new relevant and interesting subreddits during the evaluation.These comments also seem to indicate a limitation of the data set as well as the algorithm. Due to having to reduce the size of the data set, it is possible that some of the relevant connecting edges were neglected. Furthermore, having fewer subreddits and data available meant that the users were limited in what they could explore. Moreover, in attempts to produce and fast recommender system, we took a naive, albeit surprisingly well performing, algorithm. Perhaps if we were to make use of more advanced algorithms, we may have been able to produce different recommendations. However, the issue of a small data set persists.

Finally, Questions 6 and 7 deal with how easy it is to understand the visualisation and the interaction. The responses to these questions indicate that it was fairly easy to understand what an interaction did and a bit more difficult to understand what a visualisation showed. This is also illustrated in the general remarks given by participants - "there's maybe too many subreddits in the sunburst graph, but the idea is nice. It is easier to navigate the general graph though." indicating that the interaction was easier to understand than the visualisation.

5 Conclusion

In order to gain more insight into the large data set of mobilisations between subreddits originally published by Kumar et al. [3], we introduce a visual analytics system that allows users to browse, explore and gain insight into the data. The system consists of a recommender that uses simple algorithms akin to breadth-first search and depth-first search to find recommendations quickly, a clustering of subreddits that maps similar subreddits close together, a sunburst graph to allow for easy navigation and browsing through the different clusters and a graph to visualise both the clusters and recommendation networks.

In order to evaluate the system test users were asked to fill out a questionnaire wherein they rated the system on their overall satisfaction with the recommender and the system as a whole; the usefulness, efficiency and intuitiveness of the visualisations and the interaction, how easy the visualisation and the interactions were to understand, and whether they learned something new about relations between the subreddits by using our system. Our qualitative evaluation shows that users are overall satisfied with the system as a whole and the usability of the system with regards to understanding the interactions and the produced visualisations. However, they have noted that the quality of the recommendations varied and that sometimes it was not very clear the purposes of the visualisations and therefore could not clearly understand the information presented.

For future research, the recommender can potentially by improved by not using the sentiment values for the recommendations but using the cluster proximities instead. As became apparent in section 4, sometimes the recommendations for subreddits in more general clusters seemed a bit off, whereas their neighbouring points actually showed more similar subreddits. Moreover, in terms of the visualisations and the system as a whole, future research could elucidate whether changes to the flow of the system could improve the user experience, by making certain parts of the system easier to understand. In addition to helping the general population with recommendations for subreddits, the tool may also assist researchers in, for instance, the social sciences, to uncover patterns in the subreddit data. This may lead to interesting insights into (online) human culture.

To improve the system as a whole, further research can be done to creating visualisations that handles the full extent of the data set. The use of the subset of the original data set was due to the graphs' limited abilities to provide timely and fluid experience for the users with big data sets. Research into making scalable visualisations will improve user experience as well as the insight to be gained, as it will allow for users to search a wider range of different subreddits and incorporate more data for the recommender algorithm.

Ultimately, we have seen that recommendations for subreddits can be made based on sentiment relations between different subreddits. Additionally, by using t-SNE and clustering with k-means, we were able to make a visualisation that aids people in their exploration of subreddit data and their similarities. Onwards to a world where visualisations increase the number of time spent on Reddit.

References

[1] R. S. Alsabti, K. and V. Singh. An efficient k-means clustering algorithm. *Electrical Engineering and Computer Science*, 42, 1997.

[2] V. Belák, S. Lam, and C. Hayes. *ICWSM 2012*, 2012.

[3] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky. Community interaction and conflict on the web. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pp. 933–943. International World Wide Web Conferences Steering Committee, 2018.

[4] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky. Community interaction and conflict on the web. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pp. 933–943. International World Wide Web Conferences Steering Committee, 2018.

[5] S. Kumar, X. Zhang, and J. Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1269–1278. ACM, 2019.

[6] L. V. D. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning*, pp. 2579–2605, 2008.

[7] D. Sacha, A. Stoffel, F. Stoffel, B. Kwon, G. Ellis, and D. Keim. Knowledge generation model for visual analytics. 2014.

[8] J. Scholtz. Developing qualitative metrics for visual analytic environments. In *Proceedings of the 3rd BELIV'10 Workshop: BEyond time and errors: novel evaLuation methods for Information Visualization*, pp. 1–7, 2010.

[9] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. van Wijk, J. Fekete, and D. Fellner. Visual analysis of large graphs: State-of-the-art and future research challenges. *COMPUTER GRAPHICS forum*, 30:1719—-1749, 2011.

A ANSWERED QUESTIONNAIRES

Here we display the actual answeres the people gave who tested our recommender. Table 2 are the ratings given per question.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
P1	2	3	3.5	3	1	3	4
P2	2	3	3	2	1	3	4
P3	3	4	3	4	4	2	4
P4	3	4	2	4	2	2	4
P5	2	4	2	3	1	4	4.5
P6	4	4	2	4	4	3	5
P7	4	4	3	3	5	3	4
P8	2	3	4	3	3	4	3
P9	3	3	2	2	4	3	3
P10	2.5	3.5	3	3	4	3	4
P11	4	5	5	5	5	5	4
P12	3	5	5	4	5	4	4
P13	3	4	5	4	3	5	5
P14	4	4	3	4	4	3	4
P15	4	5	5	4	4	5	5

Table 2: Answers for the questionnaire

In addition to the scores, we have asked to give a comment, if they wanted, about the recommender system, we will list the comments here below:

- Some of the recommendations seemed quite spot-on, sometimes the recommender recommended the same subreddits as was input, and sometimes the connection between the recommendations was not always logical. However, it shows promise and I'd like to play with it in the future.
- Flow felt a bit weird, first going down to the circle thing and then again up to submit. Space was perhaps not used super effectively, as there was a lot of scrolling involved. Recommendations were a bit all over the place.
- Visualisations for these kinds of things are hard to do. I thought the legend could be better in that sense to explain what the different markers are. Also one of the markers had a color that was also in the visualisation, which was a little confusing. For higher scores of most recommendations do really feel like they have more in common.
- Sunburst graph is a bit difficult to navigate and scrolling e.g. for recommendations (which is the main component) feels a bit cumbersome. *Added later: recommendations were quite decent mostly
- There's maybe too many subreddits in the sunburst graph, but the idea is nice. It is easier to navigate the general graph though. Also, being able to see the legend right away would be helpful. Now I didn't immediately know what everything meant.
- Quality of recommendations varied. For my most visited subreddits they were way off, for some other ones they were okay. Also, I required some further explanation for what the edges meant because that wasn't clear to me right away
- Needed some extra information on the edges, and what the scores meant, so it wasn't quite intuitive to understand what every visualisation and interaction meant. However, it was very interesting to see the clusters on the graph and the spread of the edges being more dense in their corresponding cluster.
- Sometimes recommendations seemed quite random e.g. for some artists and other times they would be pretty decent. The list of input fields dropping down was a bit inconvenient especially for larger numbers of inputs. It was really nice to see the clusters though and explore that a little bit.
- It's beautiful! Slightly larger font sizes and reduced scrolling (so that you could see the recommendations without scrolling) might be ways to improve.
- Would be nice to see the recommendations shown together with the graph in one screen.
- pros: Interesting visualization. Very cool to see a global overview of how subreddits are related to each other and to what category each subreddit belongs. I'm impressed. That wheel thing was intuitive to understand. cons: 1) There is A LOT of information thrown at you when first opening the application. It took me a few minutes to figure out what things even mean and how they are used. 2) Most recommendations were related, but some were not.
- - would be cool if it could take the date of subreddit into account. - maybe add an example subreddit so people see the idea.