**TECHNICAL REPORT**

**Natural Language Analysis of Build a Better Grinnell Survey Data**

Shreyas Agrawal, Jiayi Chen, David Dai, Hong Doan

Spring 2023

# **I. Introduction**

“Build a Better Grinnell 2030 Community" is a visioning project with the goal to provide clarity on community strengths and areas that need improvements prioritized by the Grinnell community (Build a Better Grinnell 2030 Visioning Project). At the current phase of the project, a general survey has been given to the entire Grinnell community, with specific actions taken to intentionally target certain populations within the broader community. The survey consisted of a series of open-ended questions aimed at identifying strengths, assets, and values, as well as needs and visions. From the responses of this survey, our client wants to explore the needs identified by the community as well as differences in responses across self-described identities and affinity groups.

The research team of this project has given us two main datasets. The first dataset consists of 12 survey questions and 542 respondents, with each row representing a respondent and each column corresponding to a survey question. The questions are divided into two categories seeking responses about strengths and needs of the Grinnell community. The responses are provided in natural language, which requires the application of text analysis techniques to draw meaningful conclusions and insights from the data. The second dataset is a Table of Frequency Counts provided by the clients which is consist of the main Category and the major Issues the clients identified. The table aslo includes the percent of total responses with the total count from town and Grinnell College student. Manual summary of distinct responses and Grinnell College student responses are also included.

The main goal of this project was to provide the client tools and visualizations to identify primary needs and background of respondents by performing natural language analysis with the given textual data. The analysis process involved performing exploratory analysis techniques to identify common themes and issues, creating clusters of respondents with common needs or background, and efficiently communicating the results with our clients through visualizations and summaries. By applying these techniques, we can provide the clients helpful insights into the community's priorities and concerns to help Build a Better Grinnell focus their efforts on the most pressing areas for improvement.

# **II. Methods**

1. **Data cleaning**

The supervised data cleaning was conducted in Excel manually. The final dataset consists of “Form Response 1” and “GC Students” tab from the original BABG dataset.

Based on our discussion with the client, questions were divided into two groups listed below (Table 1.).

| Strength Questions | 1. What are things that make you glad to live in Grinnell? 2. What things have the greatest positive impact on your quality of life in Grinnell? 3. Think about a time when you felt particularly connected to the community or proud to live in Grinnell. Tell us about it. 4. Do you think the community has a set of core values (e.g. what is important, what we believe, principles guiding our behavior)? What do you think they are? |
| --- | --- |
| Needs Questions | 1. If you have considered moving from Grinnell, what are reasons you decided to stay? 2. If you do not currently live here, what changes in the community might cause you to move to Grinnell? 3. Have there been things that have made you consider leaving Grinnell? What were they? 4. What things have frustrated you about living in Grinnell? 5. What things do you think Grinnell is missing – that would benefit you personally if it had? 6. If funds were unlimited, what changes would you make to improve Grinnell? 7. If you do not currently live here, what changes in the community would make you more likely to move to Grinnell? |

*Table 1. Question groups in the survey*

1. **Processing textual data**

Our analysis was performed using Python programming language written in Google Collab notebooks. pandas library was imported to read data from input Excel files into dataframes that are easy to manipulate using built-in functions. We also used this library along with numpy throughout our project for general data manipulation and analysis in Python.

In order to understand our data and do initial data exploration, the Natural Language Toolkit (NTLK) was deployed to support the transformation of the textual survey data into word tokens suitable for statistical analysis (Bird, Klein & Loper 2009). A string of input text was split into individual words, and punctuations were removed using the word\_tokenize() function. After that, words that do not provide useful information such as conjunctions or articles were filtered out if they were listed in stopwords collection from NTLK. Spellcheck was conducted by removing tokens that did not exist in the words collection, which contained all meaningful English words. WordNetLemmatizer() was used to assign various forms of a word the same token based on their root. So for example, communities and community were merged. After this process, the strings of textual data were converted into lists of word tokens, as well as numerical data, to prepare our data for further analysis..

The number of occurrences of each word token in the records was counted using a Counter() object. Counter allowed us to make a set to count the number of words. This was advantageous because we were able to remove repeated word counts in a particular response. For example, if a respondent used the word “town” twice in one response to a particular question, then it would only be counted once, due to the property of sets not allowing for repeats. For visualizations, using matplotlib and seaborn library, bar charts were created to provide a more precise and quantitative view of word frequencies. Word clouds with the size of each word being proportional to its frequency were also made using the wordcloud library to provide easily interpretable representations of word frequency data.

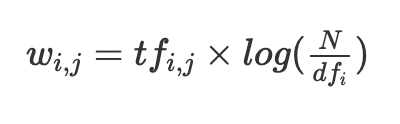
To provide a more comprehensive context for words that are used frequently, the words and phrases that come before and after them were identified. For each response used the word token of interest, this involved identifying the index of the token in the list of word tokens using the index() method, and words before and after that index were extracted and added to a DataFrame. The process was repeated for the whole subset of responses that used the token of interest, and the final DataFrame with the index of response as well as two words before and two words after the token of interest was exported to an Excel file.

1. **K-mean clustering method**

Patterns and themes in each subset of responses were identified by analyzing the clusters made using k-mean clustering method with the tools in scikit-learn machine learning library (Pedregosa et al 2011).

K-means is a machine learning clustering algorithm that groups data points based on their distance from a cluster center. The algorithm works by randomly assigning each data point to a cluster. Then, the cluster centers are updated to be the average of all the data points in each cluster. This process is repeated until the cluster centers no longer change, and K clusters were created.5 There are several methods for choosing the optimal number of clusters. In our project, we explored the elbow method, which is a heuristic approach that identifies the optimal number of clusters as the point at which the within-cluster sum of squares (WSS) curve starts to plateau (Miller 2023).

The lists of word tokens were transformed into a numerical matrix with Term Frequency Inverse Document Frequency (TF-IDF) features, which means that each cell represented a comparison value of the number of times a word appears in a response with the number of responses the word appears in (Equation 1.). This computation was done using TfidfVectorizer()method



*Equation 1. Formula to compute TF-IDF score for word token i in response j. represents the number of occurrences of i in j. is the number of responses containing i. N is the total number of responses*

Once our textual data was converted to numbers, we proceeded with our clustering method. For our distance metric, we used cosine\_similarity() method, which computes similarity as the normalized dot product of 2 vectors (Pedregosa et al 2011). By calculating the cosine similarity between text vectors, we can measure their similarity in terms of direction and disregard their magnitude, enabling effective clustering of similar textual responses based on their semantic similarities. After each response vector was generated, it was passed in as an input for KMean.fit() method. The clustering output would then be stored in a dictionary, in which the number labels of the clusters are the keys bound to a list of indices for the responses in that cluster. Responses in each cluster were also combined into a string, which was then transformed into a list of tokens using the processing method described above.

This whole process has been implemented in the clustering() function that we defined in order to easily repeat clustering on a different input. One challenge with clustering on textual data is that we will get which responses belong to which cluster, but we don’t have a meaningful label for that cluster. So, we took a supervised approach here and determined a label for each cluster based on the most commonly used tokens (words) in that cluster of respondents and a brief manual review of the responses in that cluster.

1. **Supervised Subsetting**

Supervised subsetting is based on the Table of Frequency Counts provided by the clients. This table highlighted some issue categories that were extracted from the survey responses. In order to subset respondents that belong to a particular category, we first created keywords lists manually for each category based on the issues the clients identified in the distinct responses and GC responses in the Table of Frequency Counts, mostly using synonyms for the issues. Using pandas library and its methods such as "iterrows()" and "iloc[]”, we filtered out rows that use particular keywords and made subsets of respondents. Further analysis was done on respondents of each category to better understand what they said in the strengths questions (Table 1). For this we made word frequency plots, word clouds, looked at words before and after, conducted sentiment analysis on each category of respondents, and clustering analysis to cluster respondents using methods described previously.

1. **Sentimental Analysis**

We were interested in finding out if respondents are answering positively or negatively to a question, and analyze whether each answer is positive, neutral, and negative. Sentiment analysis is a valuable technique used to determine the emotional tone behind a given text. By analyzing the sentiment, we can understand whether the expressed sentiment is positive, negative, or neutral. One way to perform sentiment analysis is by using the transformers library. We created a sentiment analysis pipeline by using the "pipeline" function from the transformers library with the parameter "sentiment-analysis". This pipeline was then used to analyze the sentiment of different texts by providing them as input. We conduct sentiment analysis on each respondent to see how they are responding throughout the survey, and we also did this for each question to quantify how many people have responded positively, negatively, and neutrally to this question.

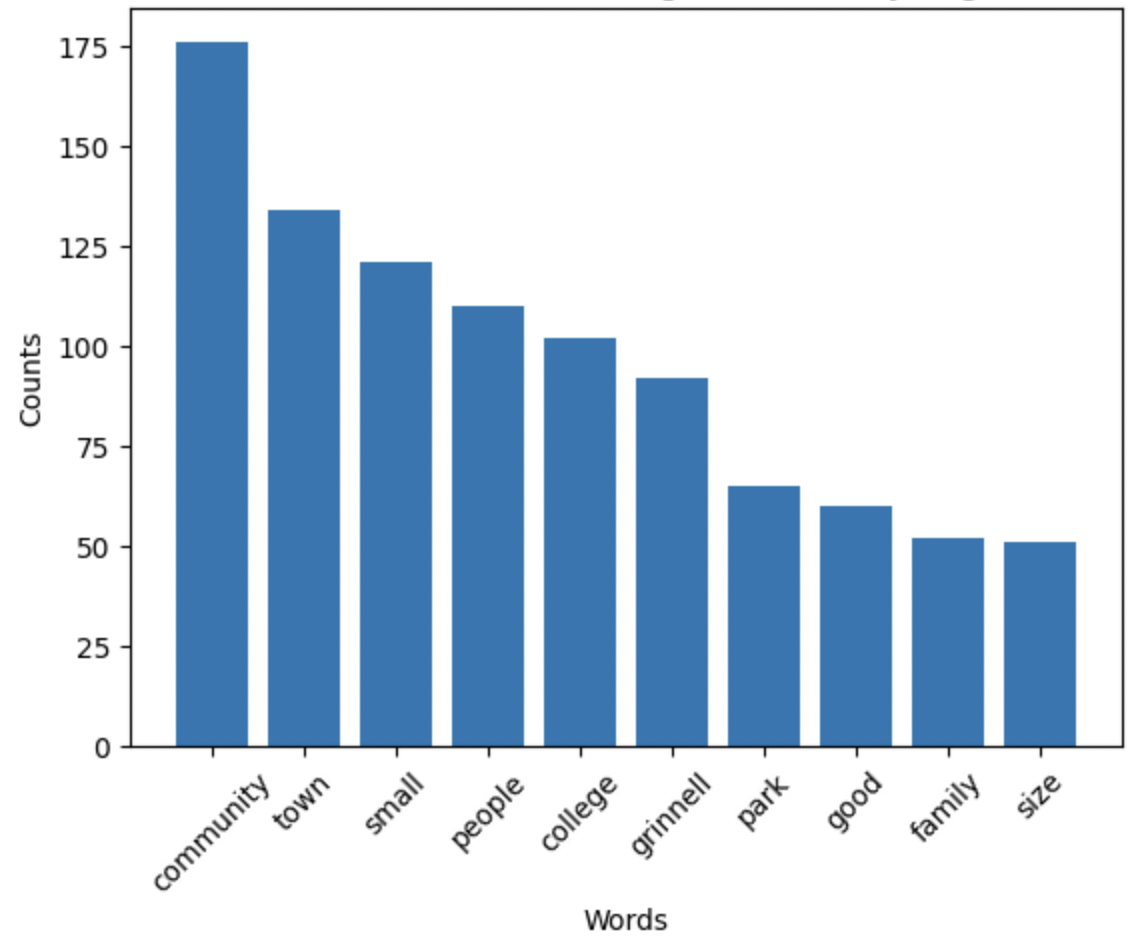
# **III. Results**

Although all the analyses discussed have been repeated over different questions and issues, due to the length constraints of this report, we have only included example results.

1. **Analysis of the occurrences of words for responses in one question**

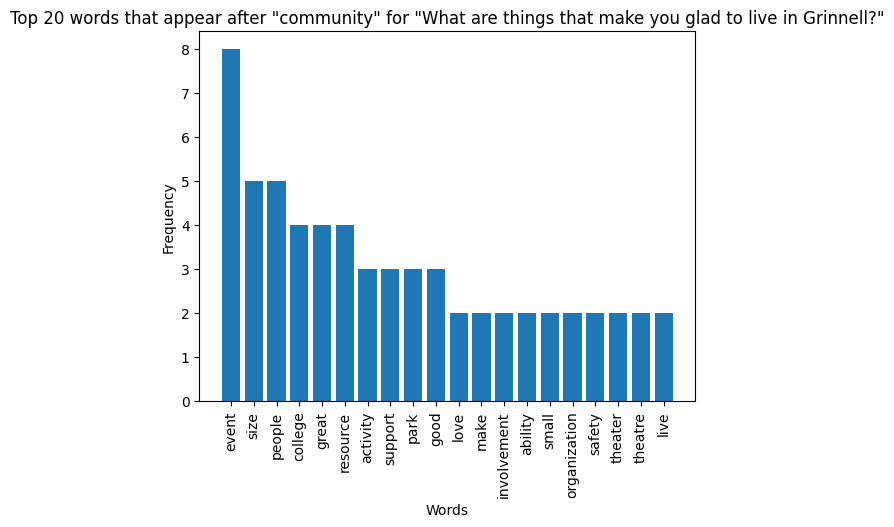
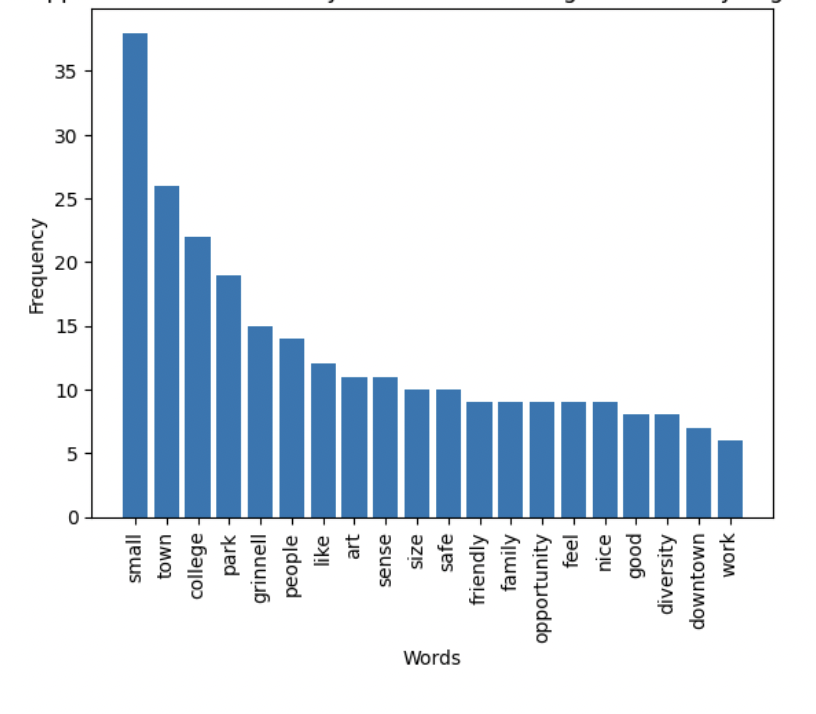
After reading the survey data into a dataframe, all responses for the first question - “What are things that make you glad to live in Grinnell?” - were extracted. After the process of tokenizing, removing stop words, spell-checking, and lemmatizing, the five most frequently used words used in the responses for this question were “community”, “town”, “small”, “people”, and“grinnell”. (Figure 1. & 2.)

*Figure 1. Word cloud for most frequently used tokens in responses for question 1 - “What are things that make you glad to live in Grinnell?”*

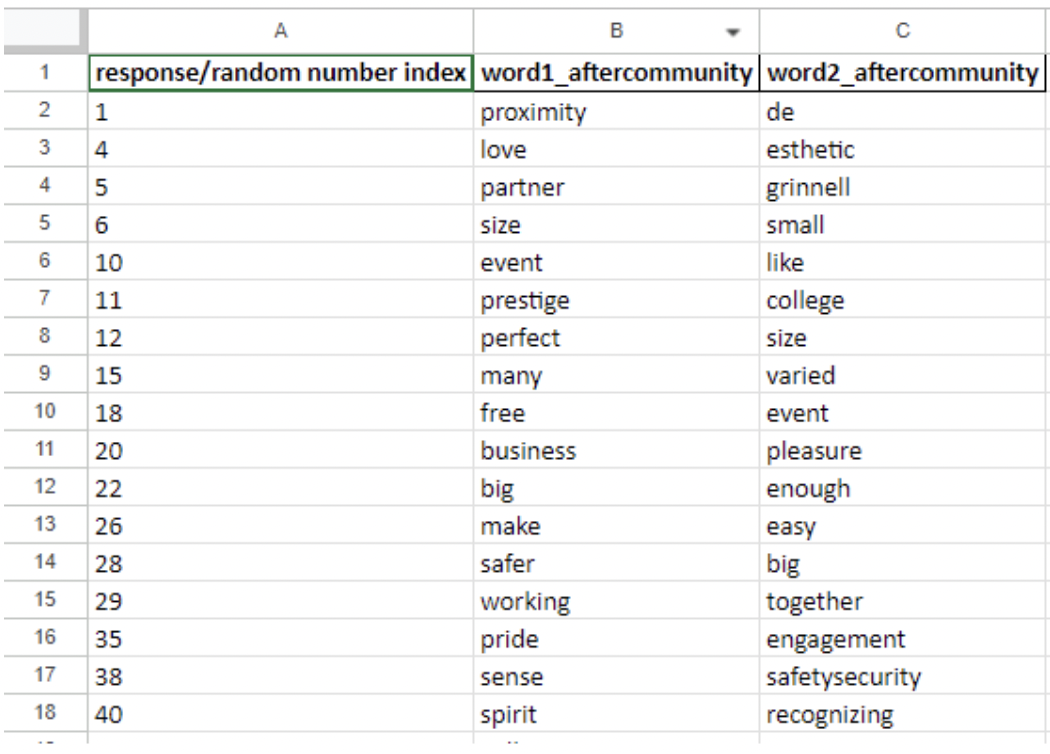


*Figure 2. Bar chart for most frequently used tokens in responses for question 1 - “What are things that make you glad to live in Grinnell?”*

As the word "community" had the highest frequency, the words that appeared before and after it were identified (Figure 4.) and their occurrences were counted (Figure 3.) .

**A** **B**

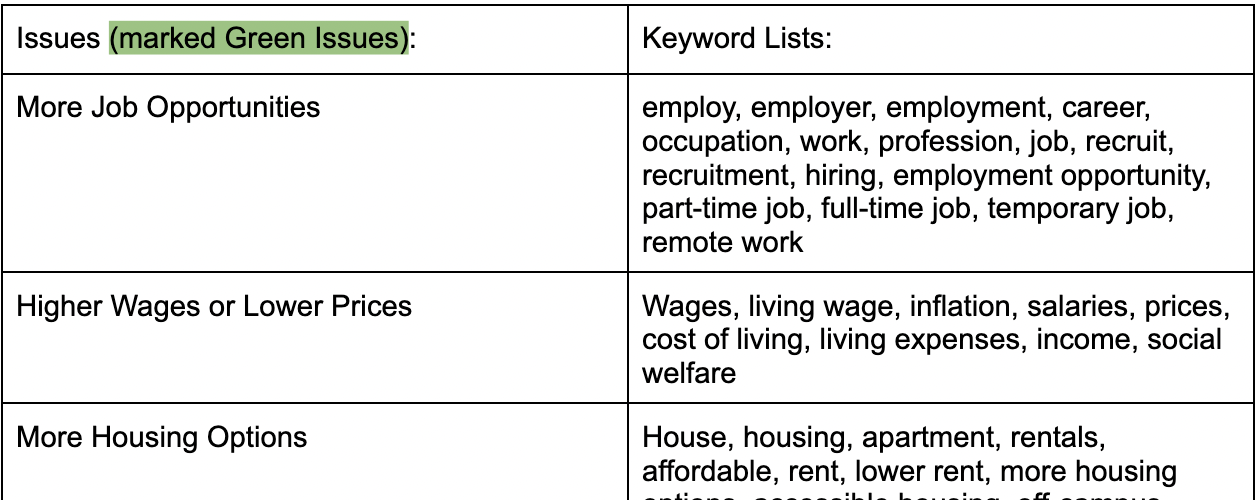
*Figure 3. Bar chart for most frequently used tokens (A) after and (B) the token “community” in responses for question 1 - “What are things that make you glad to live in Grinnell?”*



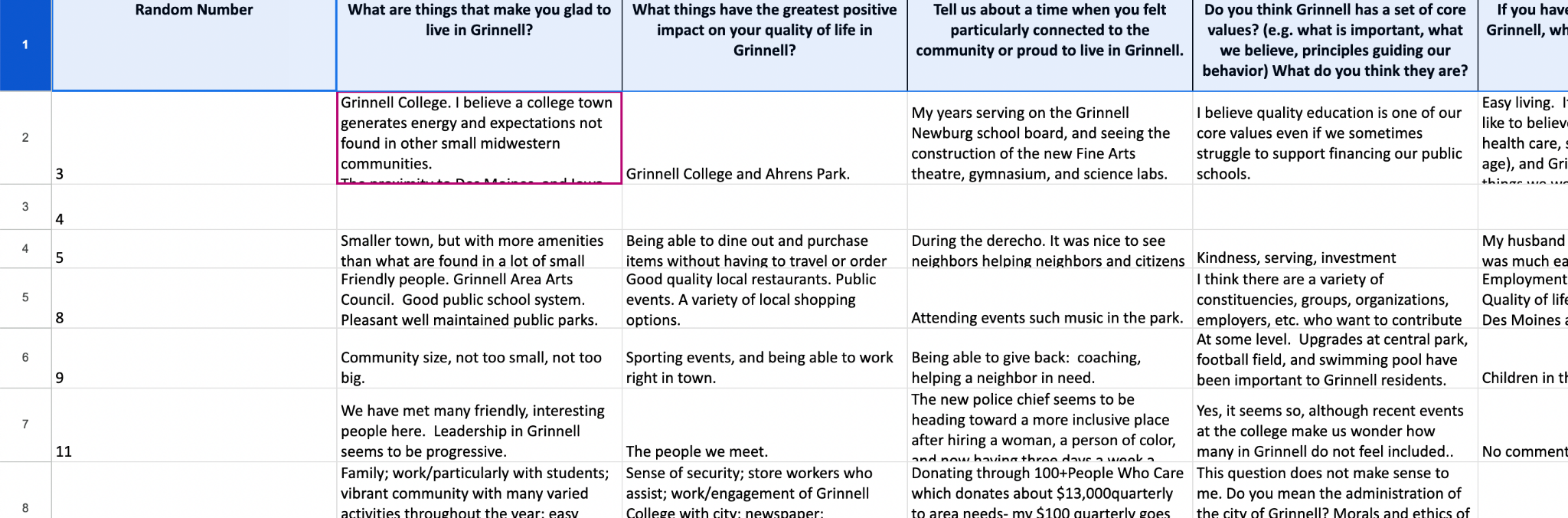
*Figure 4. Snapshot of the Excel sheet for words appeared after “community”in responses for question 1 - “What are things that make you glad to live in Grinnell?”*

1. **Analysis for each issues identified by the client**

After reviewing the Table of Frequency Counts provided by the clients, lists of keywords were generated for each prioritized issue (Figure 5.). Excel sheets were exported for responders who mentioned the identified keywords of each issue in their responses to questions about need (Figure 6.).

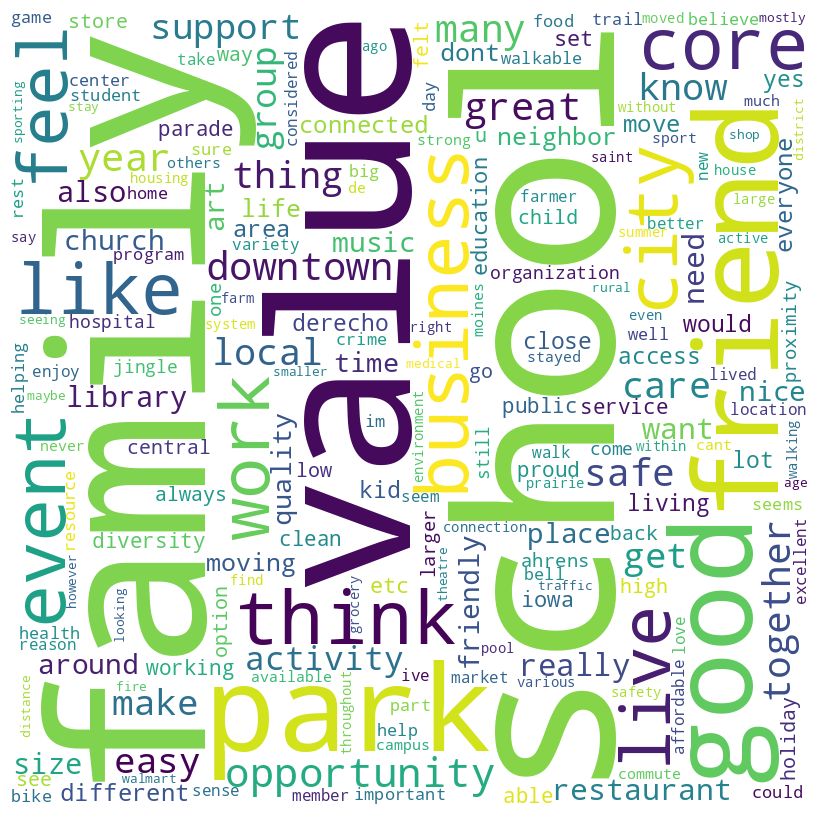
****

*Figure 5. Snapshot of the Supervised keyword lists with 35 prioritized issues clients identified*

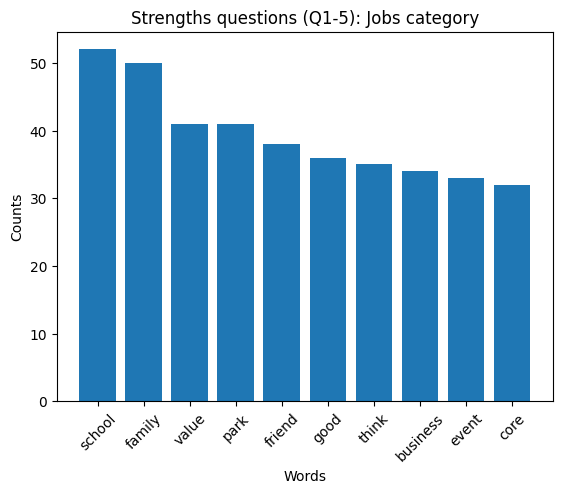
****

*Figure 6. Snapshot of the Excel sheet for the subset of respondents who mentioned keywords about More Jobs Opportunities in their responses about needs.*

A brief background of the respondents was analyzed by examining the frequency of words used in questions about the strength of the town of Grinnell. In order to determine the uniqueness of the subset, we removed words with the highest overall occurrence counts in our visualization (Figure 7).



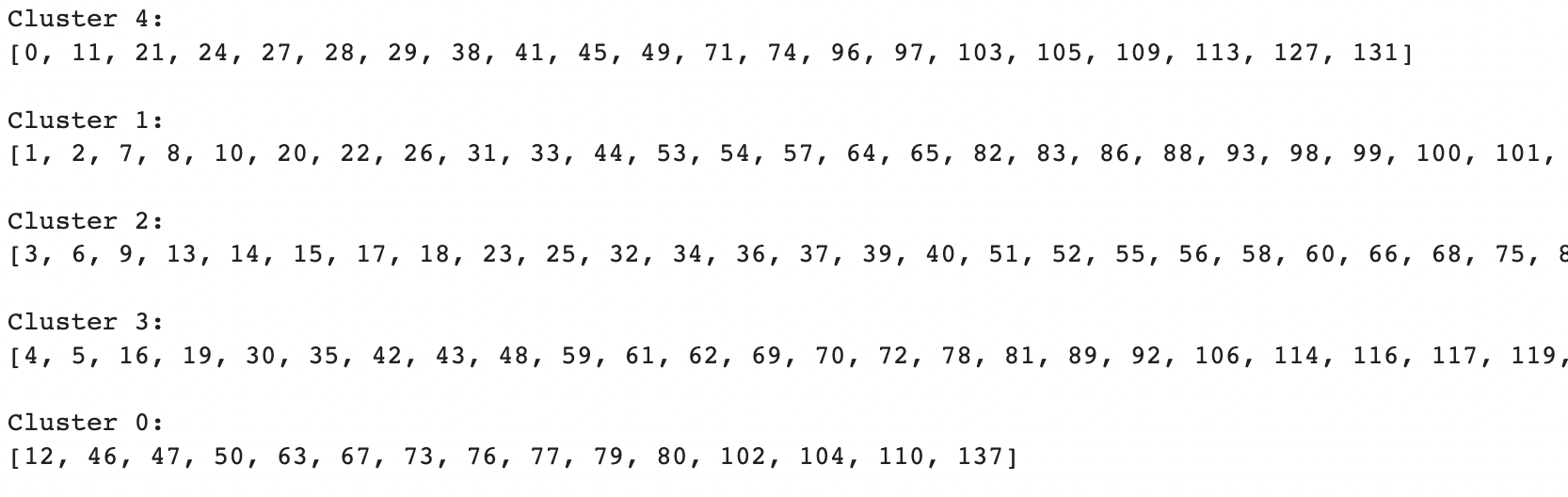
*Figure 7. Word cloud for most frequently used words by respondered who mentioned More Job Opportunities as their needs when answering to questions about the strength of Grinnell (Q1-5). 'community', 'town', 'small', 'people', 'college', 'grinnell' and 'job' were removed from the list of tokens*

****

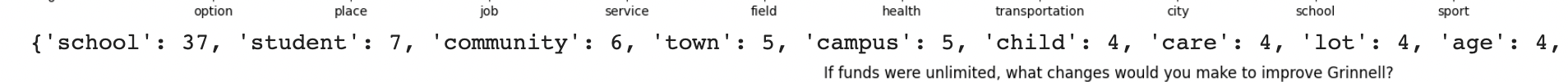
*Figure 8. Bar chart for most frequently used words by respondered who mentioned More Job Opportunities as their needs when answering to questions about the strength of Grinnell (Q1-5). 'community', 'town', 'small', 'people', 'college', 'grinnell' and 'job' were removed from the list of tokens*

Using the clustering() function that we defined, a dictionary with a list of responder’s id in each cluster was generated (Figure 9.). Although the optimal K value identified by this method was around 15, our team discussed with our advisor and decided to proceed with a lower value of 5 in order to have more distinctive clusters due to the overlapping of the clusters created.

For easy visualization, we generated pie charts with (Figure 10. & 11.)

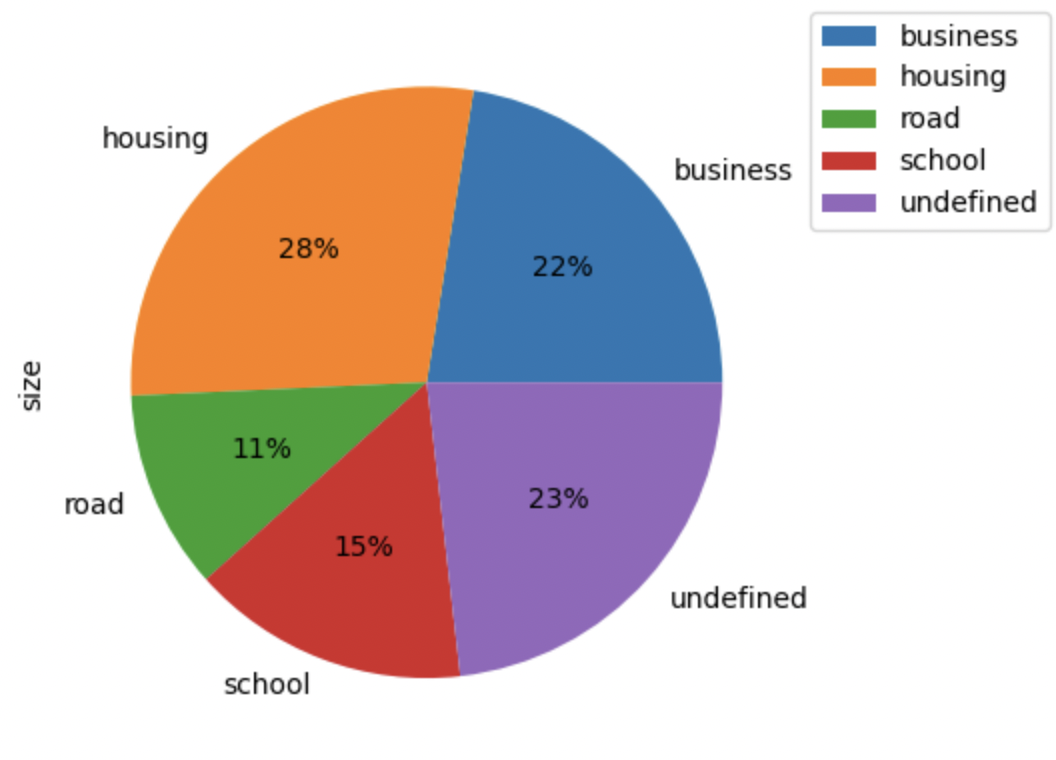
****

*Figure 9. Snapshot displaying the clusters dictionary showing respondents that belong to a cluster for More Job Opportunities using question 10 - “If funds were unlimited, what changes would you make to improve Grinnell?”*

**

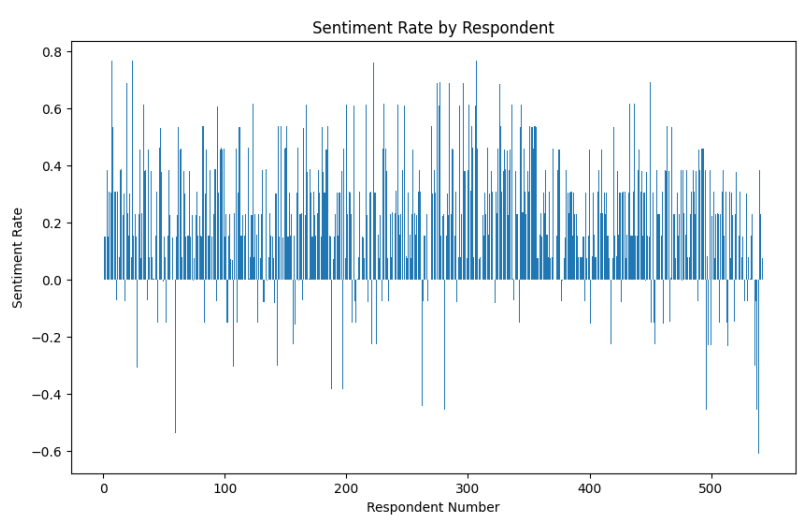
*Figure 10. Snapshot of word counts for cluster 5 in the subset of More Job Opportunities using question 10 - “If funds were unlimited, what changes would you make to improve Grinnell?”.*

*Final label for this cluster: “school”*

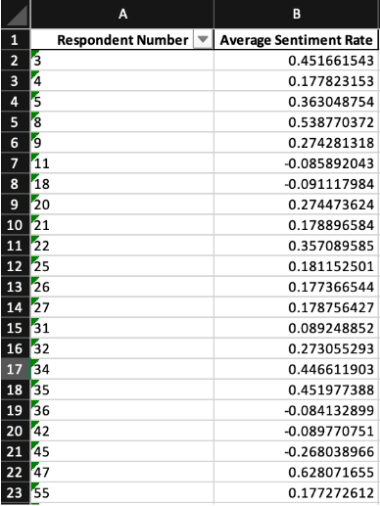


*Figure 11. Pie chart representing the size of each cluster in the subset of More Job Opportunities using question 10 - “If funds were unlimited, what changes would you make to improve Grinnell?”.*

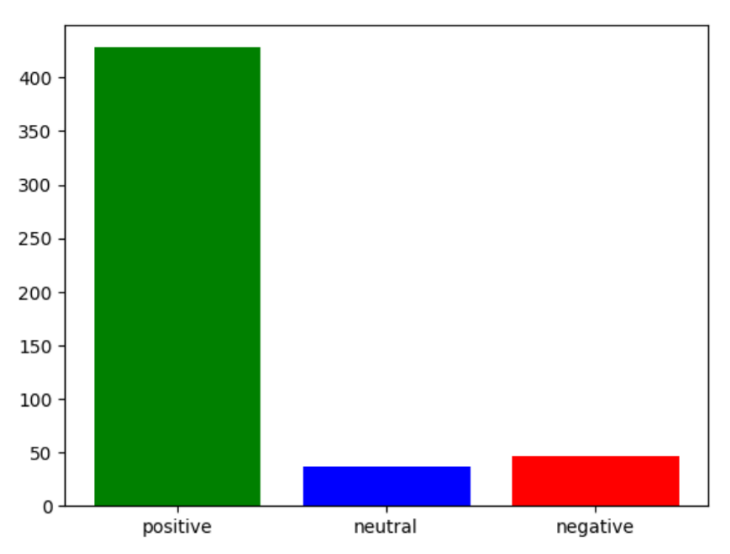
1. **Sentimental analysis for each individual question and the overall survey**

****

*Figure 12. Average Sentiment Rate Across Questions of each Respondent.*

****

*Figure 13. Snapshot of Average Sentiment Rate Across Questions for each Jobs Category Respondent*

**

*Figure 13. Sentiment Count for respondents Bar Chart for Question 1*

**IV. Discussion**

Our analysis contributes to the phase of the Build a Better Grinnell 2030 project where a comprehensive survey has been conducted, reaching out to Grinnell residents. We tailored our analysis to the client's primary interest of identifying community’s needs, as well as exploring the variance in responses across identities and affinity groups within the community.

We explored multiple other alternative clustering methods in this project alongside K-means clustering. However, these methods either lacked effectiveness in generating meaningful distinct clusters or the outputs and visualizations we could make from it were not considered useful by the clients. The two alternative clustering methods we experimented with were PAM (Partitioning Around Medoids) clustering and hierarchical clustering.

For the PAM clustering, we aimed to leverage its advantages, such as robustness to outliers and the ability to provide interpretable cluster representatives. However, due to the nature of the open-ended question survey and the corresponding textual dataset, PAM clustering did not yield useful and distinct medoids to serve as cluster centers. Nevertheless, for clients with relatively small and quantitative datasets in the future, we recommend considering PAM analysis for improved clustering results.

We were still able to create subcategories of respondents using our K-means clustering algorithm which showed to be effective, and the outputs were also useful to the client, so we added that as part of our deliverable. In addition to standard clustering algorithms, we used some semi-supervised approach to creating categories and groups as well. An example of this was using a keywords list to identify groups of people who use certain keywords in their responses and creating a category out of that. Another example was using stacked matrix sentimental analysis to see which respondents answer positively to some questions, and negatively to others.

As for stacked matrix sentimental analysis, our goal was to offer clients a sentiment-based overview and enhance their understanding of the sentimental relationships between responses from pairs of survey questions. For instance, we categorized survey questions as positive, neutral, or negative, resulting in a 3x3 matrix where each cell represents the frequency of occurrences in different categories. The categories are labeled as Positive, Neutral, and Negative, corresponding to the responses for two questions in the survey. We believed these matrices could be valuable for further exploration, such as identifying negative responses to positive-oriented questions or providing a grouping method. However, the clients later indicated that these matrices were not aligned with their analysis priorities and might not be useful for their needs. However, sentiment analysis for the categories we created using keywords was a useful output for the client, hence average sentiment for each issue category was part of the deliverable.

We also must acknowledge the limitations of the algorithm and the results of the sentiment analysis. For instance, if a response contains a strongly negative word like "disaster," the algorithm may categorize the entire response as negative, even if the overall context is positive. This is because the presence of the highly negative word outweighs the positive context of the response.

Our project will contribute towards identifying common issues that the survey respondents have identified in the community, how people are responding to the survey in terms of their sentiment, and investigating responses of people belonging to particular categories.

**References**

Build a Better Grinnell 2030 Visioning Project. Build A Better Grinnell. (n.d.). http://www.buildabettergrinnell.org/

Bird, S., Klein, E., & Loper, E. (2009). Natural language processing with Python: analyzing text with the natural language toolkit. Reilly Media, Inc.

Education Ecosystem (LEDU). (2018, September 12). Understanding K-means clustering in machine learning. Medium. https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1

Miller, R. E. (2023). Introduction to Partitions Clustering. STA-230 - Introduction to Data Science (Spring 2023). https://remiller1450.github.io/s230s23/clustering2.html

Pedregosa, Fabian and Varoquaux, Gabriel and Gramfort, Alexandre and Michel, Vincent and Thirion, Bertrand and Grisel, Olivier and Blondel, Mathieu and Prettenhofer, Peter and Weiss, Ron and Dubourg, Vincent and others. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825–2830.

**Acknowledgement**

Thank you, Professor Ryan Miller, for your guidance and support. Thank you, takeover team, for providing us with the preliminary sentimental analysis code.