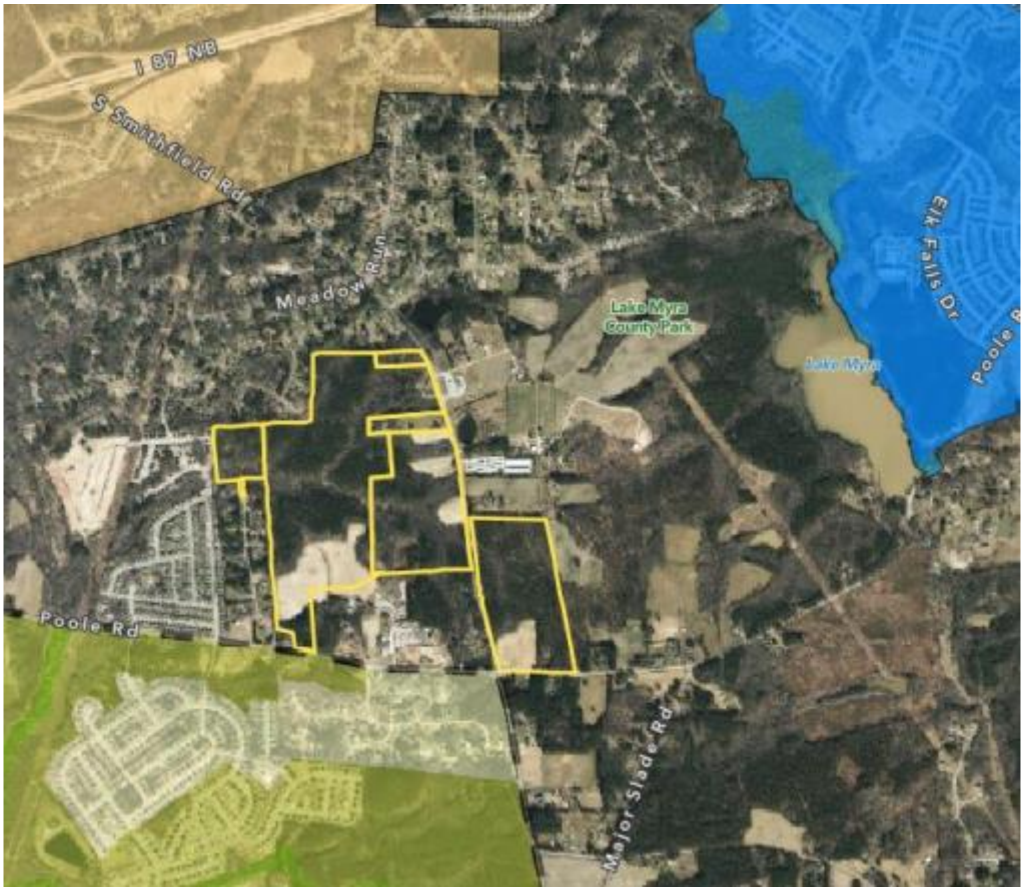
Text Mining Project on Submitted Public Comments



ZMA-15-22 Proposed Parkside at Westlake

How Do Residents Feel About it?

ABSTRACT

Using advanced text analysis software, an exploratory review is conducted on public comments submitted to the Town of Knightdale on the proposed ZMA-15-22 Parkside at Westlake Neighborhood Mixed-Use Development.

By Tabitha Hagen

UNCW's MIS 506 Text and Unstructured Data course

Photo borrowed from the official report from the Town of Knightdale's Public Hearing

Created on April 23, 2023

From the Joint Public Hearing April 20, 2023

Problem Statement

The very densely proposed Neighborhood Mixed-Use (NMX) development has attracted the attention of many residents and this report will explore the main themes of their public comments and assist the Knightdale Town leaders in their decisions moving forward. What was found…. What was not found…. What questions were answered?

As a member of the Town of Knightdale's Land Use Review Board (LURB), individuals are appointed to serve "as the Town's Planning Board per North Carolina General Statute 160A-361"[1]. Members serve as the Community Appearance Commission, the Tree Board, have monthly LURB meetings and combine with the Knightdale Town Council for a monthly Joint Public Hearing. The data used in this report comes directly from data submitted by residents to the Town of Knightdale for the Joint Public Hearing on Thursday, April 20, 2023, and the only item on the official Agenda was on ZMA-15-22 Parkside at Westlake[2].

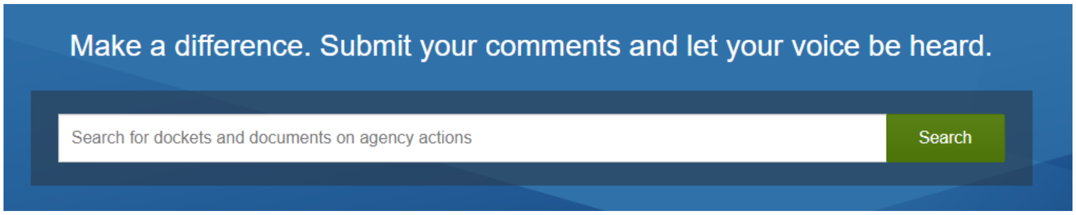
Objectives:

1. Look at the words that appear the most on the using the R Programming Language in RStudio.
1. Measure how important a word is in the collection of public comments using the online Orange Data Mining Tool.
2. Look at pairs of words or words used together using RStudio's bigram visualizations and Network Diagrams.
3. Infer which topics are most prominent in the comments using Topic Modeling using the online Orange Data Mining Tool.

[1] https://www.knightdalenc.gov/government/advisory-boards/land-use-review-board
[2] https://pub-knightdalenc.escribemeetings.com/Meeting.aspx?Id=a6f18fe9-ef20-4676-8189-a9f9ac070478&Agenda=Agenda&lang=English

Literature Review

The George Washington Regulatory Studies Center notes that in the past, public comments were either mailed via “snail-mail” or delivered in person in order to provide an interactive way for a citizen to be proactive[3]. In January 2003, the federal government even launched a website https://www.regulations.gov/ to “make it easier for the public to participate in the rulemaking process”[3]. The government website has become even more essential in the current times so that public users can keep up with federal regulatory materials. More importantly, as is prominent on the government website and seen in the picture below, is the desire for the public to easily contribute through official means, not necessarily through biased media. The George Washington Regulatory Studies Center goes on to point out that research on public opinion is a growing field that needs advanced text mining and analysis techniques because the amount of text data is too much for the human to adequately digest.



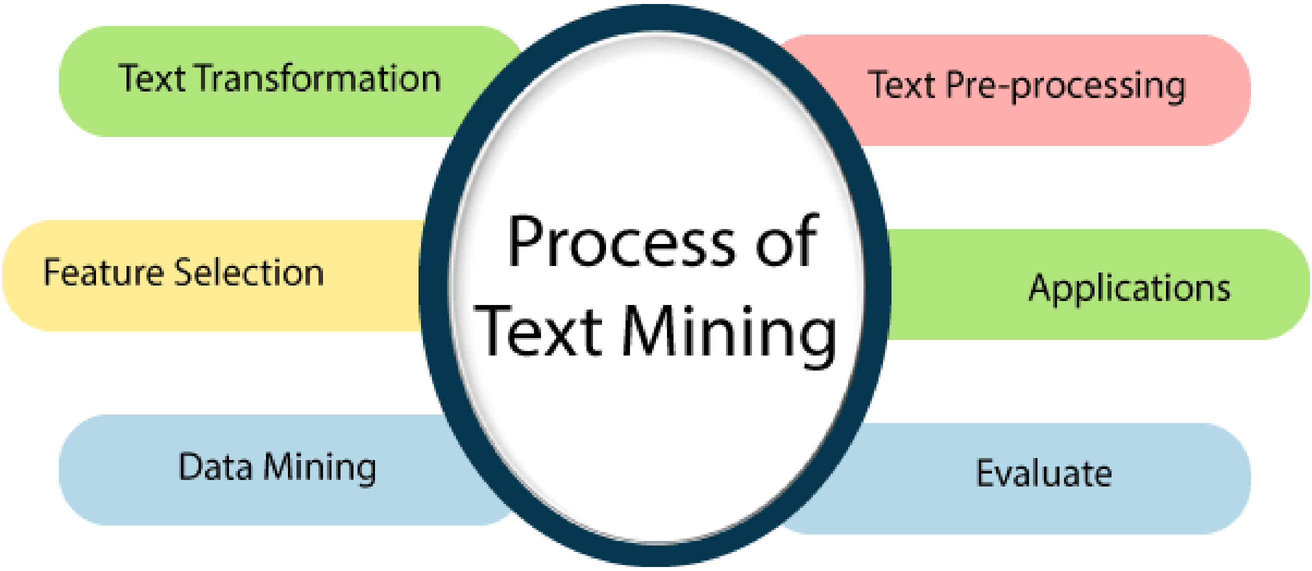
Make a difference. Submit your comments and let your voice be heard.

Since the 2020 Covid-19 Pandemic, leaders in both the private and public sector have had to reevaluate the value of public opinion and more specifically, how to obtain a good sense of public opinion on a particular topic. Prior to this time period, public comments were usually taken in by formal measures on paper and in person. However, when we needed the public to isolate for public health reasons, we began to do more interactions over the Internet. Public meetings were held virtually and many of our data collection was done on the internet to limit the amount of time in the same room with others. Donghong Wang and Jiliang Guo conclude that” the Chinese government integrates big data technology with government affairs in order to establish a smart government affairs platform, which aims to improve the enthusiasm of the masses to ask politics and enhance the timeliness and transparency of government platform answers”[4]. They also feel that “natural language processing technology surfaced to seek the hot issues that the public reflects” and their paper, “*The Big Data Analysis and Visualization of Mass Messages under ‘Smart Government Affairs’ Based on Text Mining*” compares methods for increased efficiency in this area. Essentially, text mining and advanced analytics allow any leaders, whether government, corporate, or otherwise, to be more informed as they face decision-making and gauging public input.

[3] https://regulatorystudies.columbian.gwu.edu/using-comments-data-research
[4] https://www.hindawi.com/journals/mpe/2022/8594233/

Literature Review Continued …

Founded on the basis set by educational partners in Russia who developed techniques for advanced analytics through machine language back in the mid-1990's[5], Megaputer Intelligence Inc. has been working to advance the field of data analytics with their PolyAnalyst products. In their "The Use of Text Mining to Analyze Public Input" paper, they say that citizen comments are highly unstructured and call for an exploratory analysis[6]. This exploratory investigation of unstructured data should include looking at keywords, word association, as well as classification. This report sets out to carry out this type of exploratory analysis.



Picture borrowed from https://www.javatpoint.com/text-data-mining

[5] https://www.megaputer.com/company/history/
[6] https://www.megaputer.com/wp-content/uploads/text-mining-analyzing-public-input.pdf

Data and Analysis

Data Set Description

The citizen input was collected and sent to both the Knightdale Town Board Members and the LURB just prior to the Joint Public Meeting on Thursday, April 20, 2023. The ZMA-15-22 Parkside at Westlake data was collected into a spreadsheet by the Knightdale Clerk and distributed to the Board members. Two last minute additions were added via email that afternoon and have been added to the end of the spreadsheet as well as dated with the email timestamp. When a resident chooses to submit their Public Comments, they do so by using a public form which states "All comments submitted will be shared with the Town Council and included in the official record of the meeting"[7]. When the minutes of the meeting are created and recorded, it includes the resident's name, address, and comments. The phone number and email addresses have already been redacted as those are for Town use not public use.

Looking at the rows or observations in this data set, we see:

- *There are 104 Public Comments taken from April 16, 2023, to April 20, 2023*

Looking at the Columns or variables in this data set, we see:

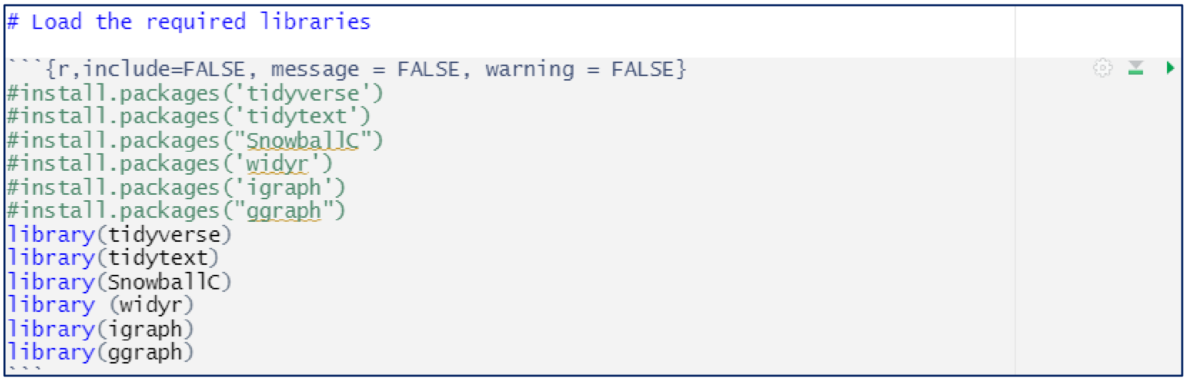
- *The original spreadsheet included 8 columns/variables, however, to align with how the Town Clerk publishes the official Minutes of the meeting, the email and phone number of the residents have been removed.*
- *The column/variable titles have had the space between the words replaced by an underscore (_) to facilitate easier text processing.*
- *The remaining column variable titles are:*
    - Date_Submitted
    - Name
    - Address
    - Public_Comment_Subject
    - Please_indicate_if_you_are_in_favor_in_opposition_or_do_not_have_a_stated_position_and_have_a_concern_or_neutral_statement
    - If_commenting_on_a_Public_Hearing_item_please_list_specific_reasons_why_you_are_in_favor_or_opposed_to_the_item

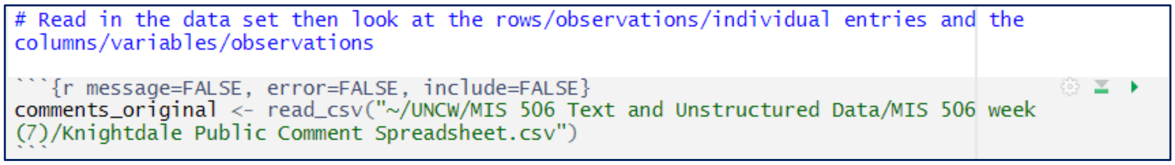
[7] https://www.cognitoforms.com/KnightdaleNC1/KnightdalePublicComment

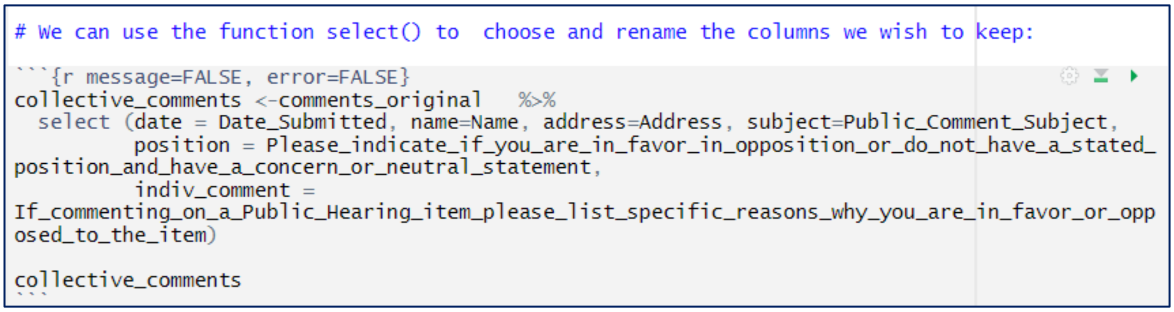
Data and Analysis Continued …

Data Analysis Process

1. **Setup** the needed libraries in R Studio



~~~
# Load the required libraries

```{r,include=FALSE, message = FALSE, warning = FALSE}
#install.packages('tidyverse')
#install.packages('tidytext')
#install.packages("SnowballC")
#install.packages('widyr')
#install.packages('igraph')
#install.packages("ggraph")
library(tidyverse)
library(tidytext)
library(SnowballC)
library (widyr)
library(igraph)
library(ggraph)
~~~
```

2. **Text Transformation** - Import the dataset into the RStudio program so that it can be utilized for further steps. This involves taking the data from the hard drive, Internet, or cloud storage and getting into a usable data structure which is called a *tibble* in R.

- *Import data using R's Tidyverse Package*



```
# Read in the data set then look at the rows/observations/individual entries and the
columns/variables/observations

```{r message=FALSE, error=FALSE, include=FALSE}
comments_original <- read_csv("~/UNCW/MIS 506 Text and Unstructured Data/MIS 506 week
(7)/Knightdale Public Comment Spreadsheet.csv")
```
```

3. **Feature Selection** – This occurs as we choose and rename the columns into useful variable names.



```
# We can use the function select() to  choose and rename the columns we wish to keep:

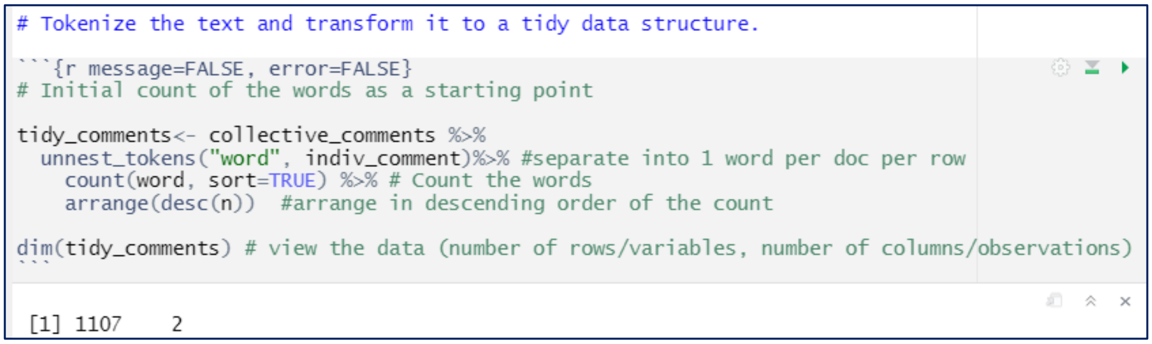
```{r message=FALSE, error=FALSE}
collective_comments <-comments_original   %>%
  select (date = Date_Submitted, name=Name, address=Address, subject=Public_Comment_Subject,
          position = Please_indicate_if_you_are_in_favor_in_opposition_or_do_not_have_a_stated_
position_and_have_a_concern_or_neutral_statement,
          indiv_comment =
If_commenting_on_a_Public_Hearing_item_please_list_specific_reasons_why_you_are_in_favor_or_opp
osed_to_the_item)

collective_comments
```
```
```

Data and Analysis Continued …

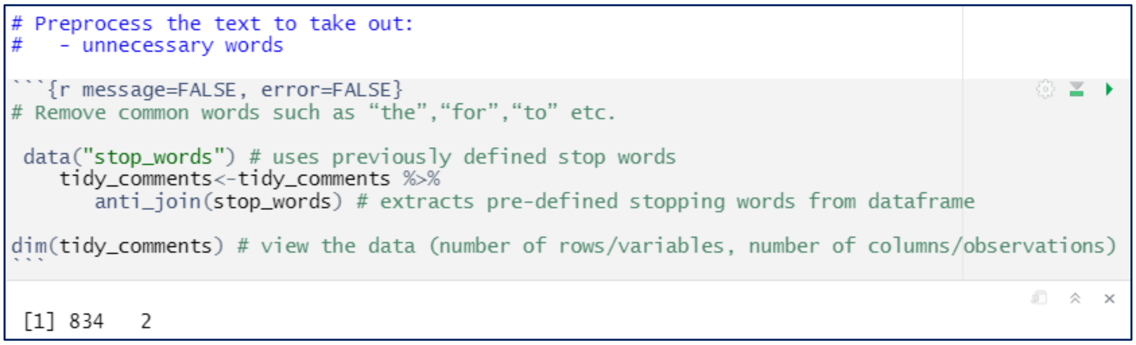
Data Analysis Process Continued …

4. **Text Preprocessing** – Using R's Tidytext package, the text is restructured which gives us a consistent way of storing data that makes analyzing the data easier.

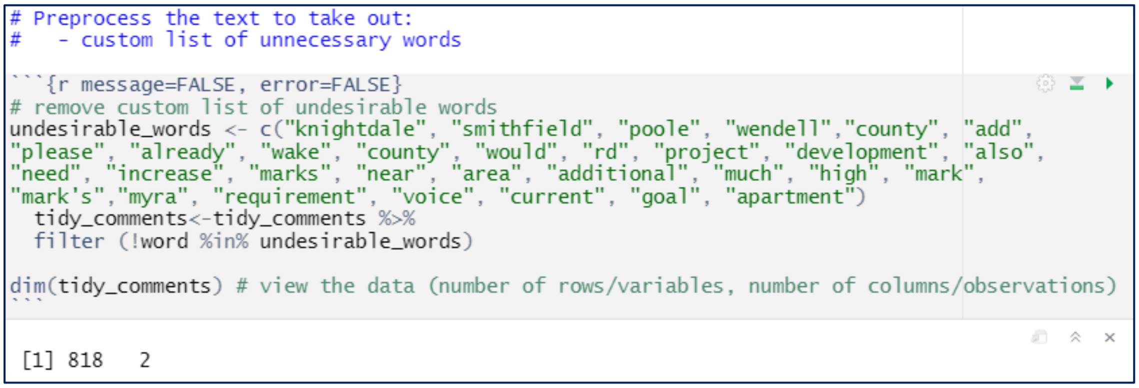
- *First Tokenization – **the data has 1107 words.***



~~~
# Tokenize the text and transform it to a tidy data structure.

```{r message=FALSE, error=FALSE}
# Initial count of the words as a starting point

tidy_comments<- collective_comments %>%
  unnest_tokens("word", indiv_comment)%>% #separate into 1 word per doc per row
    count(word, sort=TRUE) %>% # Count the words
    arrange(desc(n))  #arrange in descending order of the count

dim(tidy_comments) # view the data (number of rows/variables, number of columns/observations)
~~~

 [1] 1107   2
```

- *Default Undesirable words – **the data now has 834 words.***



```
Preprocess the text to take out:
- unnecessary words

```{r message=FALSE, error=FALSE}
# Remove common words such as "the","for","to" etc.

 data("stop_words") # uses previously defined stop words
    tidy_comments<-tidy_comments %>%
       anti_join(stop_words) # extracts pre-defined stopping words from dataframe

dim(tidy_comments) # view the data (number of rows/variables, number of columns/observations)
```

 [1] 834   2
```

- *Custom Undesirable Words– **the data now has 818 words.***



```
Preprocess the text to take out:
- custom list of unnecessary words

```{r message=FALSE, error=FALSE}
# remove custom list of undesirable words
undesirable_words <- c("knightdale", "smithfield", "poole", "wendell","county", "add",
"please", "already", "wake", "county", "would", "rd", "project", "development", "also",
"need", "increase", "marks", "near", "area", "additional", "much", "high", "mark",
"mark's","myra", "requirement", "voice", "current", "goal", "apartment")
  tidy_comments<-tidy_comments %>%
  filter (!word %in% undesirable_words)

dim(tidy_comments) # view the data (number of rows/variables, number of columns/observations)
```

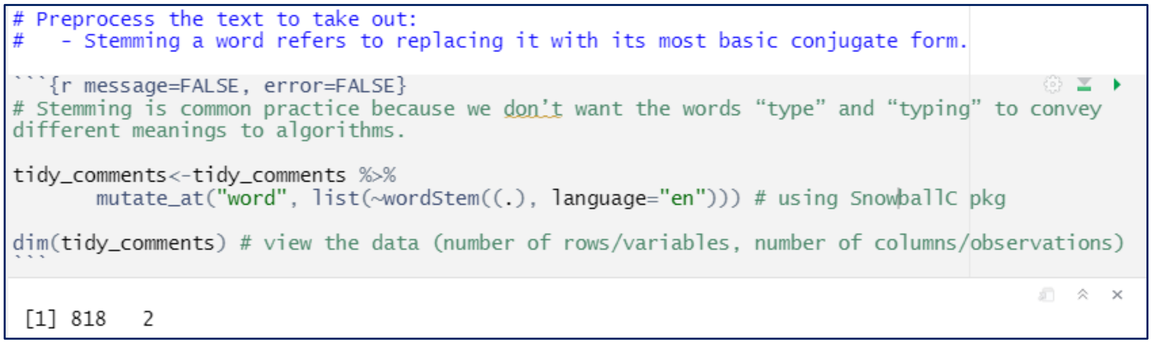
 [1] 818   2
```
```

Data and Analysis Continued …

Data Analysis Process Continued …

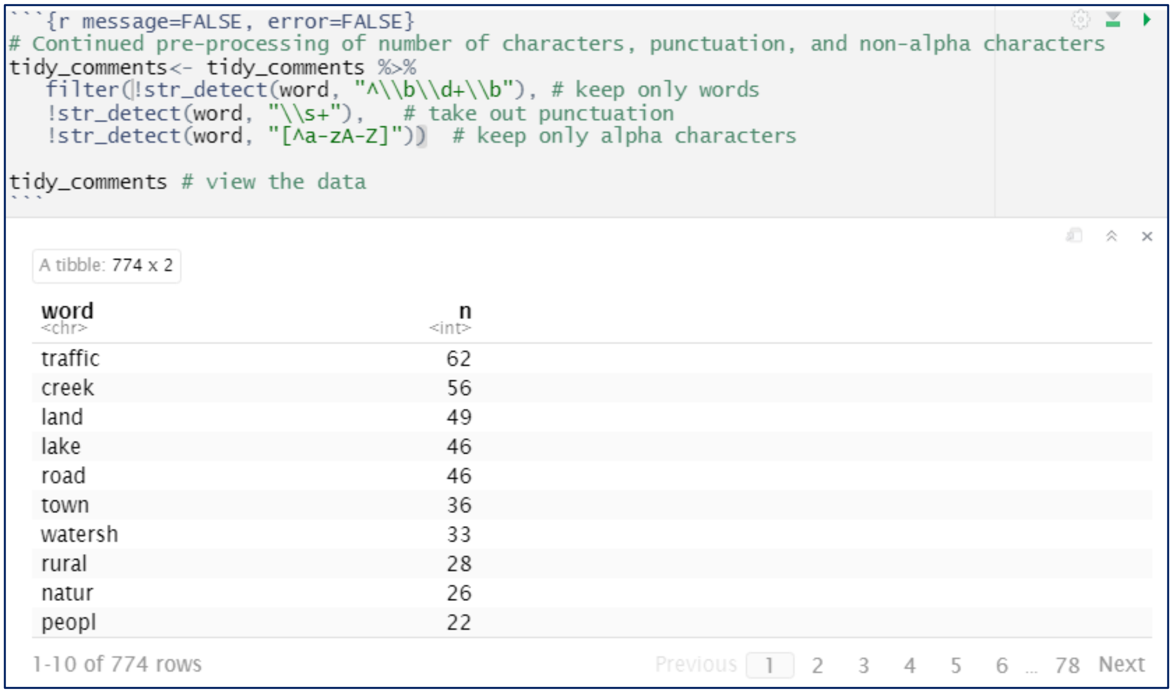
4. Text Preprocessing Continued …

   - *Stemming using the Snowball – **the data still has 818 words.***



~~~
# Preprocess the text to take out:
#    - Stemming a word refers to replacing it with its most basic conjugate form.

```{r message=FALSE, error=FALSE}
# Stemming is common practice because we don’t want the words “type” and “typing” to convey
different meanings to algorithms.

tidy_comments<-tidy_comments %>%
        mutate_at("word", list(~wordStem((.), language="en"))) # using SnowballC pkg

dim(tidy_comments) # view the data (number of rows/variables, number of columns/observations)
```

 [1] 818   2
~~~

   - *Punctuation and any non-alpha characters like Numbers – **the data now has774 words or rows.***



~~~
```{r message=FALSE, error=FALSE}
# Continued pre-processing of number of characters, punctuation, and non-alpha characters
tidy_comments<- tidy_comments %>%
   filter(!str_detect(word, "^\\b\\d+\\b"), # keep only words
   !str_detect(word, "\\s+"),    # take out punctuation
   !str_detect(word, "[^a-zA-Z]"))  # keep only alpha characters

tidy_comments # view the data
```
~~~

A tibble: 774 x 2

word <chr>	n <int>
traffic	62
creek	56
land	49
lake	46
road	46
town	36
watersh	33
rural	28
natur	26
peopl	22

1-10 of 774 rows

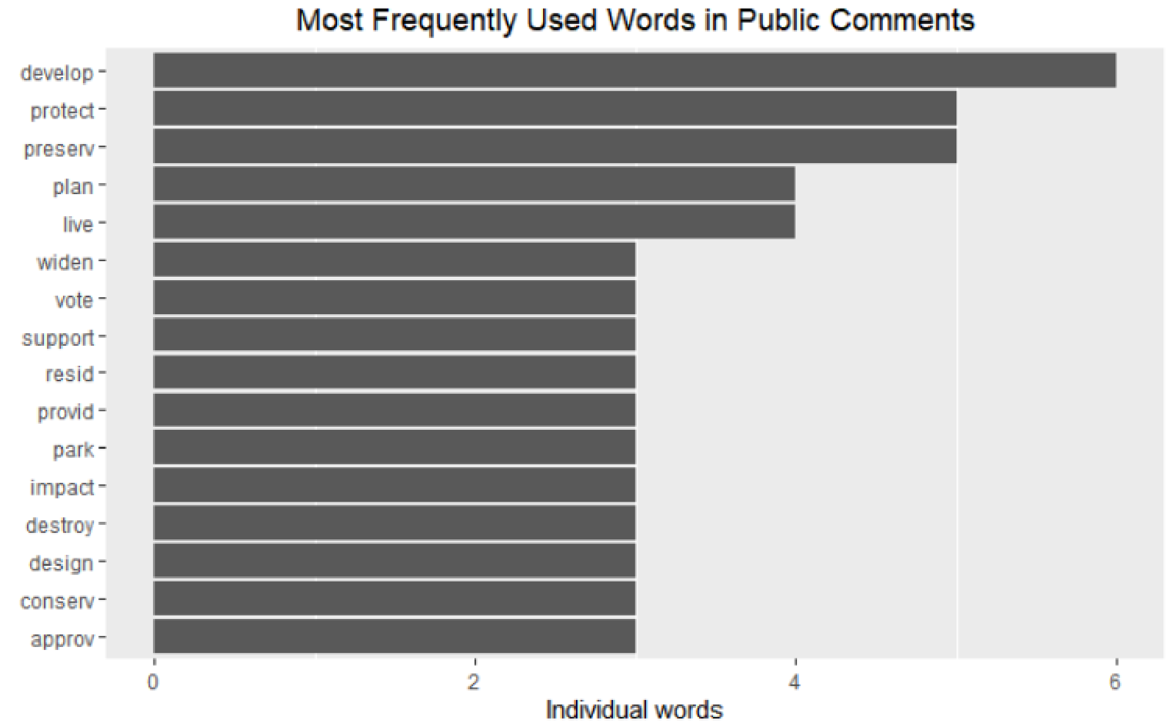
Data and Analysis Continued …

Data Analysis Process Continued …

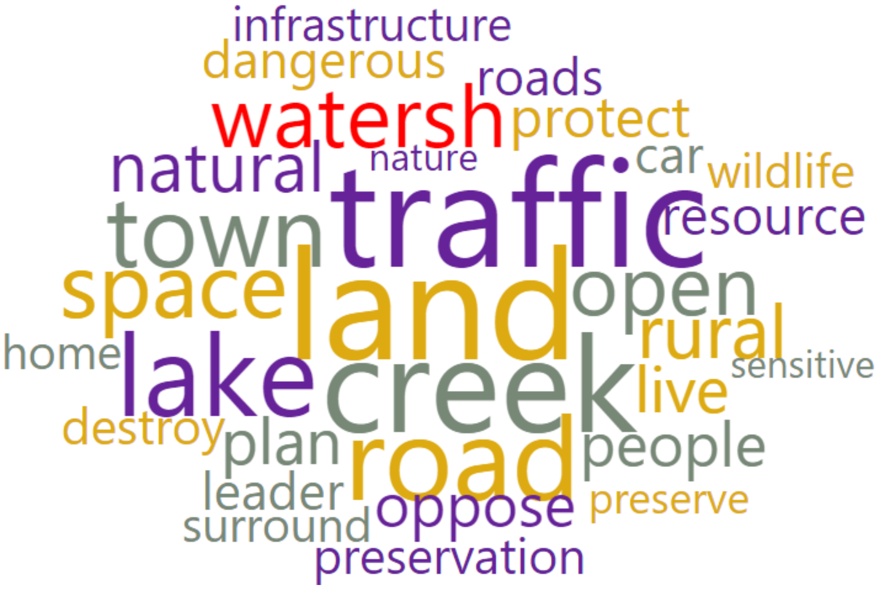
5. Transform the Data into a useful form for visualization and modeling.
   - *Visualize Word Frequency*

     Term frequencies

This picture to the right was created using the RStudio Application



*This picture to the right shows a simple **Wordcloud** using the Orange Data Mining Tool Application*



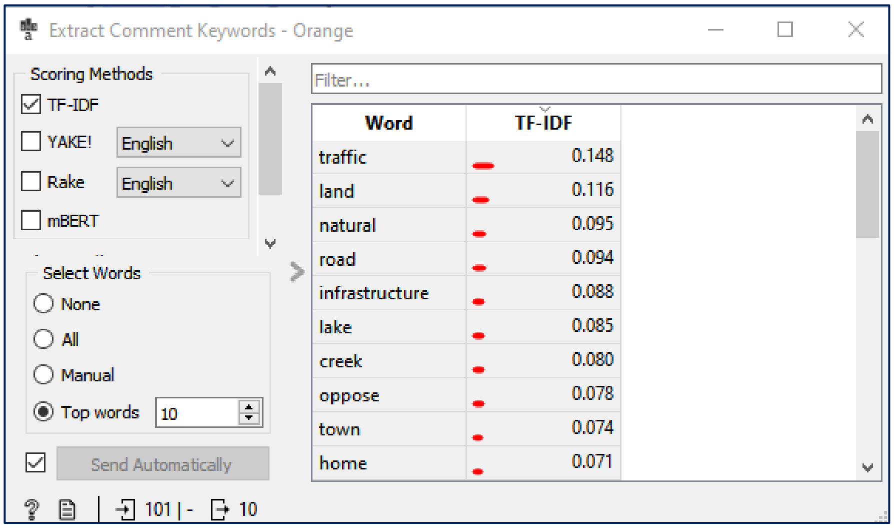
Data and Analysis Continued …

Data Analysis Process Continued …

5. **Transform the Data into a useful form for visualization and modeling.**
   - *Visualize Word Importance using the Orange Data Mining Tool Application*

     TF-IDF is "a method that scores by term frequency weighted by inverse document frequency"[8].

This picture to the right shows the high TF-IDF scores and lists the Top 10 repeated words after the Pre-Processing has been applied.



Extract Comment Keywords - Orange

Scoring Methods

☑ TF-IDF

□ YAKE! English

□ Rake English

□ mBERT

Select Words

○ None

○ All

○ Manual

● Top words 10

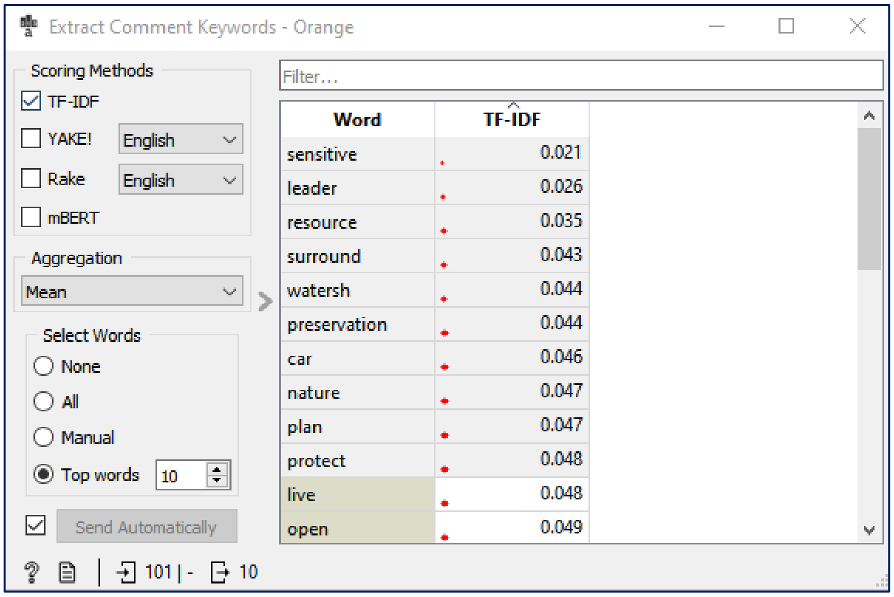
☑ Send Automatically

101 | - 10

Filter…

Word	TF-IDF
traffic	0.148
land	0.116
natural	0.095
road	0.094
infrastructure	0.088
lake	0.085
creek	0.080
oppose	0.078
town	0.074
home	0.071

This picture to the right shows the low TF-IDF score and lists the Top 10 unique or important words after the Pre-Processing has been applied.



Extract Comment Keywords - Orange

Scoring Methods

☑ TF-IDF

□ YAKE! English

□ Rake English

□ mBERT

Aggregation

Mean

Select Words

○ None

○ All

○ Manual

● Top words 10

☑ Send Automatically

101 | - 10

Filter…

Word	TF-IDF
sensitive	0.021
leader	0.026
resource	0.035
surround	0.043
watersh	0.044
preservation	0.044
car	0.046
nature	0.047
plan	0.047
protect	0.048
live	0.048
open	0.049

[8] https://orangedatamining.com/widget-catalog/text-mining/keywords/

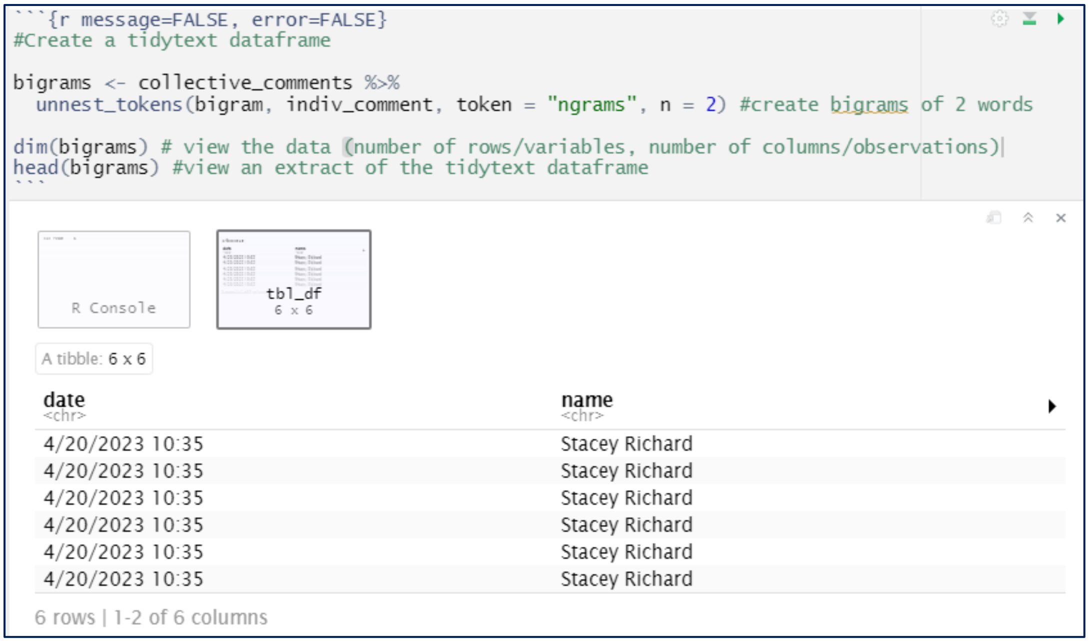
Data and Analysis Continued …

Data Analysis Process Continued …

5. **Transform the Data into a useful form for visualization and modeling.**

Ngrams are successive words in a document - Bigrams can be word 1|word2 and word2|word1, so both ways are calculated.

(a) Tokenization into bigrams – **the data has 7,280 pairs of words.**



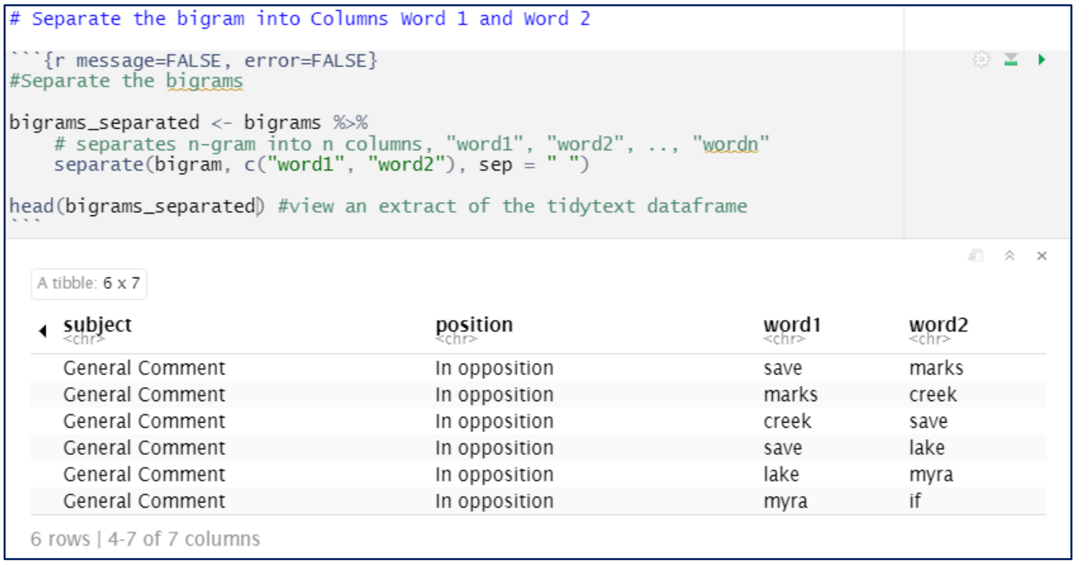
~~~~
```{r message=FALSE, error=FALSE}
#Create a tidytext dataframe

bigrams <- collective_comments %>%
  unnest_tokens(bigram, indiv_comment, token = "ngrams", n = 2) #create bigrams of 2 words

dim(bigrams) # view the data (number of rows/variables, number of columns/observations)
head(bigrams) #view an extract of the tidytext dataframe
```
~~~~

~~~~
A tibble: 6 x 6

date                                        name
<chr>                                       <chr>
4/20/2023 10:35                             Stacey Richard
4/20/2023 10:35                             Stacey Richard
4/20/2023 10:35                             Stacey Richard
4/20/2023 10:35                             Stacey Richard
4/20/2023 10:35                             Stacey Richard
4/20/2023 10:35                             Stacey Richard

6 rows | 1-2 of 6 columns
~~~~

(b) Separate bigrams into columns – **the data still has 7,280 pairs of words, just now there is 2 word columns.**



~~~~
# Separate the bigram into Columns Word 1 and Word 2

```{r message=FALSE, error=FALSE}
#Separate the bigrams

bigrams_separated <- bigrams %>%
    # separates n-gram into n columns, "word1", "word2", .., "wordn"
    separate(bigram, c("word1", "word2"), sep = " ")

head(bigrams_separated) #view an extract of the tidytext dataframe
```
~~~~

~~~~
A tibble: 6 x 7

subject                position               word1        word2
<chr>                  <chr>                  <chr>        <chr>
General Comment        In opposition          save         marks
General Comment        In opposition          marks        creek
General Comment        In opposition          creek        save
General Comment        In opposition          save         lake
General Comment        In opposition          lake         myra
General Comment        In opposition          myra         if

6 rows | 4-7 of 7 columns
~~~~

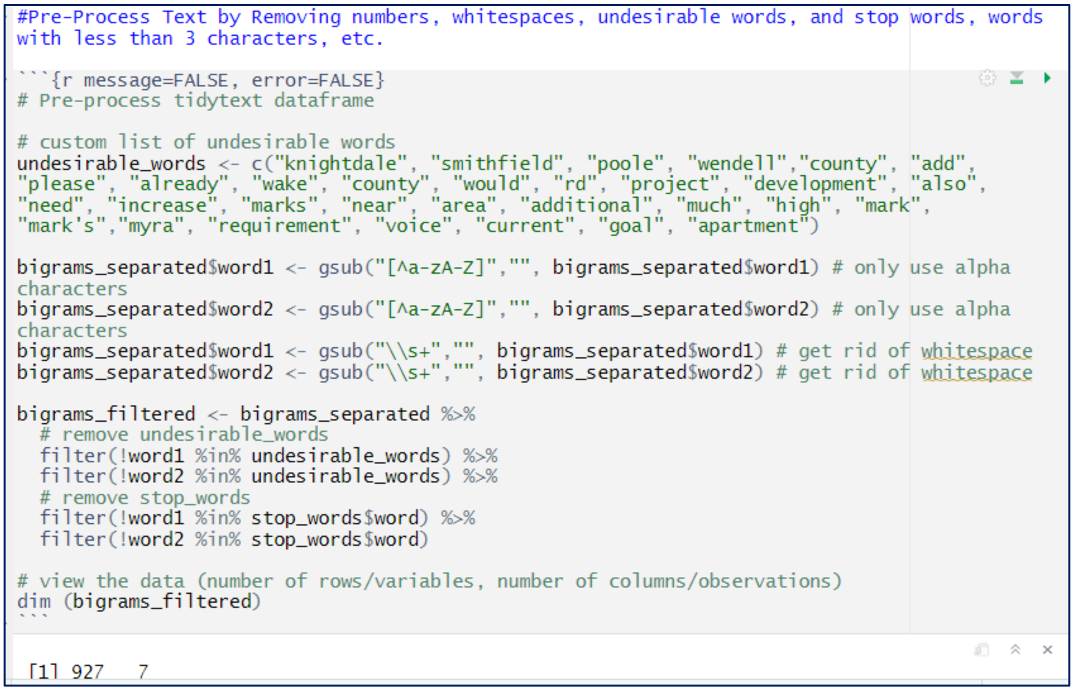
Data and Analysis Continued …

Data Analysis Process Continued …

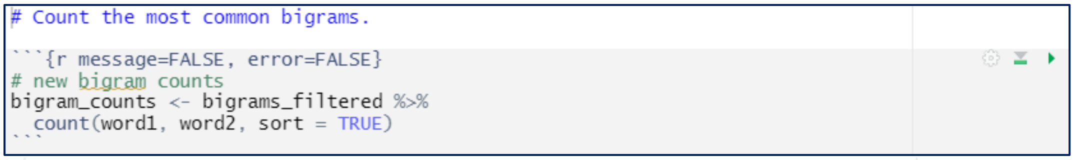
5. Transform the Data into a useful form for visualization and modeling.

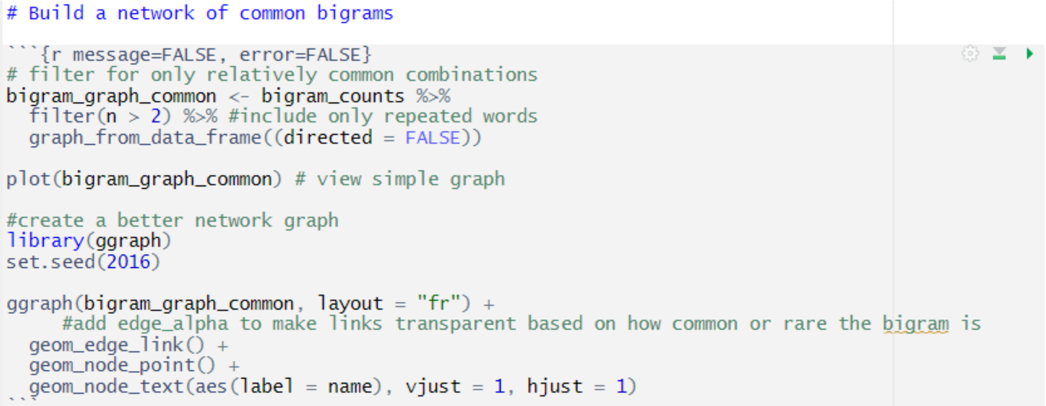
Ngrams are successive words in a document.

(c) Cleaning – **the data still has 927 pairs of words.**



~~~
#Pre-Process Text by Removing numbers, whitespaces, undesirable words, and stop words, words
with less than 3 characters, etc.

```{r message=FALSE, error=FALSE}
# Pre-process tidytext dataframe

# custom list of undesirable words
undesirable_words <- c("knightdale", "smithfield", "poole", "wendell","county", "add",
"please", "already", "wake", "county", "would", "rd", "project", "development", "also",
"need", "increase", "marks", "near", "area", "additional", "much", "high", "mark",
"mark's","myra", "requirement", "voice", "current", "goal", "apartment")

bigrams_separated$word1 <- gsub("[^a-zA-Z]","", bigrams_separated$word1) # only use alpha
characters
bigrams_separated$word2 <- gsub("[^a-zA-Z]","", bigrams_separated$word2) # only use alpha
characters
bigrams_separated$word1 <- gsub("\\s+","", bigrams_separated$word1) # get rid of whitespace
bigrams_separated$word2 <- gsub("\\s+","", bigrams_separated$word2) # get rid of whitespace

bigrams_filtered <- bigrams_separated %>%
  # remove undesirable_words
  filter(!word1 %in% undesirable_words) %>%
  filter(!word2 %in% undesirable_words) %>%
  # remove stop_words
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word)

# view the data (number of rows/variables, number of columns/observations)
dim (bigrams_filtered)
~~~

 [1] 927   7
```

(d) Network Graph to see which words relate to each other more.

(i) Count the most common bigrams.



```
# Count the most common bigrams.

```{r message=FALSE, error=FALSE}
# new bigram counts
bigram_counts <- bigrams_filtered %>%
  count(word1, word2, sort = TRUE)
```
```



```
# Build a network of common bigrams

```{r message=FALSE, error=FALSE}
# filter for only relatively common combinations
bigram_graph_common <- bigram_counts %>%
  filter(n > 2) %>% #include only repeated words
  graph_from_data_frame((directed = FALSE))

plot(bigram_graph_common) # view simple graph

#create a better network graph
library(ggraph)
set.seed(2016)

ggraph(bigram_graph_common, layout = "fr") +
     #add edge_alpha to make links transparent based on how common or rare the bigram is
  geom_edge_link() +
  geom_node_point() +
  geom_node_text(aes(label = name), vjust = 1, hjust = 1)
```
```
```

Data and Analysis Continued …

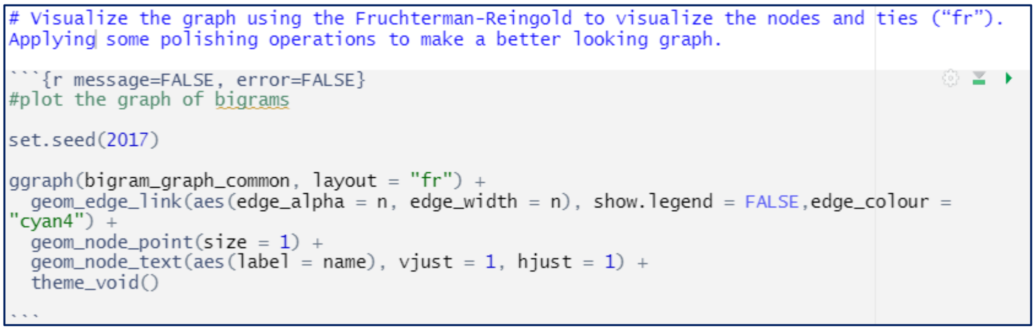
Data Analysis Process Continued …

5. **Transform the Data into a useful form for visualization and modeling.**

Ngrams are successive words in a document.

(d) Network Graph to see which words relate to each other more.

(ii) Enhance the graph to make it more visually appealing.

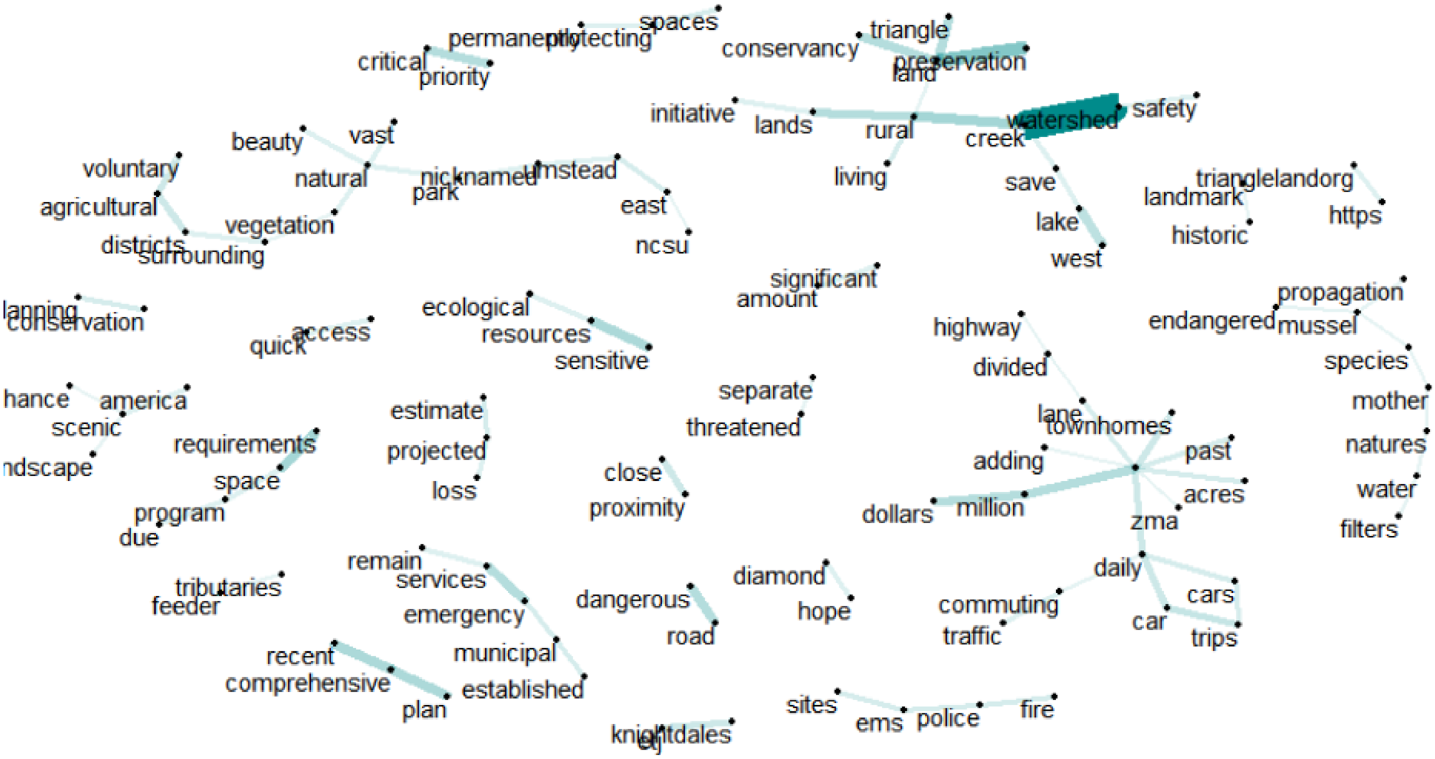


~~~
# Visualize the graph using the Fruchterman-Reingold to visualize the nodes and ties ("fr").
Applying some polishing operations to make a better looking graph.

```{r message=FALSE, error=FALSE}
#plot the graph of bigrams

set.seed(2017)

ggraph(bigram_graph_common, layout = "fr") +
  geom_edge_link(aes(edge_alpha = n, edge_width = n), show.legend = FALSE,edge_colour =
"cyan4") +
  geom_node_point(size = 1) +
  geom_node_text(aes(label = name), vjust = 1, hjust = 1) +
  theme_void()
```
~~~

This picture below shows the words "watershed", "creek" and "safety" are prominent by showing dark cyan lines as edges.

Another important group of words include "emergency", "services", "municipal", "remain" and "established"

"Ecological", "resources" and "sensitive" also stand out.

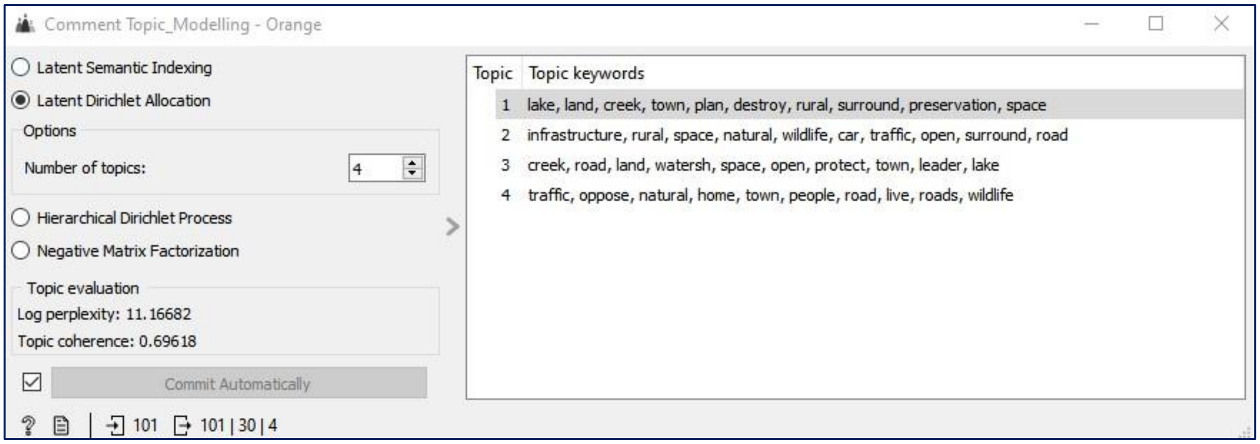


Data and Analysis Continued …

Data Analysis Process Continued …

6. **Transform the Data into a useful form for visualization and modeling.**

Topic Modeling using Orange Data Mining Tool – Here the application uses Latent Dirichlet Allocation (LDA) which is a statistical model that attempts to group the words in the comments into groups of similar words. In the Orange Data Mining Tool, the "Topic Model" is a quick choice to formulate groups. By increasing or decreasing the number of Topic groups, you can find the best number of groups given the current text.



Comment Topic_Modelling - Orange

○ Latent Semantic Indexing
● Latent Dirichlet Allocation
Options
Number of topics: 4
○ Hierarchical Dirichlet Process
○ Negative Matrix Factorization

Topic evaluation
Log perplexity: 11.16682
Topic coherence: 0.69618

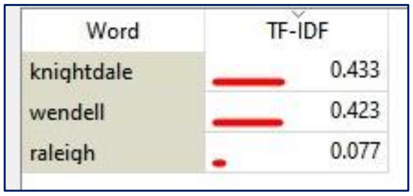
☑ Commit Automatically

101 | 101 | 30 | 4

Topic	Topic keywords
1	lake, land, creek, town, plan, destroy, rural, surround, preservation, space
2	infrastructure, rural, space, natural, wildlife, car, traffic, open, surround, road
3	creek, road, land, watersh, space, open, protect, town, leader, lake
4	traffic, oppose, natural, home, town, people, road, live, roads, wildlife

This picture above shows the following groups which can be summarized below:
(1) the town planning to destroy the rural land and natural resources
(2) the man-made infrastructure and roads vs. natural rural resources
(3) town leaders protect the creek, land, lake, and rural space
(4) people oppose traffic and roads where wildlife lives

Additional Note: The TF-IDF extract from the Orange Mining Tool in the picture below was applied to the pre-processed data on the Address of the residents making comments. It is easier to visualize where the comments are coming from.



Word	TF-IDF
knightdale	0.433
wendell	0.423
raleigh	0.077

Summary

I found that the abundance of related words in the comments for ZMA-15-22 Proposed Parkside at Westlake was about protecting the natural habitat and environment in the rural land. Residents have the opinion that town leaders want to destroy the natural rural space while the people enjoy the creek, lake, and wildlife.

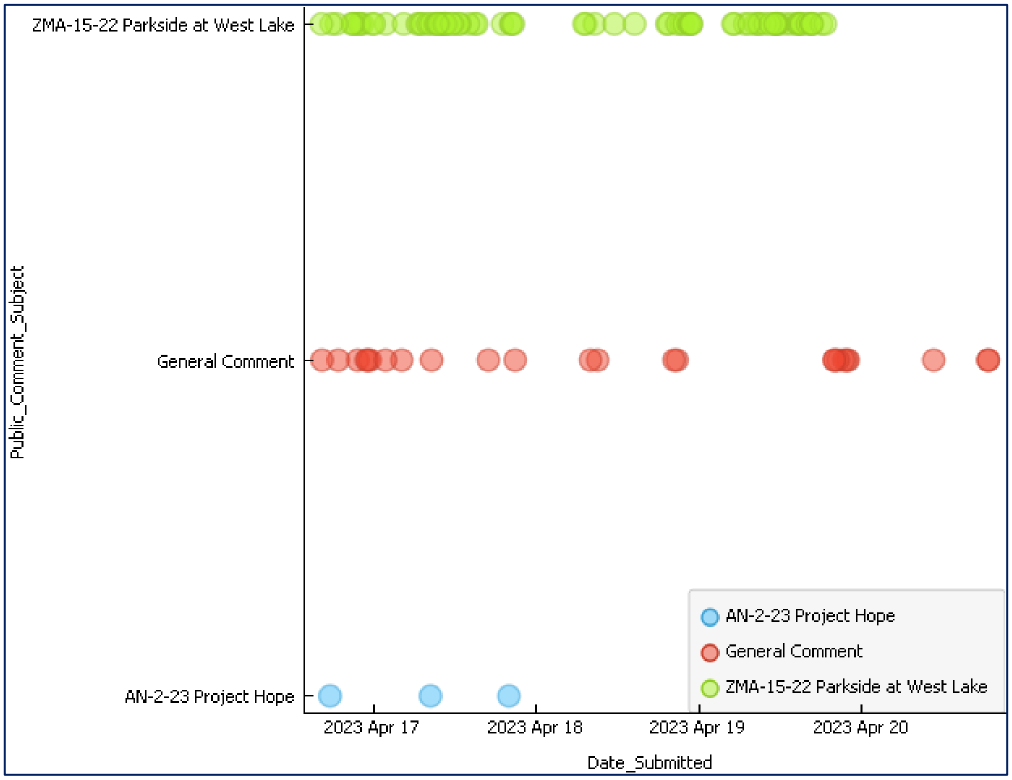
Through the advanced text analytical tools of RStudio (based on the R Programming Language) and the Orange Data Mining Tool (based on the Python programming language), the text was preprocessed and transformed. The purpose of this setup was to use certain features so that text exploration could occur such as word frequency, word importance, word relationships, and word grouping that led to the four general group topics. Then, visualizations could be created to help with understanding of these comments overall.

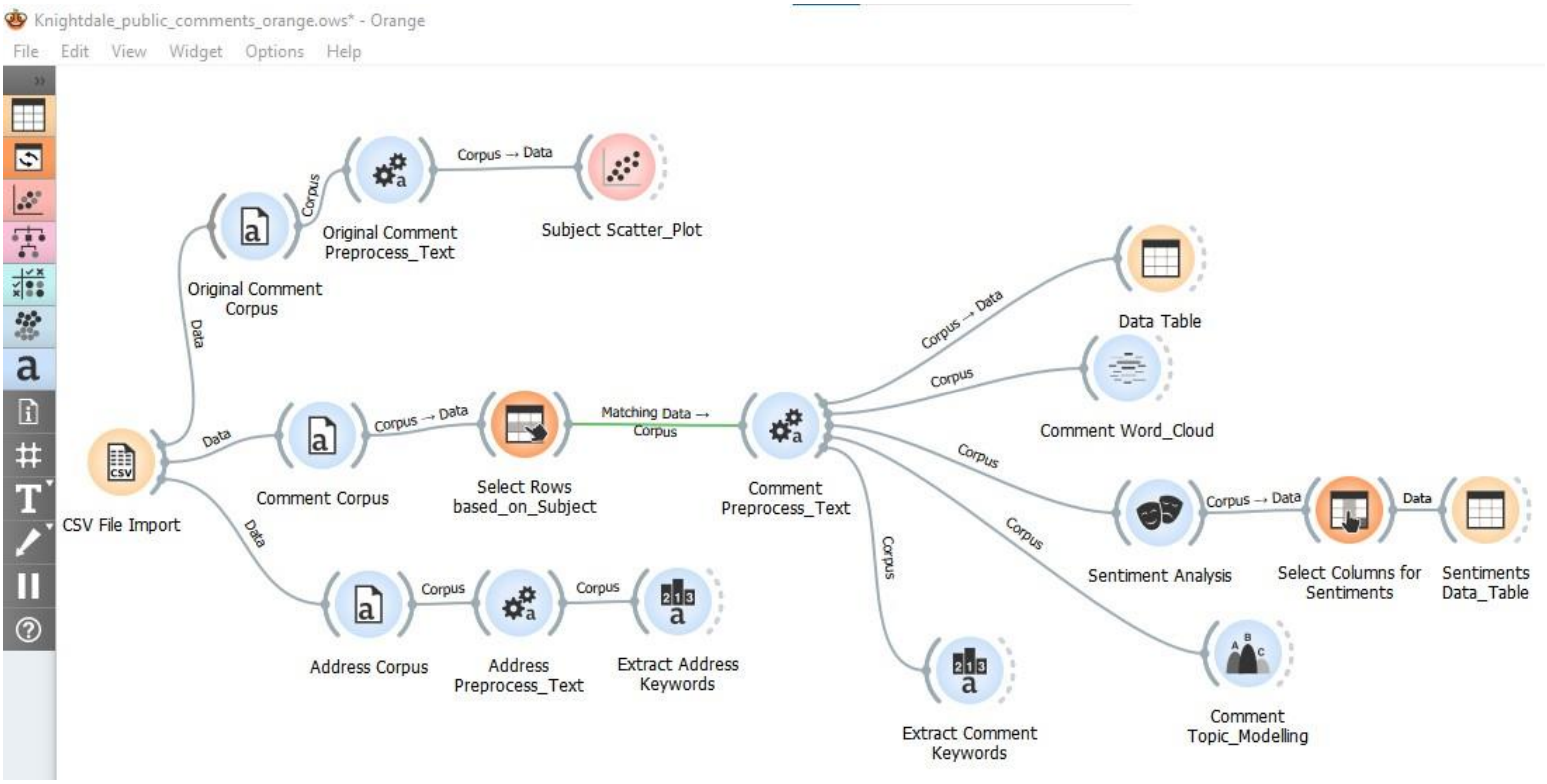
Appendix A (at the top) shows a chart on the subjects contained within these 104 Public Comment entries and showed that they represented three subjects. The topics "ZMA-15-22 Parkside at West Lake" and "General Comment" was expected as there are more than two developments proposed along the corridor nearby. However, the "AN-2-23 Project Hope" entries were surprising as they are miles away and not even traffic will impact the Parkside at Westlake proposed development.

The picture at the bottom of Appendix A shows the map in the Orange Data Mining Application and how the upper branch was easy to segment the Comments subjects to create the top chart. In the middle, the select rows for Vic O'Neal, Steven Hollowell (*who had two comments listed on different subjects*), and Becky Thigpen on the subject of AN-2-23 Project Hope were taken out of the dataset. The final or bottom branch of the map shows the addresses segmented to determine where the comments are coming from as seen at the bottom of the previous page. The RStudio programming did not have this problem addressed, but it should have been done in the cleaning step of the Text Preprocessing part of the Data Analysis.

From the Joint Public Hearing April 20, 2023

Appendix A


ZMA-15-22 Parkside at West Lake
General Comment
AN-2-23 Project Hope
Public_Comment_Subject
2023 Apr 17
2023 Apr 18
2023 Apr 19
2023 Apr 20
Date_Submitted
AN-2-23 Project Hope
General Comment
ZMA-15-22 Parkside at West Lake


Knightdale_public_comments_orange.ows* - Orange
File
Edit
View
Widget
Options
Help
Corpus
Corpus → Data
Original Comment Preprocess_Text
Subject Scatter_Plot
Original Comment Corpus
Data
Corpus → Data
Data Table
Corpus
Comment Word_Cloud
Data
Corpus → Data
Matching Data → Corpus
CSV File Import
Comment Corpus
Select Rows based_on_Subject
Comment Preprocess_Text
Corpus
Corpus → Data
Data
Corpus
Sentiment Analysis
Select Columns for Sentiments
Sentiments Data_Table
Data
Corpus
Corpus
Corpus
Address Corpus
Address Preprocess_Text
Extract Address Keywords
Extract Comment Keywords
Comment Topic_Modelling