Logo, company name

Description automatically generated

**TEXT MINING PROJECT**

**ISM 6359**

**Data mining**

Professor Ryan LaBrie

Ngoc Oanh Nguyen

Winter, 2023

1. **State a business reason for selecting your tools (problem you would like to solve).**

Mental health is a fundamental part of human beings and is especially taken into consideration in this developed era. The burden of depression and other mental health conditions is increasing worldwide, and people are affected by mental disorders and suicide rates have soared recently. In 2019-2020, 20.78% of adults were experiencing a mental illness. That is equivalent to over 50 million Americans.

People affected with mental health conditions can suffer greatly and function poorly in daily life. Millions of adults in the U.S. experience serious thoughts of suicide, with the highest rate among multiracial individuals. The percentage of adults reporting serious thoughts of suicide is 4.84%, totaling over 12.1 million individuals. In addition, over 1 in 10 youth in the U.S. are experiencing depression that is severely impairing their ability to function at school or work, at home, with family, or in their social life. Almost a third (28.2%) of all adults with a mental illness reported that they were not able to receive the treatment they needed.

Arrow

Description automatically generated with low confidence

Figure 1 - Mental health conditions in the U.S

In this project, I will use this corpus for toxic language detection. By creating a classification model to detect whether a comment is poisonous or not, which could be used to identify mental-health-related problems later.

1. **Document how you used the tool. Many tools are super rich in features, and you probably won’t be exploring them all, explain the parts you did use.**

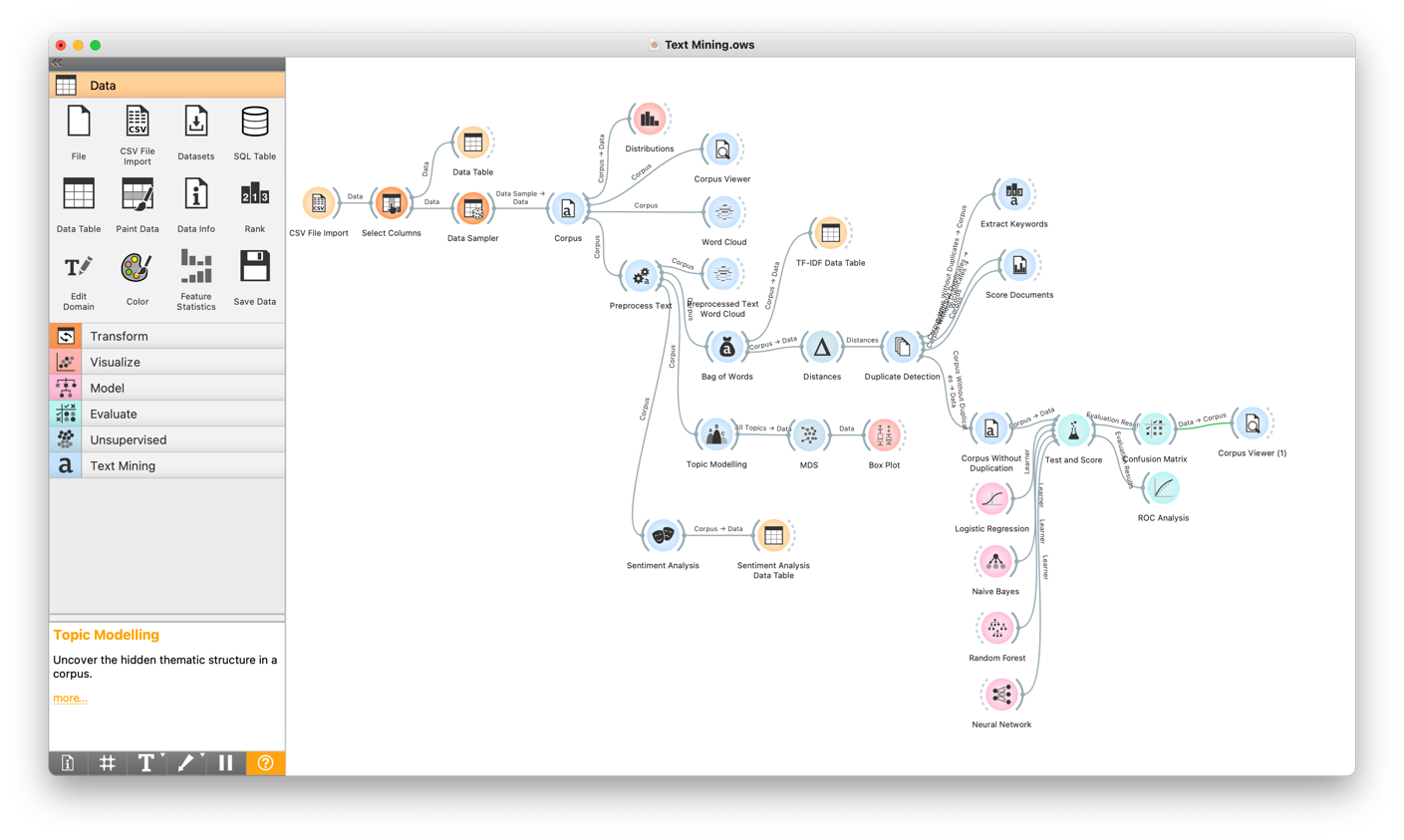


Figure 2 - Workflow

Firstly, I have to install the Text Mining features from Options/Add-ons. It might take a while and require reopening the Orange tool after completing installation.

Graphical user interface, application

Description automatically generated

Figure 3 - Text Mining installation

After that, since the dataset is a csv file, I load the file named “mental\_health.csv” into the tool using the CSV File Import widget. I also change the type of data of columns “text” and “label” to text and categorical respectively.

A screenshot of a computer

Description automatically generated

Figure 4 - Importing file

Then I use the Select Columns widget to set “label” as the target attribute for this dataset and “text” as the meta one.

Graphical user interface, application

Description automatically generated

Figure 5 - Setting target attribute

Due to its largeness, the Data Sampler is used to make a sample of 20% of the original dataset, so the data size reduces from about 28000 to just above 5000.

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure 6 - Sampling the data

Following this, I put data into the Corpus widget to create a corpus for text mining. The Corpus Viewer is used to display the corpus content.

Graphical user interface, application

Description automatically generated

Figure 7 - Creating a corpus

After that, by using the Distribution widget, I check that the data set is balanced between the number of documents that have the "1" and "0" and there are no missing values.

Graphical user interface

Description automatically generated

Figure 8 - Data distribution

The Corpus Viewer is used to display corpus contents as follow.

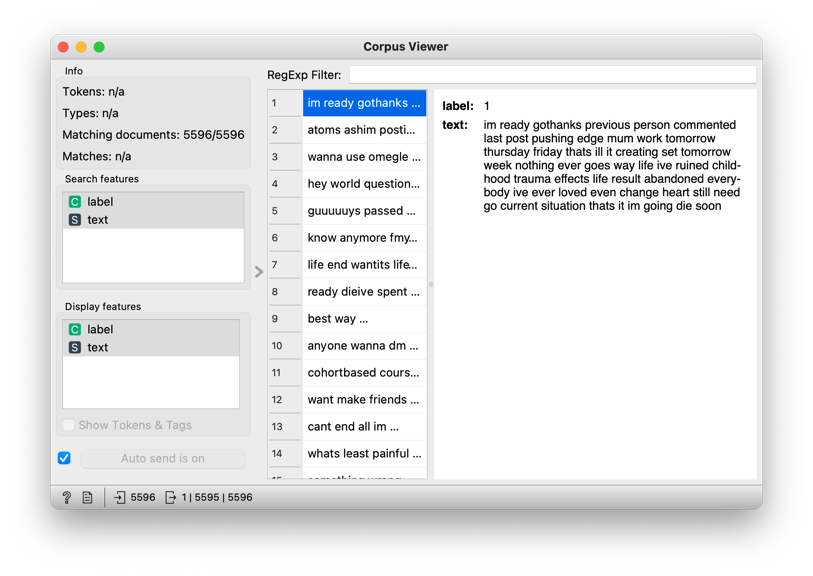


Figure 9 - Viewing the corpus

I also use Word Cloud to display tokens in the corpus and their size denoting the frequency of the word in the corpus. As can be seen from the figure, the most common words are “im”, “like”, “want”, “know”, “one”, “me”, etc.

Text

Description automatically generated

Figure 10 - Raw word cloud

Graphical user interface, application

Description automatically generatedA picture containing shape

Description automatically generatedHowever, there are a lot of them do not contribute important meaningful for text analysis. As a result, I utilize the Preprocess Text widget to construct a text pre-processing pipeline. In the Transformation processor, I lowercase all tokens, remove accents, parse HTML and remove URLs. Then I tokenize the corpus by the regular expression which defaulted to keep only words. Then it is filtered the stopwords based on uploaded the file containing common stopwords. I also filter numbers and remove tokens matching the listed regular expressions. Additionally, the processors normalize the corpus using the Porter stem and range it with N-grams with the range of 1 and 2.

Figure 11 - Preprocess Text

Figure 12 - Common stop words

After preprocessing, I check the word cloud of the corpus. Notably, the most common words are different now, consisting of "feel", "life", "friend", "live", "love", "fuck", and "kill". These have a more considerable contribution to detecting if the comment is poisonous or not.

Text

Description automatically generated

Figure 13 - Preprocessed word cloud

Then the Topic Modeling widget is utilized to discover abstract topics in a corpus based on clusters of words found in each document and their respective frequency. As can be seen from the result table using the Latent Semantic Indexing, there are 10 topics from the corpus with both positive and negative weights per topic. Positive words are colored green and negative words such as “pain”, “hate”, “fuck”, etc. are colored red. However, there are words that have both negative and positive meanings including “friend”, “monitor”, “pain” and “love”; so, they need to be put in a particular context to define what emotion they express.

Graphical user interface, text, application

Description automatically generated

Figure 14 - Topic modeling

Graphical user interface, application

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidenceNext, I connect Topic Modeling to MDS. and set the link to All Topics - Data to output a matrix of word weights by topic. After that, to explore which words are representative for the topic with the Box Plot widget by setting the output to Data – Data and the subgroup to Selected as well as check the Order by relevance to subgroups box.

Figure 15 - Data box plot

Figure 16 - MDS of Topic modeling

The Sentiment Analysis is also applied to compute the sentiment from text. Then the Data Table is used to observe new features. The compound represents the total sentiment of the text, where -1 is the most negative and 1 is the most positive.

Graphical user interface, table

Description automatically generatedGraphical user interface, application

Description automatically generated

Figure 17 - Sentiment analysis data table

Figure 18 - Sentiment analysis parameters

Graphical user interface, table

Description automatically generatedFrom the preprocessed text, I put data into Bag of Words widget to generate a bag of words from the corpus. I change the parameters to sublinear which is logarithm of term frequency (count) and the IDF ([inverse document frequency](http://nlp.stanford.edu/IR-book/html/htmledition/inverse-document-frequency-1.html)). The TF-IDF Data Table check the output and the final column in white represents term frequencies for each document.

Figure 19 - TF-IDF data table

Graphical user interface, application

Description automatically generated

Figure 20 - Bags of Words parameter

Following, the Distances widget computes a matrix of pairwise distances and Duplicate Detection one detects and removes duplicates from the corpus.

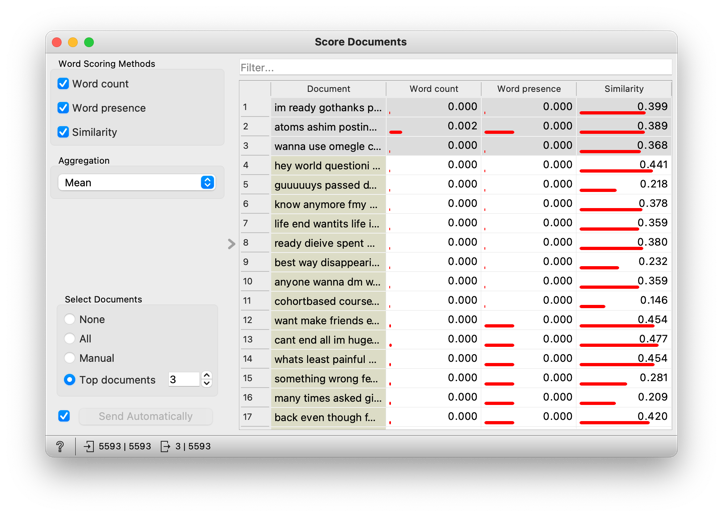
Graphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, application

Description automatically generated

Figure 21 - Distances parameters

Figure 22 - Duplicate detection

Graphical user interface

Description automatically generatedAfter that, I infer characteristic words from the corpus without duplicates by Extract Keywords widget and score documents based on word appearance by Score Documents one.

Figure 23 - Scoring documents

Figure 24 - Extracting keywords

As a result, I create a corpus without duplicates and put it in “Test and Score”.

1. **When you choose a data mining algorithm(s) for you mining model, tell us why you chose that one (or that category of algorithms).**

I need to classify whether the comment is poisonous or not, so I choose the typical classification algorithms including Logistic Regression, Naïve Bayes, Random Forest, and Neural Network. I also use the Cross-validation function with 10 folds inside the “Test and Score” widget for this model.

1. **Document how/where you got your data (if it is publicly available, or internal for a work project).**

This Mental Health Corpus is an available dataset on Kaggle ([Dataset Link](https://www.kaggle.com/datasets/reihanenamdari/mental-health-corpus?resource=download)). It is a collection of texts related to people with anxiety, depression, and other mental health issues. The corpus consists of two columns: one containing the comments, and the other containing labels indicating whether the comments are considered poisonous or not.

The data in the corpus may be useful for researchers, mental health professionals, and others interested in exchanging valuable mental health information and understanding the language and sentiment surrounding mental health issues for future research and practice in mental healthcare.

1. **Given an explanation/analysis of the output (What did you learn or uncover).**

After running, the “Test and Score" provides a table that helps to evaluate the best model which uses the Logistic Regression. It has the highest evaluation results with the Classification Accuracy of 89.9% and the AUC of 0.964.

Graphical user interface

Description automatically generated

Figure 25 - Test and Score results

Moreover, its ROC (Receiver Operating Characteristic) Curves is the top one so it can distinguish more accurately if a comment is considered as poisonous or not.

Graphical user interface

Description automatically generated

Figure 26 - ROC analysis

Lastly, I use the Corpus Viewer to display corpus contents and show the tokens and tags.

Graphical user interface, text, application

Description automatically generated

Figure 27 - Viewing a corpus with tokens and tags

1. **Conclude with the 3 W’s (What Went Well, What Did NOT go Well, What Would you do Differently Next Time).**

* What Went Well

The data preparation did not take so much time for me since the dataset is a clean one, so I did not have to transform them. Moreover, there is no missing data and confusingly classified attributes. Everything is in good condition which facilitates me to focus on manipulating the dataset.

The preprocessing phase run smoothly since I used Orange for the previous Data Mining project. Moreover, this tool is quite similar to the RapidMiner which the professor exemplified, so it did not take me a decent time to learn about this tool in terms of different names and operators in Text Mining. In addition, it is also rich in features and visualizations, thus it is easy for me to adapt and explore the data.

* What Did NOT go Well

Finding a good dataset is extremely time-consuming. In the first place, I chose another dataset but that one is not cleaned with a lot of missing values which are important. Therefore, I had to switch to other datasets before deciding to deal with this Mental Health corpus.

My computer crashed a lot of times due to the weak CPU since the corpus is extremely large (at about 28,000 documents). Consequently, I have to sample the dataset down to 20% but it took a really long time to run algorithms.

* What Would I do Differently Next Time

I want to spend more time exploring the data to understand each token and their distribution to the target one more comprehensively.

I will try to detect the outliers of the data to enhance the classification accuracy.

I also want to try a different dataset next time with other algorithms such as clustering and use a different tool like KNIME or Python with a larger size of dataset broaden my knowledge in this field.