BC Stats Capstone Final Report

Text Analytics: Quantifying the Responses to Open-Ended Survey Questions

Team Members: Carlina Kim, Karanpal Singh, Sukriti Trehan, Victor Cuspinera
Partner: BC Stats | Mentor: Varada Kolhatkar

2020-06-23

Executive Summary

BC Stats conducts the Work Environment Survey (WES) for BC Public Service's ministries with the goals of identifying areas for improvement and understanding employee's experiences. We have used natural language processing and machine learning classification techniques to automate labelling of responses to the open-ended questions into various themes and subthemes. These models may be used on their own or to assist human annotators to speed up the manual labelling process. We have also developed an app to explore the data with visualizations.

Introduction

The BC Public Service commits to maintaining the health of work environments and identifying areas for improvement through the Work Environment Survey (WES). The survey consists of ~80 quantitative questions using a 5 Likert scale and two open-ended qualitative questions, shown below, which have not been fully looked into by BC Stats. Due to this, we are solely focusing on these qualitative questions.

Question 1: *"What one thing would you like your organization to focus on to improve your work environment?"*

Question 2: *"Have you seen any improvements in your work environment and if so, what are the improvements?"*

BC Stats has been manually encoding the responses to these questions into various themes and subthemes, which is time consuming and expensive. We propose using multi-label machine learning classifiers and natural language processing to automate this process. We compare our results of classifying main themes with the results obtained by last year's MDS Capstone group, who worked on the same problem. In addition, we explored two problems which were not explored last year. First, we propose an approach to classify subthemes, and second, we propose an approach to classify responses of Question 2. Accordingly, our objectives are as follows:

Our Objectives

1) Build a model to automate multi-label text classification that:
   - Predicts label(s) for Question 1 and 2's main themes
   - Predicts label(s) for Question 1's subthemes
2) Build an app for visualizations of text data:
   - Identify and compare common words used for each question
   - Identify trends on concerns (from Question 1 responses) and appreciations (from Question 2 responses) in BC ministries workplaces over the given years.

Data

There are 31,000+ labelled comments (2013, 2018, 2020) and 12,000+ additional unlabelled comments (2015) for Question 1. Question 2 has 6,000+ labelled comments (2018) and 9,000+ additional unlabelled comments (2015, 2020). The 12 theme and 63 subtheme labels for Question 1 have been used in the past and deemed reliable by BC Stats, whereas Question 2's themes needed further analysis.

Data Science Methodology

Multilabel-Theme Classification

Our main approach involves using multi-label theme classification methods. We first develop a theme model that labels the survey comments into their predicted themes. Next, we follow a top-down approach where we define a hierarchical two-stage model to predict the subthemes. In stage 1, we predict the themes for the comments using the main theme model. In stage 2, based on the predicted themes from stage 1, we further classify the comment down into their respective subthemes.

We handled sensitive information in our preprocessing step using Named Entity Recognition to anonymize the comments. We used traditional feature-based classifiers as well as pre-trained embeddings and deep-learning classifiers to produce our models.

Baseline Models: TF-IDF Vectorizer + LinearSVC

For our baseline approach, we used a multi-labelling method called Classifier Chains, which models correlations between the multiple labels. TF-IDF Vectorizer creates a sparse representation of text with TF-IDF features that weighs most interesting words more. We use this representation with Classifier Chains and linear support vector classifier (LinearSVC), which is our baseline model. Other traditional models such as RandomForest and GaussianNB resulted in either slow training times or low results.

Advanced Models: Pre-trained Embeddings + Deep Learning Models

Other than TF-IDF representation, we also explored semantically richer text representations with word and text embeddings such as GloVe, FastText and Universal Sentence Encoder. These were used to build embedding matrixes and padded data to fit into the embeddings sizes. This was important as it allowed us to upload our sensitive data onto public cloud services (Google Colab) to apply more computationally expensive deep learning models. The advanced models we explored were Multi-channel CNNs, CNNs, and Bidirectional GRUs or Bi-GRUs.

Multi-channel CNNs are multiple versions of standard CNN models. For our model (Figure 1), we have used different kernel sizes for each channel. Specifically, we have defined a model with 3 input channels for kernel sizes 4, 6, and 8.

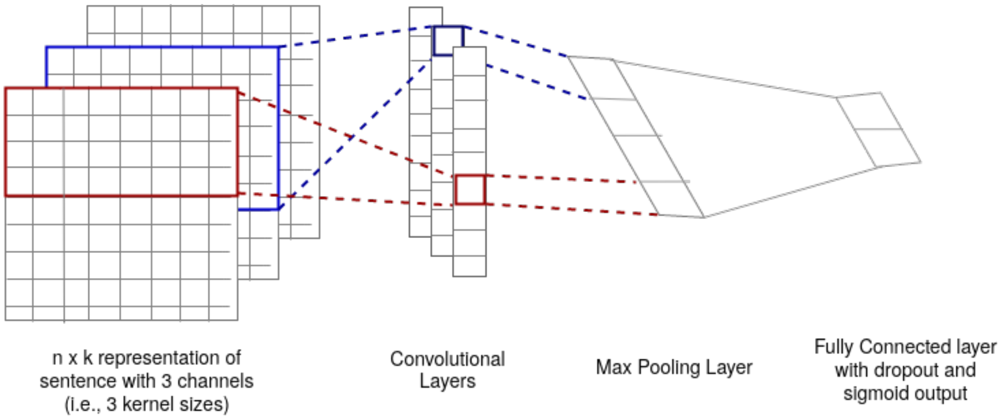


Figure 1: Visual representation of our multi-channel CNNs with kernel sizes 4, 6, and 8.

The GRU part of Bi-GRUs are similar to LSTMs, with the major difference being that GRUs have 2 gates (reset gate and update gate) instead of 4 (forget, input, update, output). Bidirectional means the model uses sentence sequences from both left-to-right and right-to-left to better represent the overall context of the word. We achieved the best results with the FastText embeddings trained on a common crawl with Bi-GRU.

Comparing Question 2 to Question 1:

Question 2 posed several challenges as the inital themes for this question were deemed unreliable to cover the full scope of the problem. Additionally, BC Stats wanted Question 2's themes to align better with themes from Question 1. To address this, we compared the words used and the frequency of these words in the responses for both questions.

The plot below (Figure 2) shows a linear trend in the frequency of common words between Question 1 and Question 2. For example, the word "communication" would appear approximately 1 out of 100 words read in the comments for both questions. We also assume concern and appreication within the workforce focus on similar topics. The similarity in the vocabulary and frequency of words supported our decision to use our theme model developed using Question 1's themes for Question 2's predictions.

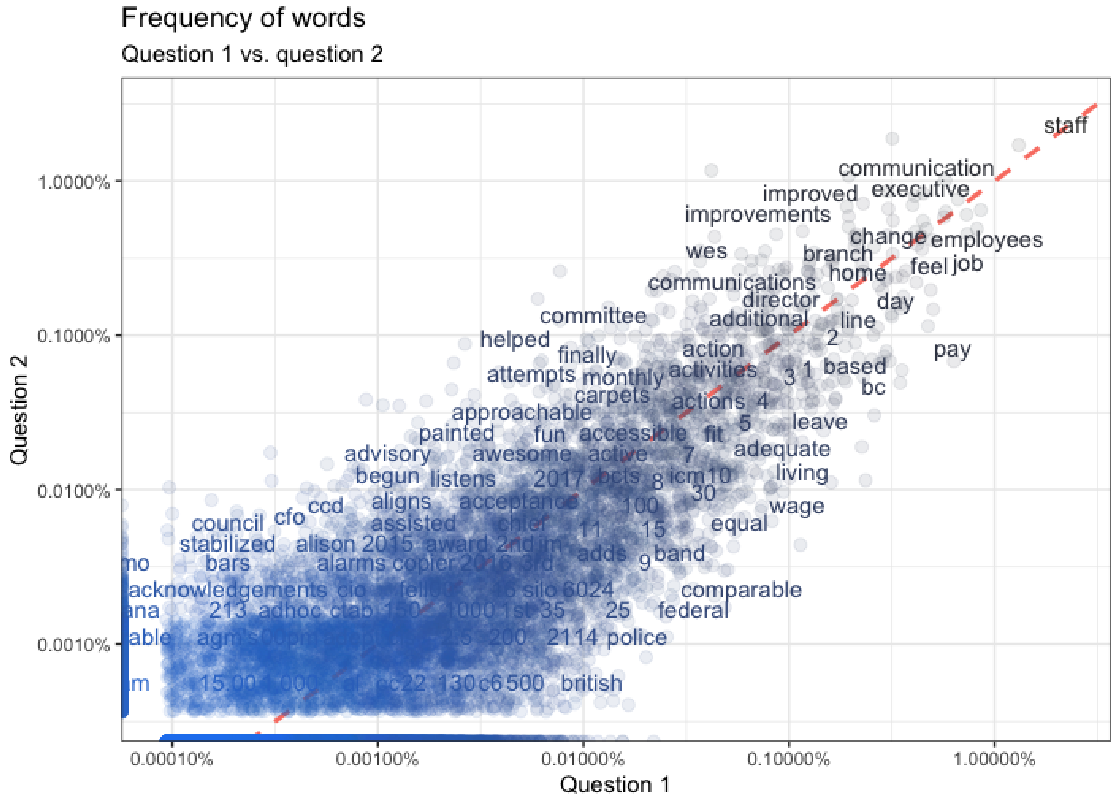


Figure 2: Frequency word comparison chart for responses from both questions.

Data Product and Results

Our data product consists of five components: trained theme and subtheme models, a pipeline to evaluate the models, a pipeline to classify new comments, an app that summarizes the text data with visualizations, and a report detailing the methodologies and results.

Results for Theme Labelling Models

As our data contained sparsity and class imbalance, accuracy does not represent the true underlying results due to false positives and negatives. Therefore, we measured the success of our models using precision and recall. Precision shows the average proportion of predictions that are correct. Recall shows the average proportion of all the correct labels that were predicted. We focused on balancing the precision and recall results for our models.

We also used precision recall curves to determine our deep learning model. Figure 3 shows the results for our deep learning models using FastText embeddings. Our best performing model was FastText + BiGRU shown as the highest curve on the precision recall plot.

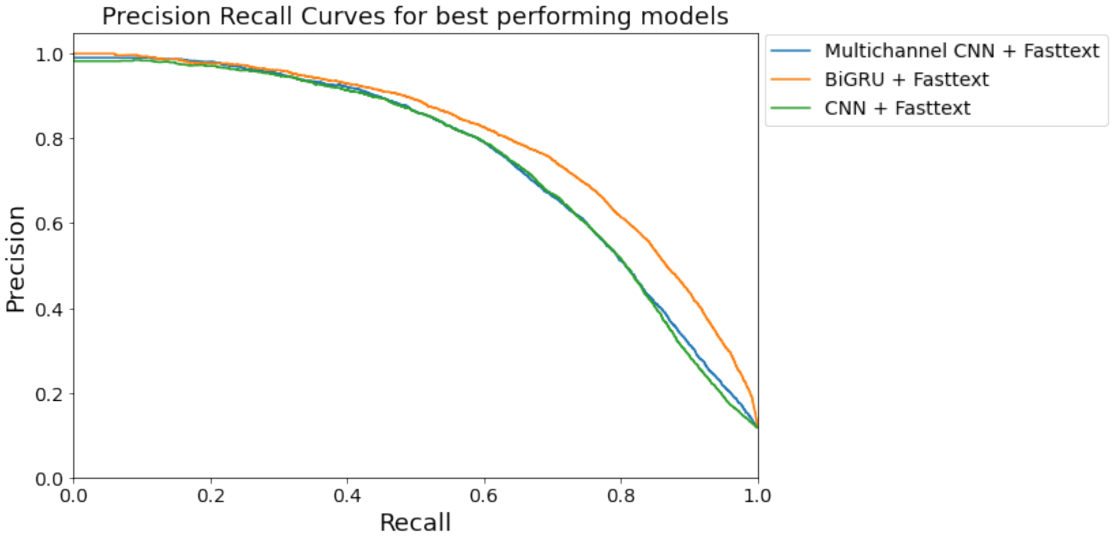


Figure 3: Precision recall curve showing results at various thresholds of our deep learning models using FastText embeddings using validation data. BiGRU + FastText performed the best results.

The table below shows the results of the BiGRU + FastText being evaluated at various thresholds on the validation set.

Table 1: Evaluation of Bi-GRU + FastText model at various thresholds.

Threshold	Accuracy	Precision	Recall	F1.measure
0.3	0.5004918	0.7035023	0.7394032	0.7210061
0.4	0.5240999	0.7436425	0.7053820	0.7240072
0.5	0.5339367	0.7741571	0.6691299	0.7178222
0.6	0.5290183	0.7989853	0.6367819	0.7087213
0.7	0.5197718	0.8268972	0.6001115	0.6954836
0.8	0.4999016	0.8568653	0.5534021	0.6724839
0.9	0.4652764	0.8993289	0.4857780	0.6308166

At the 0.4 threshold, our model observed the best precision and recall scores of 0.744 and 0.705 respectively. We used these values at threshold 0.4 for our theme model.

Below are our results for the baseline and deep learning model. We compared our results to last year's capstone results. The baseline models performed similarly while we have increased our recall score in our advanced model. This was part of our goal as increasing recall reduces the false negatives on our data.

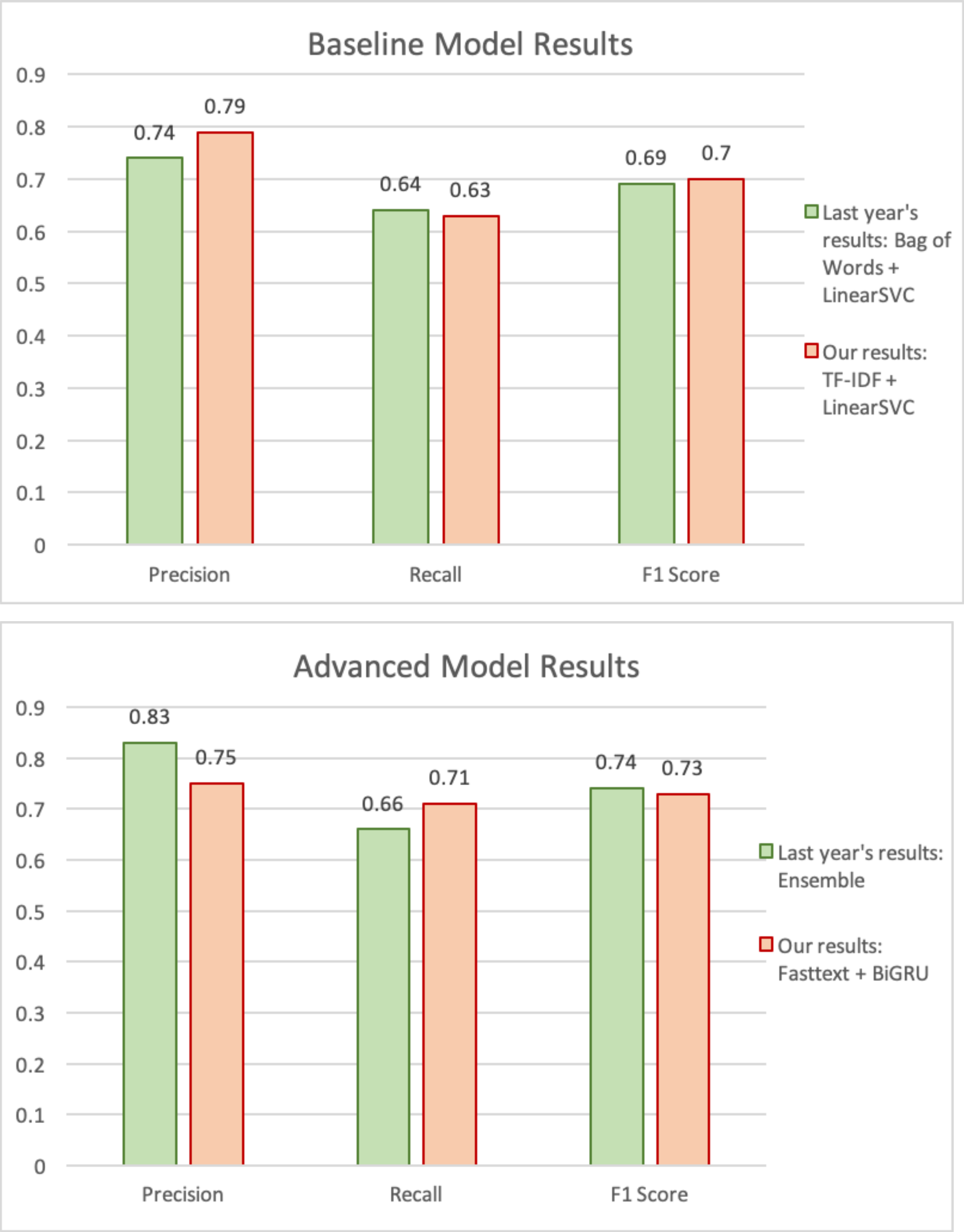


Figure 4: Comparison bar charts of our model's and last year's capstone group's model's performance on their precision, recall and F1 scores.

The label-wise results below (Table 2) indicate that most themes have high precision and recall. **"Y_count"** is the test data's true number of comments in the theme, **"Pred_count"** is how many comments our model predicted, and **"Y_count_train"** is the number of training data the model had for each theme.

Table 2: Prediction on test set for our main theme models performance for labelling each theme

Label	Y_count	Pred_count	Y_count_train	Accuarcy	Precision	Recall	F1.Score
CB	972	967	3113	0.9689959	0.9007239	0.8960905	0.8984012
TEPE	1389	1288	4667	0.9510545	0.9184783	0.8516919	0.8838252
CPD	867	884	2578	0.9394082	0.7726244	0.7877739	0.7801256
SP	629	614	2103	0.9519987	0.7638436	0.7456280	0.7546259
FWE	403	426	1413	0.9671073	0.7276995	0.7692308	0.7478890
SW	1048	916	3235	0.9055713	0.7445415	0.6507634	0.6945010
Exec	744	824	2511	0.9184766	0.6371359	0.7056452	0.6696429
VMG	868	922	2743	0.8989613	0.6225597	0.6612903	0.6413408
EWC	556	453	1809	0.9394082	0.6887417	0.5611511	0.6184341
Sup	760	659	2416	0.9135977	0.6600910	0.5723684	0.6131078
RE	549	403	1747	0.9383066	0.6947891	0.5100182	0.5882353
OTH	229	156	801	0.9604973	0.4294872	0.2925764	0.3480519

When using the model, theme comments with high precision and recall scores (> 0.67) do not need to be confirmed and can be encoded automatically. The themes with precision and recall scores above 0.67 are **CB, TEPE, CPD, SP, FWE**. We are confident that the model is prone to less error when predicting these.

Themes with lower precision and recall scores such as **OTH** should be manually verified by the BC Stats team. For OTH theme, is it the case that the model performs poorly because it is a catchall category and probably the model was unable to find patterns for this category. We recommend using a combination of both machine learning and human annotators for optimal results.

Results for Subtheme Models

We've built 12 independent models for each theme with their respective subthemes. These models will be used on the comment depending on the theme it was labelled to in stage 1. Majority of these models showed best results using FastText + BiGRU, except for the subthemes in theme `CB` which used FastText + CNN.

The precision recall curve shows the test results for all of the subtheme models. The minimum desirable threshold for the subtheme models precision and recall values shared by BC Stats was both 0.5. All our subtheme models surpass this threshold, when predicting on the test comments with their true main themes.

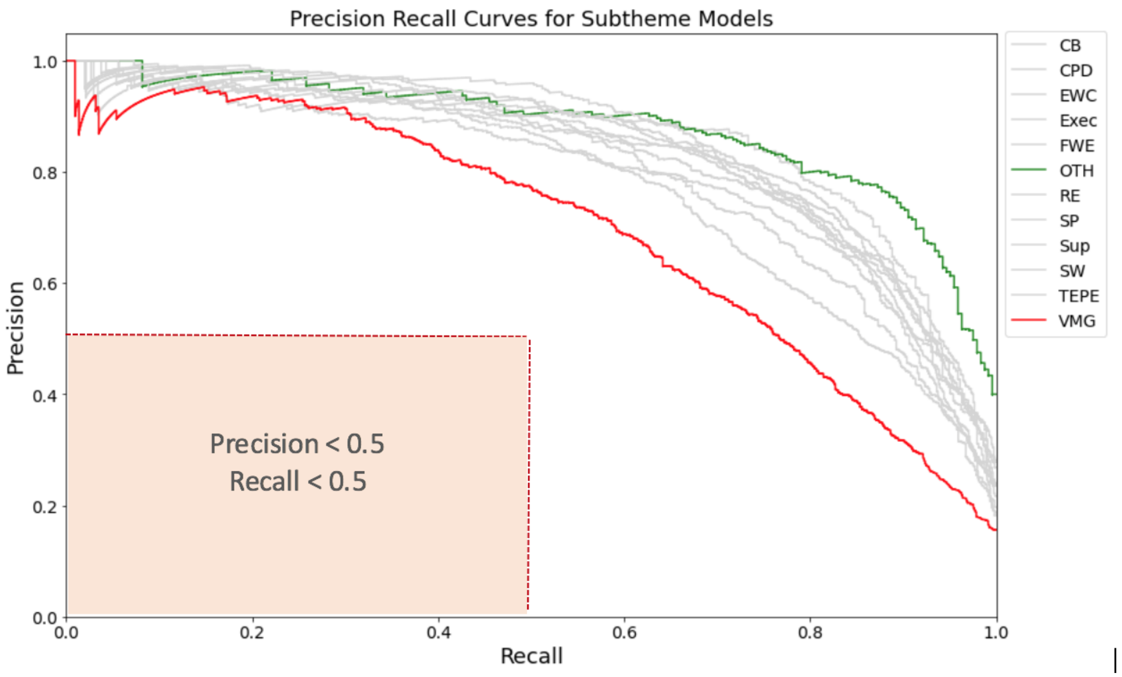


Figure 5: Precision recall curve for best performing subtheme models which all passes BC Stat's satisfaction condition (> 0.5).

Below are our results using the 2-stage hierarchical approach. The following results are based on the test data after themes for the comments have been predicted using the main theme model.

Table 3: Results of subtheme models performance during 2nd stage of hierarchical approach. After comments have been predicted using the main theme model.

Subtheme_model	Accuracy	Precision	Recall	F1.Score
TEPE	0.8761410	0.7100000	0.6826923	0.6960784
CB	0.9129682	0.6428571	0.6456693	0.6442601
CPD	0.9087189	0.6360759	0.6347368	0.6354057
FWE	0.9554611	0.6160714	0.6287016	0.6223224
SP	0.9335851	0.6171516	0.6106061	0.6138614
SW	0.8731508	0.6245059	0.5306465	0.5737631
Exec	0.8923513	0.5113759	0.5756098	0.5415950
EWC	0.9231980	0.5391791	0.4699187	0.5021720
RE	0.9288637	0.5911330	0.4088586	0.4833837
Sup	0.8912496	0.5125698	0.4503067	0.4794252
VMG	0.8629210	0.4106090	0.4432662	0.4263131
OTH	0.9579792	0.3950617	0.2622951	0.3152709

Despite OTH having the best precision and recall curve shown in Figure 4, we have achieved low results using our hierarchical 2-stage approach. This is due to error from the main themes model being propagated into our 2nd stage of our hierarchical model. The theme OTH had low precision and recall in our themes model which can be attributed to its small sample size. We believe that increasing the sample size may lead to increase in precision and recall metrics.

Predicting Themes for Question 2

We evaluated Question 2's labelling with Question 1's themes using a sample data of annotated comments provided by BC Stats using 2020 survey responses. The comparison of the results are close in range and hence appropriate to use the main theme model for Question 2 as well.

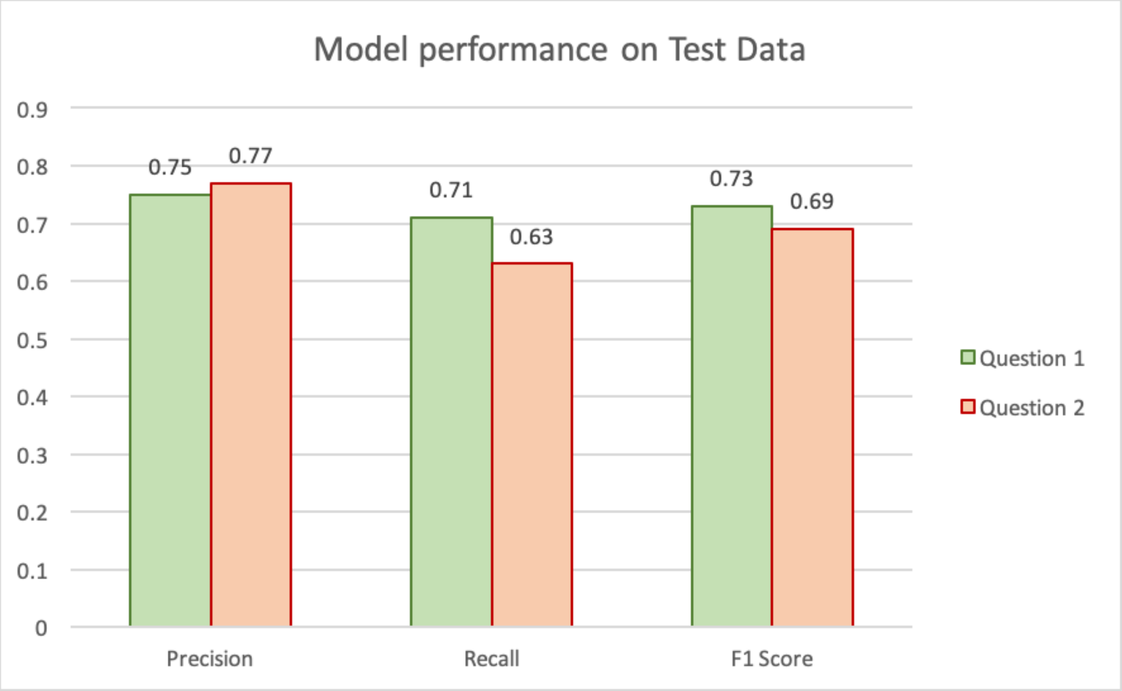
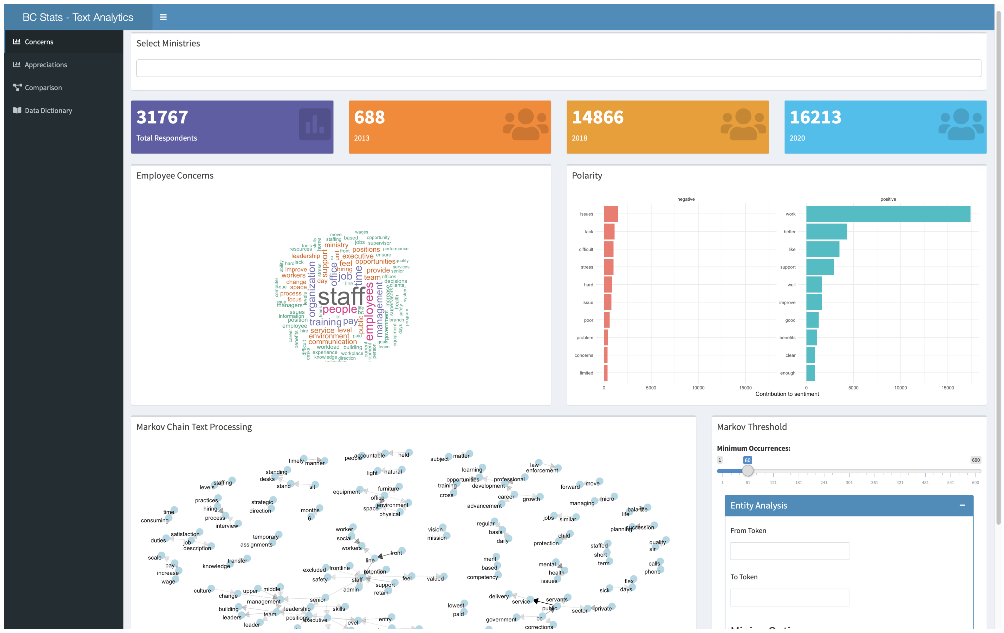


Figure 6: The bar graph compares how the main theme model performed predicting for both Question 1 and Question 2.

Dashboard

The dashboard provides a way for BC Stats to explore the summarized text data. There are 3 tabs, Concerns, Appreciations, and Comparison. The Concerns and Appreciations tabs looks into Question 1 and 2 respectively. Within each tab, we show the common texts in the comments using word cloud, interactions between relevant words using word ngrams, sentiment analysis of comments. In the Comparisons tab, you can compare the concern and appreciations expressed in each theme for the ministries over the given years. For example, you can see whether the theme Compensation and Benefits increased or decreased in appreciations and concerns for your ministry of interest from 2013-2020.



Minimum Occurrences: 60

Figure 7: First page of our dashboard

Conclusions and Recommendations

Classification Models

We've built several models to help the process of manually labelling survey questions. Although we haven't fully achieved a perfect model, it would make the manual labeling process easier and faster. Specifically, comments can be automatically encoded into themes **CB, TEPE, CPD, SP,** and **FWE** as they had precision and recall values ranging between 0.73-0.90 in the main theme model.

Our subthemes model, although had promising results during training, faced the consequences of the errors prior in the main theme model step. However, they were all able to perform higher than the minimum requirement desired by BC Stats. Particularly for subthemes models that performed low, such as OTH, we expect better results with more data.

We also recommend creating embeddings and padded data on sensitive data so it can be uploaded into public cloud services (Google Gollab, AWS). This can allow more complex machine learning algorithms to be incorporated to improve the model.

References

- BC Stats. (August 2018). 2018 Work Environment Survey Driver Guide.
- Province of British Columbia. (2020). About the Work Environment Survey (WES). Retrieved 2020-05-09
- Quinton, A., Pearson, A., Nie, F. (2019). BC Stats Capstone Final Report, Quantifying the Responses to Open-Ended Survey Questions. GitHub account of Aaron Quinton.
- Read J., Pfahringer B., Holmes G., Frank E. (2009) Classifier Chains for Multi-label Classification.
- Wikipedia. (2020, May 3). Multi-label classification. In Wikipedia, The Free Encyclopedia. Retrieved 2020-05-15.
- Linear SVC Machine learning SVM example with Python
- Brownlee J. (December, 2019) How to Develop a Multichannel CNN Model for Text Classification
- Wikipedia. (2020, June 14). Convolutional neural network. In Wikipedia, The Free Encyclopedia. Retrieved 2020-06-22.
- Silwimba F. (October, 2018) Bidirectional GRU for Text classification by relevance to SDG#3 indicators.
- Li S. (2018) Named Entity Recognition with NLTK and SpaCy