



BC Stats

Text Analytics:

Quantifying the Responses to Open-Ended Survey Questions

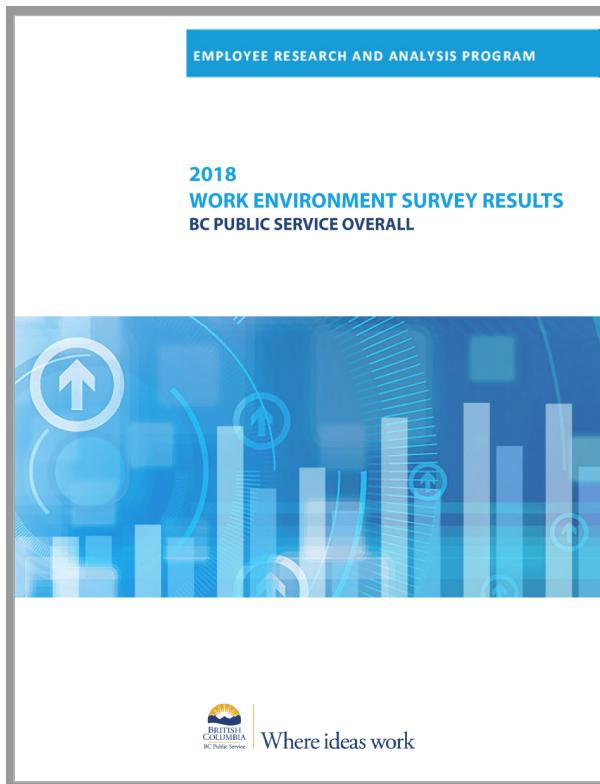
Carlina Kim, Karanpal Singh, Sukriti Trehan, Victor Cuspinera

Partner: BC Stats | Mentor: Varada Kolhatkar

2020-06-19

Introduction

The Survey



Work Environment Survey (WES)

- Conducted by BC Stats for employees within BC Public Service
- Measures the health of work environments and identifies areas for improvement
- ~80 quantitative questions (5 Likert scale) and 2 open-ended qualitative questions

Data

Open-ended Questions

Question 1

- **What one thing would you like your organization to focus on to improve your work environment?**

Example: "*Better health and social benefits should be provided.*"

Question 2

- **Have you seen any improvements in your work environment and if so, what are the improvements?**

Example: "*Now we have more efficient vending machines.*"

*Note: these examples are fake comments for privacy reasons.

Example of Data: Question 1

What one thing would you like your organization to focus on to improve your work environment?

| Comments* | CPD | CB | EWC | ... | CB_Improve_benefits | CB_Increase_salary |
|--|-----|----|-----|-----|---------------------|--------------------|
| Better health and social benefits should be provided | 0 | 1 | 0 | ... | 1 | 0 |

Theme: CB = Compensation and Benefits

Subtheme: CB_Improve_benefits = Improve benefits

Question 1:

- Comments encoded into **12 themes** and **63 subthemes**
- +31,000** labelled comments for 2013, 2018, 2020, **+12,000** additional comments from 2015

Question 2:

- Themes also encoded, but not as reliable as Question 1's
- +6,000** labelled comments for 2018, **+9,000** additional comments from 2015, 2020

*Note: this is a fake comment as an example of the data.

Our Objectives

1) Build a model to automate multi-label text classification that:

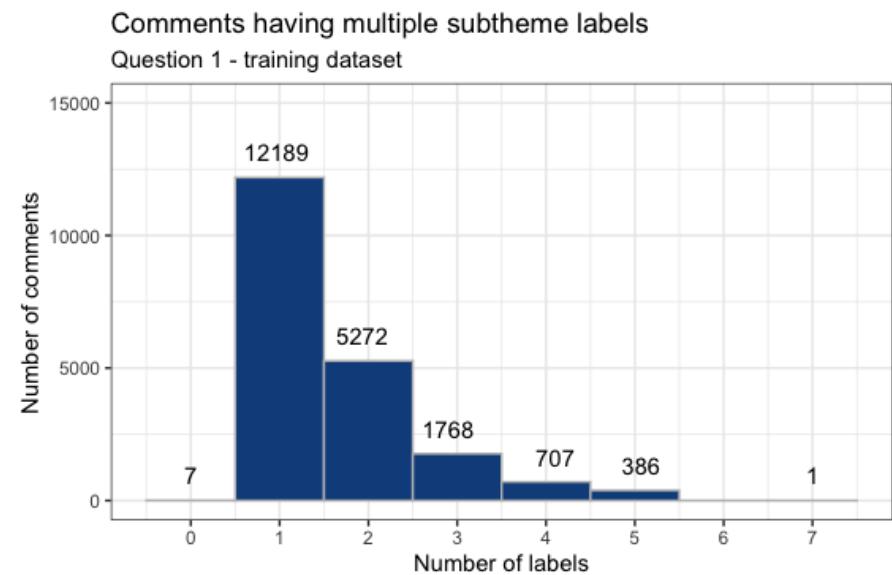
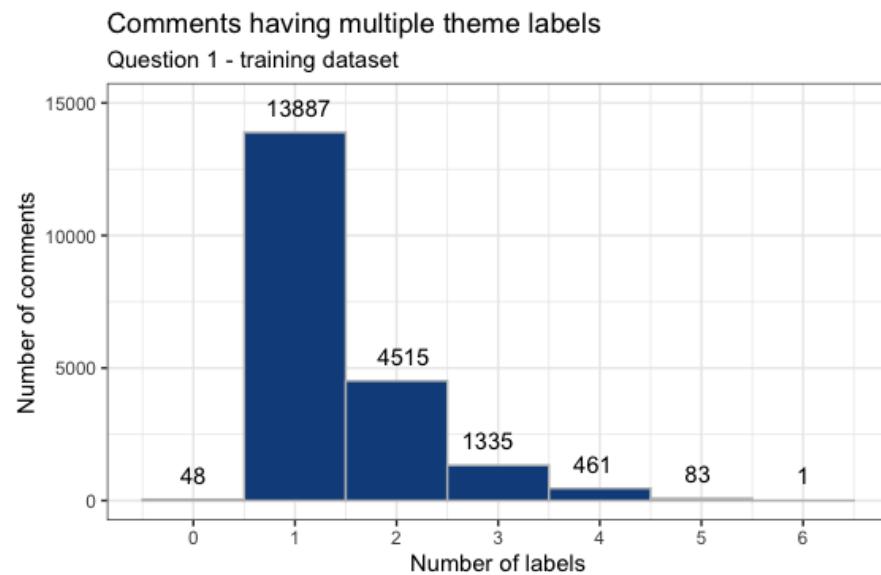
- Predicts label(s) for Question 1 and 2's main **themes**
- Predicts label(s) for Question 1's **subthemes**

2) Build an app for visualizations of text data:

- Identify and compare **common words** used for each question
- Identify **trends on concerns (Q1)** and **appreciations (Q2)** for BC ministries over the given years

Challenges with data

1) Sparsity

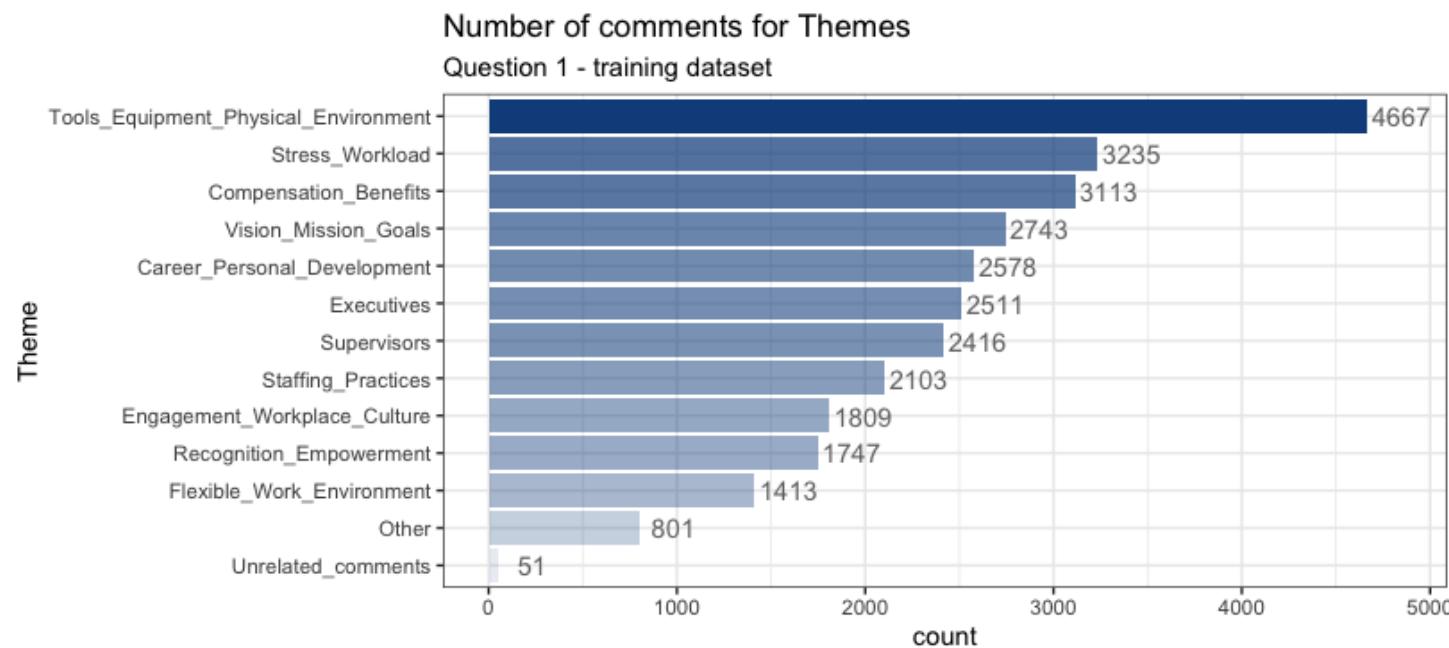


There are 12 themes and 62 subthemes that comments can be encoded into.

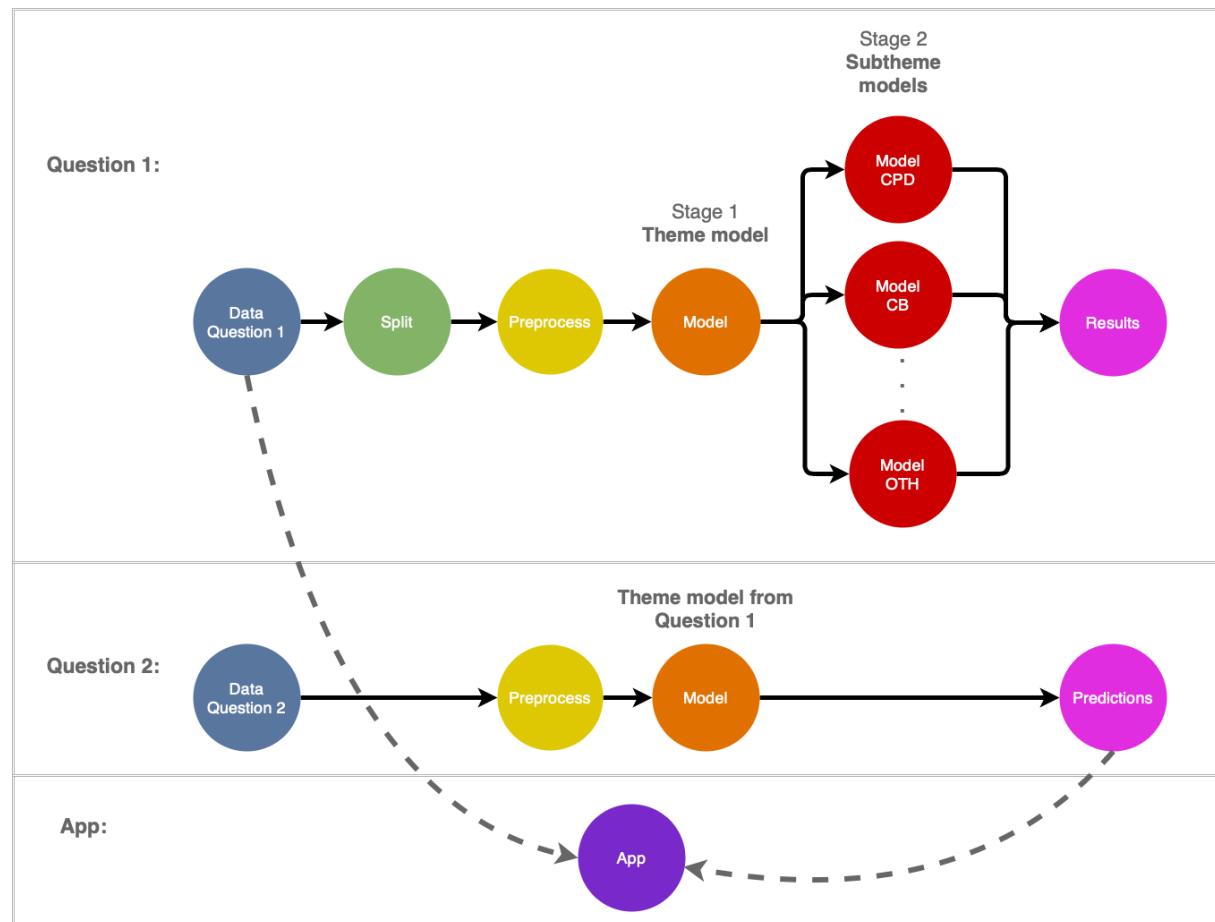
- Average number of labels per comment: Themes = ~1.4 , Subthemes = ~1.6

Challenges with data

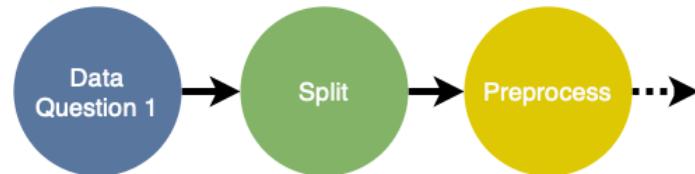
2) Class Imbalance



Text classification methodology



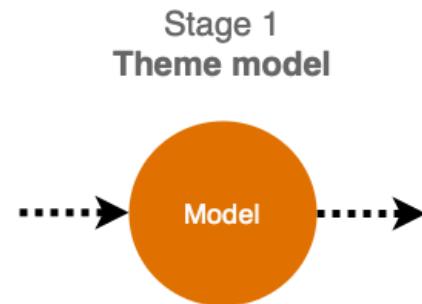
Data Split & Preprocess



- Raw -> 80% train, 20% test.
- Training -> 80% train, 20% validation
- removed **sensitive information** using **Named Entity Recognition (NER)** to remove person, organization, location, and geopolitical entity

Example comment to get flagged: "George and I love when the department gives us new coupons!"

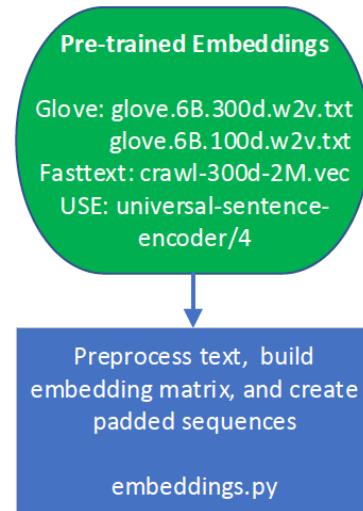
Modelling Techniques



- **Baseline models:** used **TF-IDF Vectorizer** and traditional machine learning classifiers (RandomForest, GaussianNB, etc)
 - best results with **LinearSVC**
- **Deep Learning models:** ran multiple models including **CNNs** and **sequential models** with pre-trained embeddings

Pre-Trained Embeddings

Fasttext, Glove, Universal Sentence Encoder

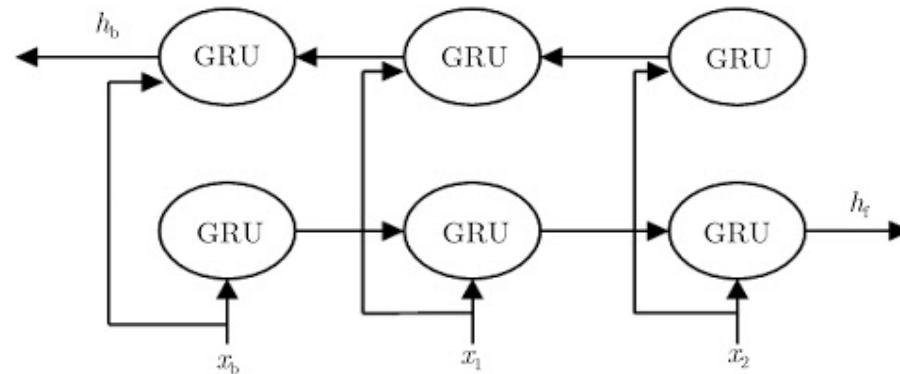


- **Fasttext:** trained on Common Crawl
- **Glove:** trained on Wikipedia and Gigaword (newswire text)
- **Universal Sentence Encoder:** trained on Wikipedia and Common Crawl
- Built embedding matrixes & transformed comments to padded sequential data to fit into embedding size.

Deep Learning Models

BiGRU

- GRUs: preserves order of words and has 2 gates (reset gate and update gate)
- Bidirectional GRUs: Uses sentence sequences from both left-to-right and right-to-left

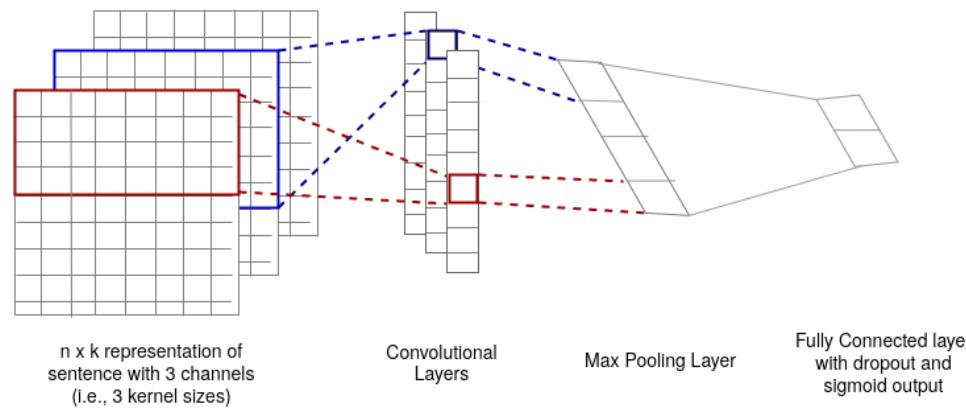


Source: [Bi-directional Gated Recurrent Unit](#)

Deep Learning Models

Multichannel CNNs

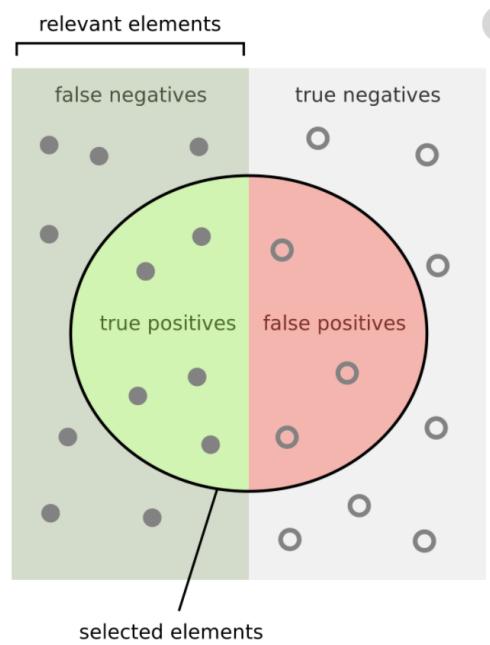
- Multiple versions of standard CNN models with different kernel sizes.
- Specifically, we have defined a model with 3 input channels for kernel sizes 4, 6 and 8



- n - number of words , k - dimension of the word embedding

How we measured success

Precision & Recall



How many selected items are relevant?

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

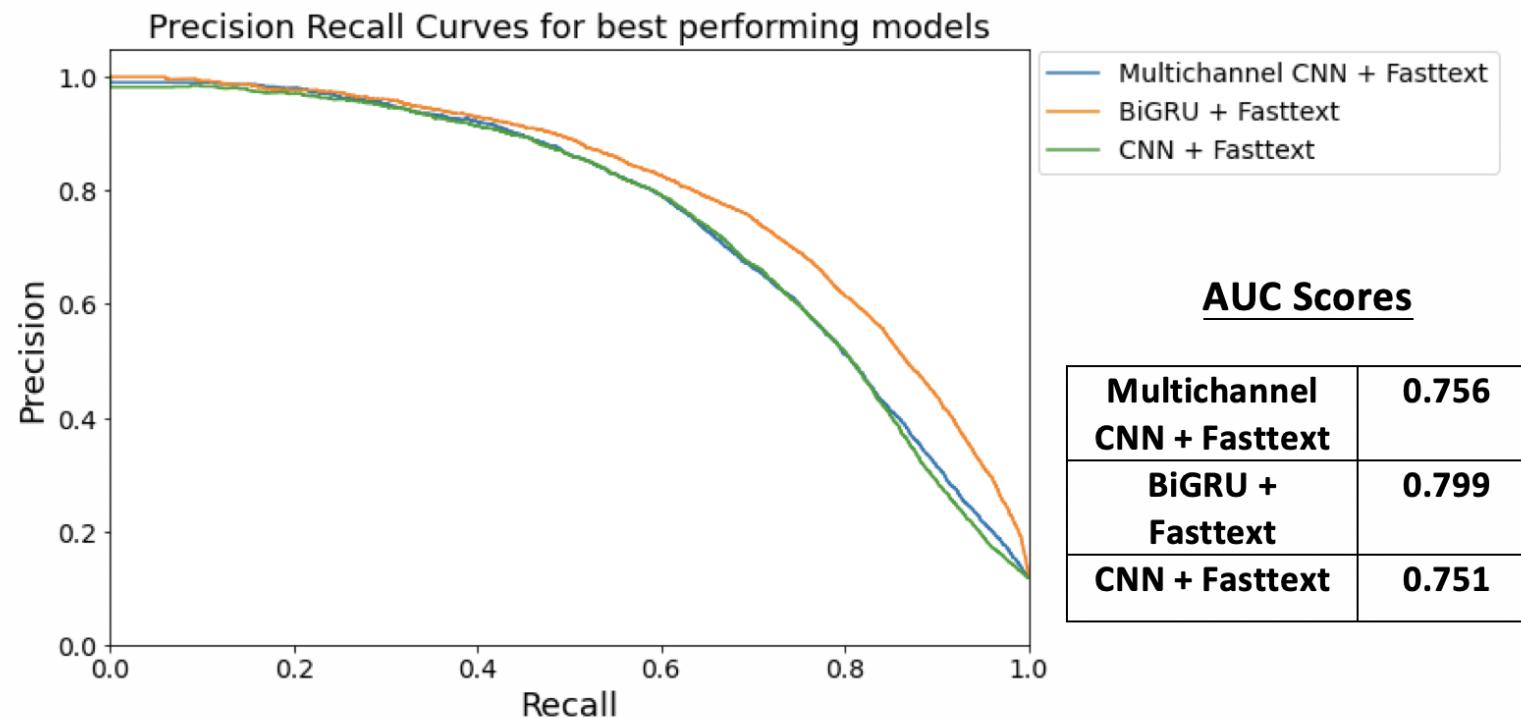
How many relevant items are selected?

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

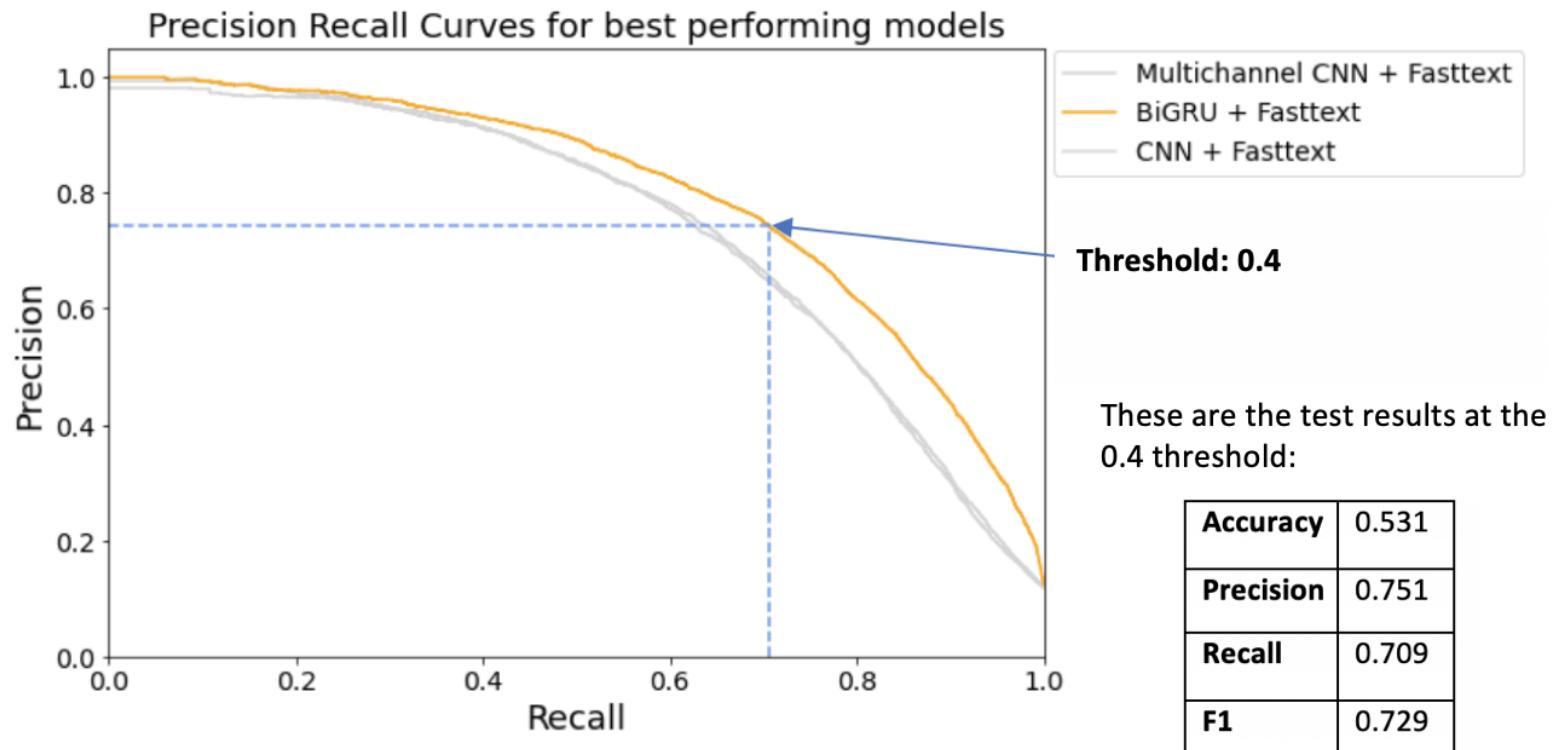
- **Precision Recall curve:** plotting precision vs recall at various threshold rates
- **Micro-average:** weighted average of the precision and recall

Source: [Precision and Recall](#)

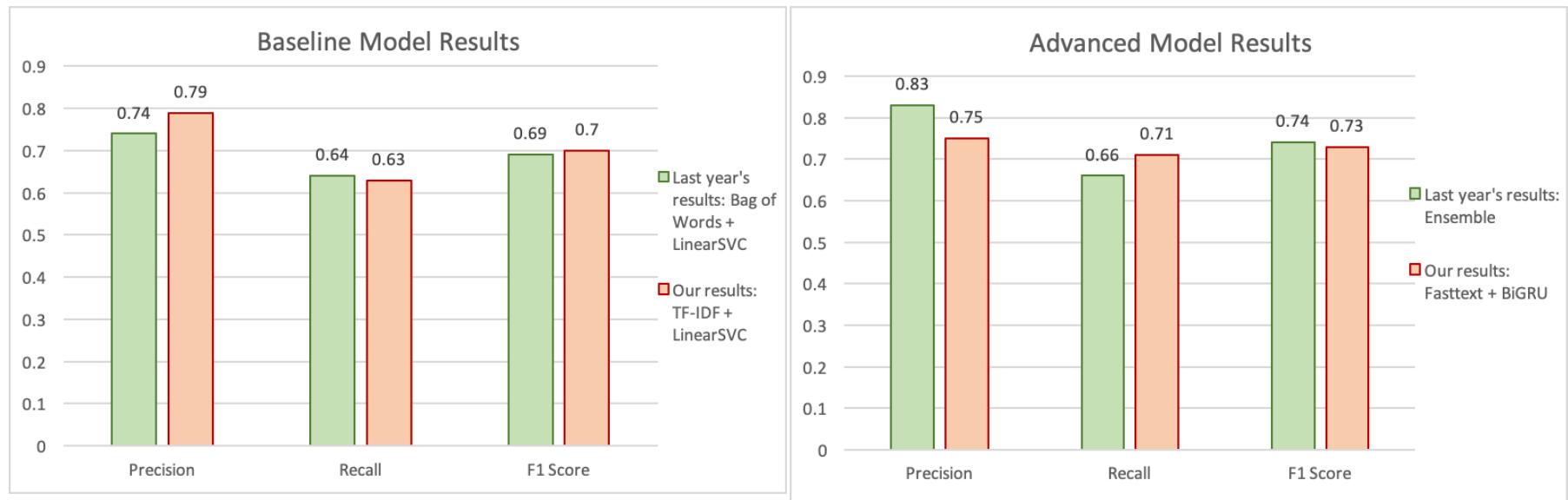
Precision Recall Curve for Q1 Theme Models



Advanced Model: Fasttext + BiGRU



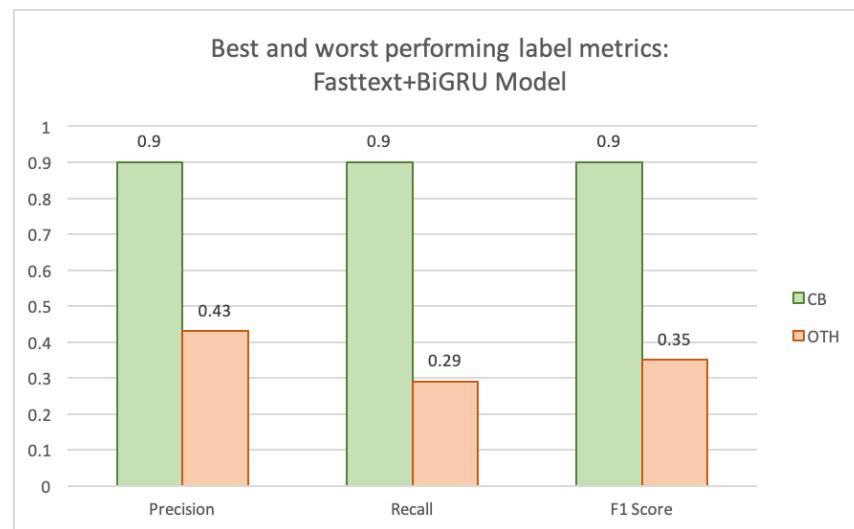
Results for Theme Labelling Models



Source: [BC Stats Capstone 2019-Final Report, by A. Quinton, A. Pearson, F. Nie](#)

Label Wise Results for Fasttext + BiGRU

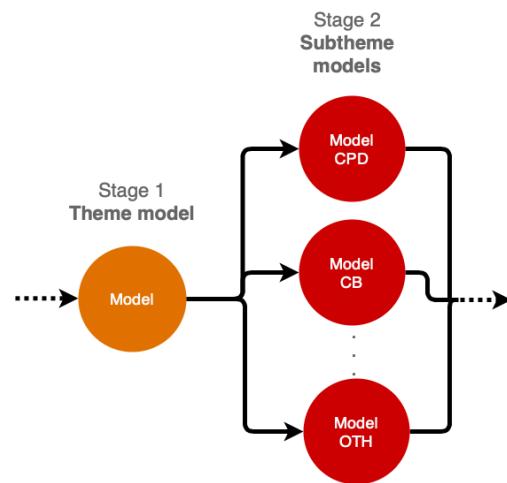
Predicting each theme



- Themes with high F1 scores (CB) can be encoded **automatically using the model**, while themes with lower scores (OTH) can be **manually verified** by BC Stats.
- Recommendation to use a **combination of machine learning and manual encoding**.
- Rest of the themes have the following ranges:
- **Precision: 0.69-0.92**
- **Recall: 0.51-0.85**

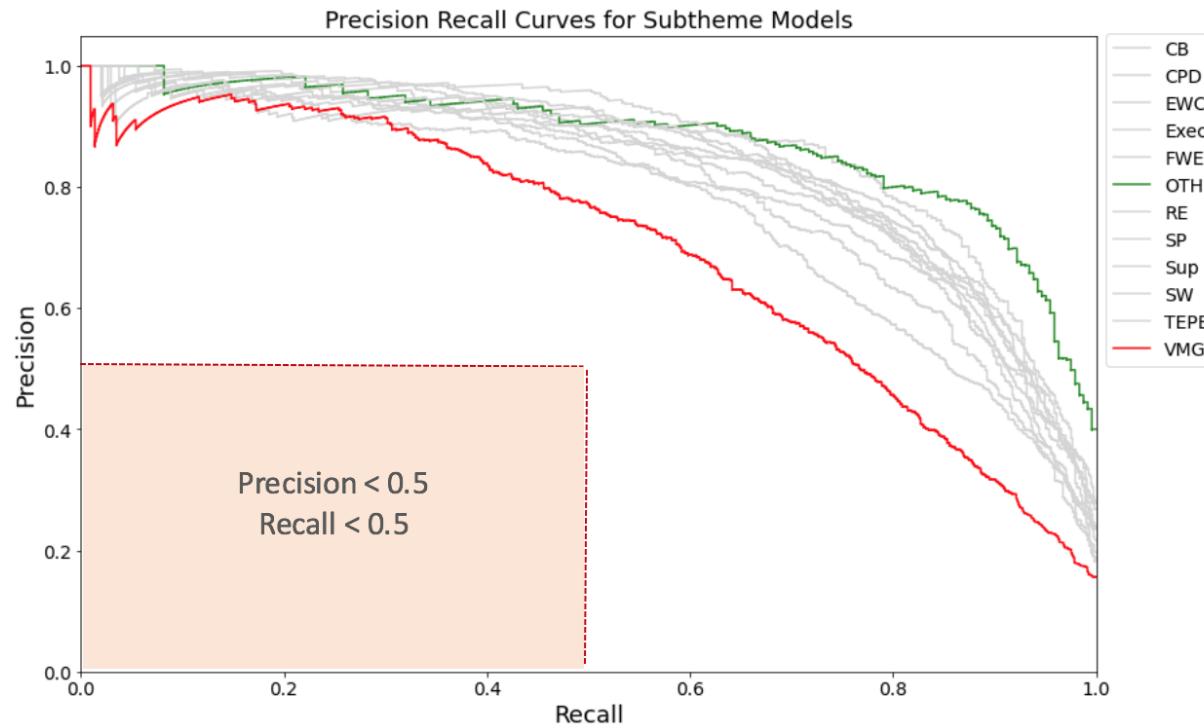
Building Subtheme Models

Hierarchical approach



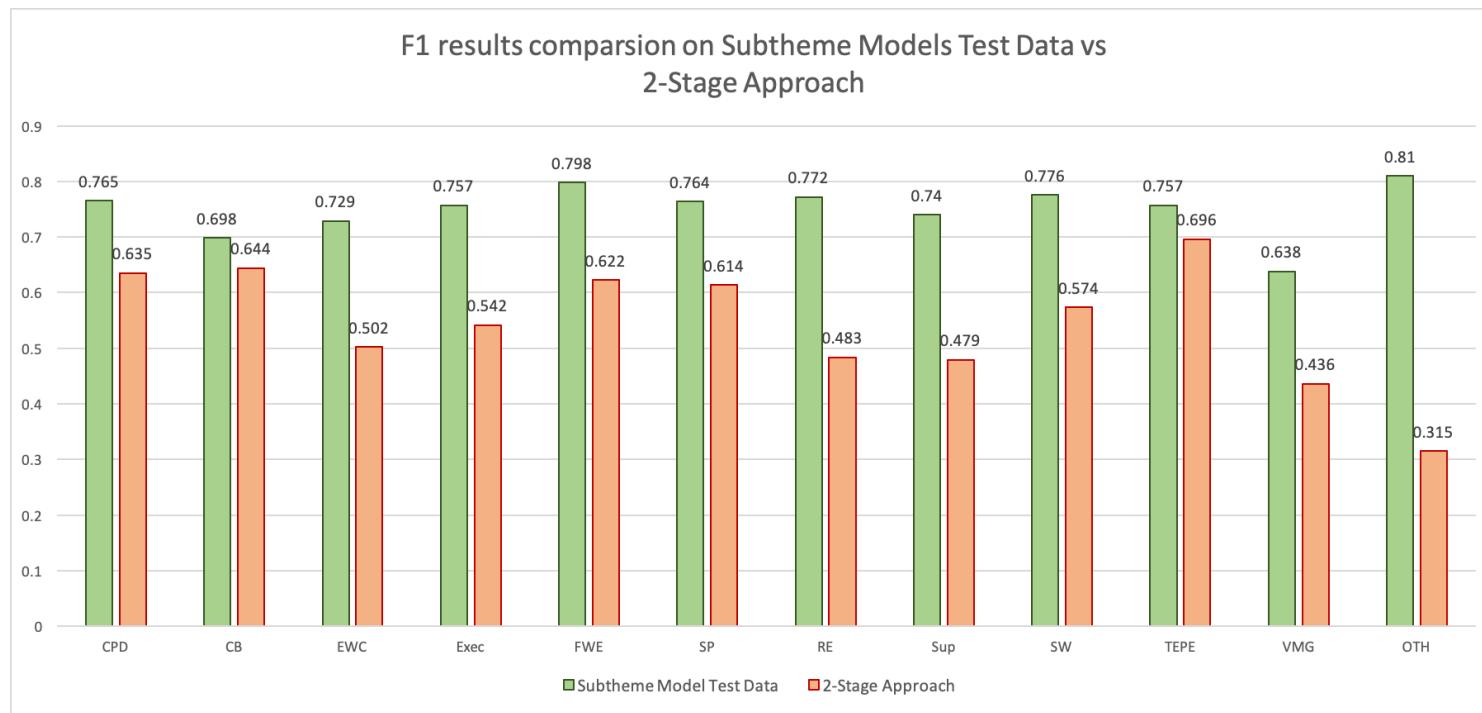
Subthemes are predicted based on the theme(s) our model has assigned to the comment.

Precision Recall Plot for Subtheme Models



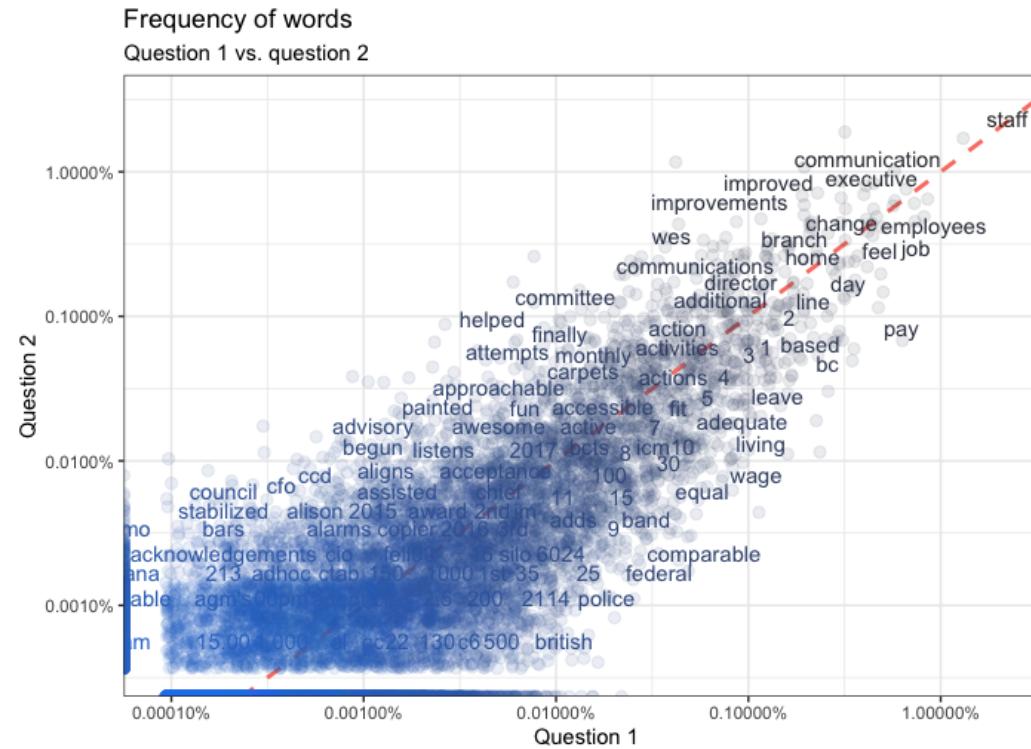
- The minimum desirable of both precision and recall values shared by BC Stats for labelling subthemes was 0.5.
- All subtheme models surpassed this threshold.

Subtheme models results



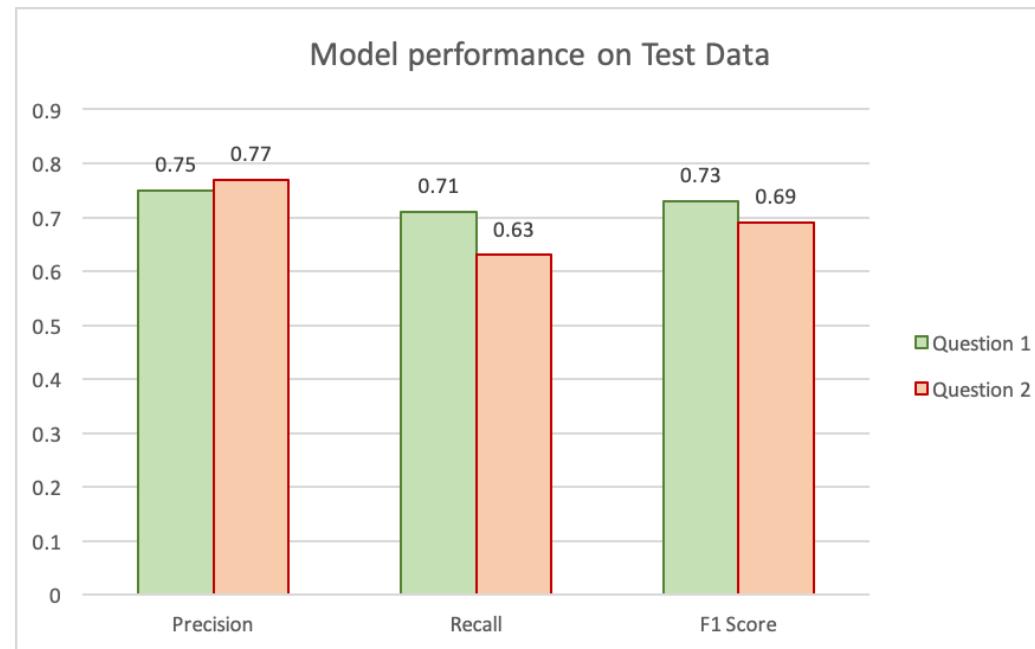
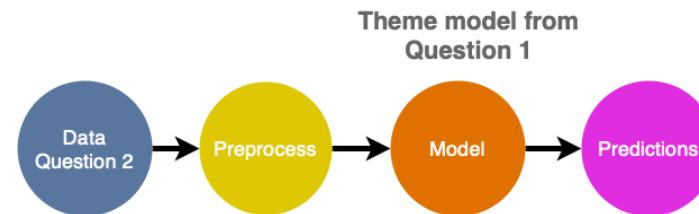
- $$F1 = 2 * \frac{precision * recall}{precision + recall}$$
- Increasing data for label OTH will improve results

Comparing Question 2 to Question 1

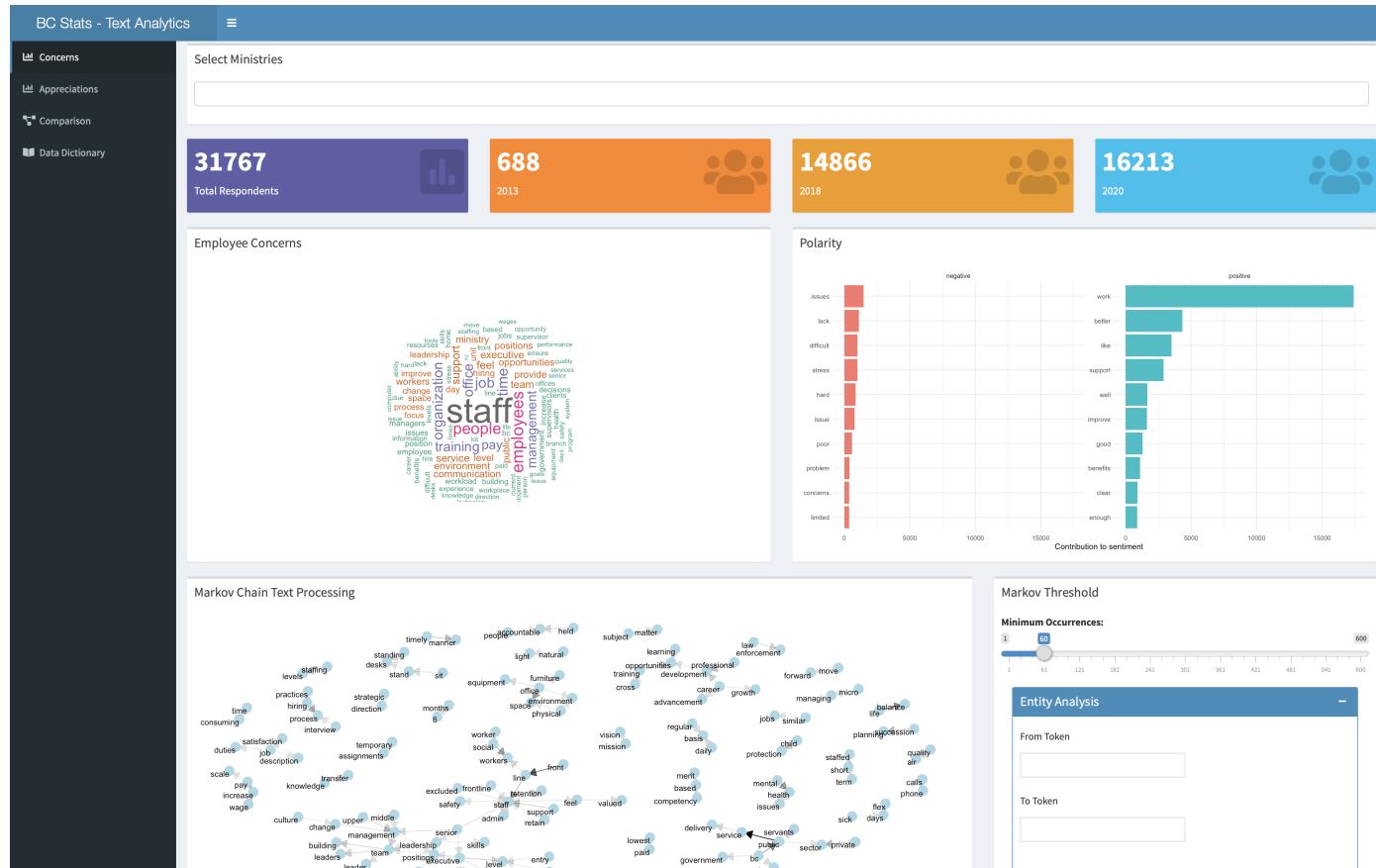


- Observed a **linear trend** in frequency of common words between Question 1 and Question 2.
 - Validated **using the themes from Question 1** to label comments from Question 2.

Question 2: Predicting Themes

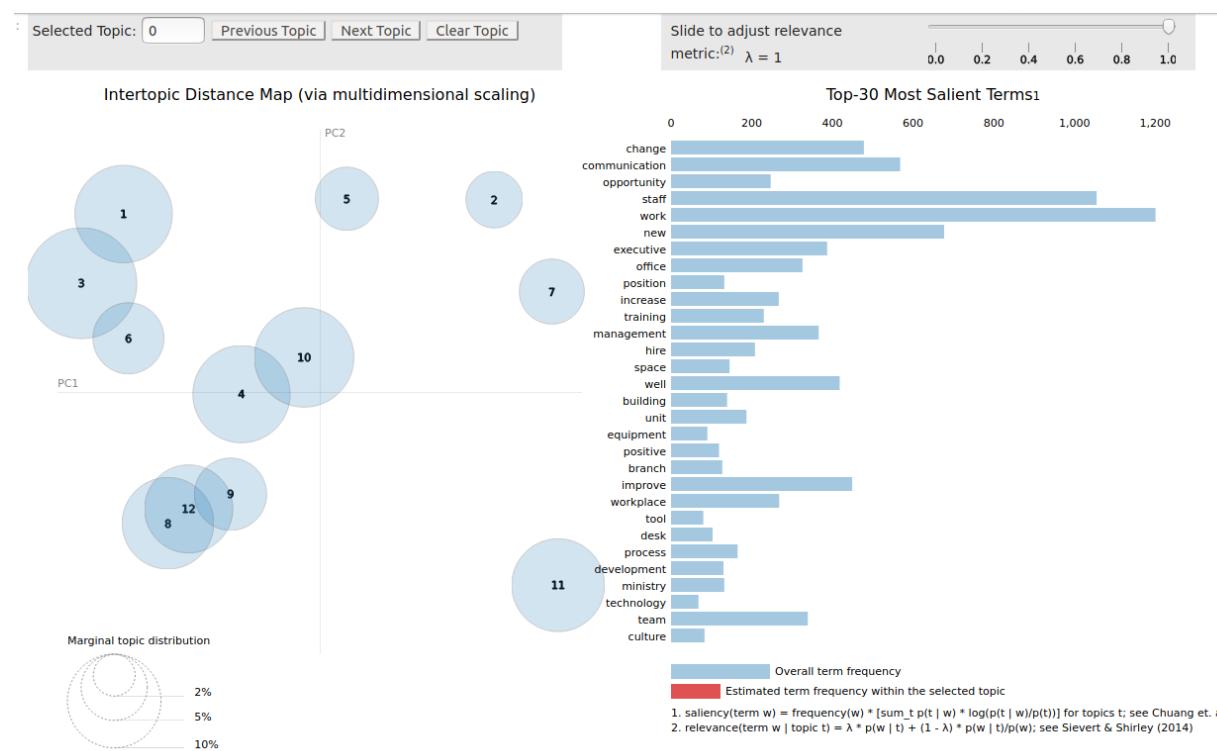


Dashboard



Methodologies that did not work

- **Overfitting** in CNNs and multi-channel CNNs
- **Universal Sentence Encoder** embeddings
- **Topic modelling** for Question 2 (too much overlap in words, ambiguity)



Recommendations & Conclusions

- Expected to observe better results with **more data**
 - will improve results for subthemes using hierarchical approach
- Use model to automate labelling on themes and subthemes with high precision and recall
 - **CB, TEPE, CPD, SP, and FWE** (for theme model)
 - manually encode or verify model's results on other comments

Thank you. Questions?



27/28

Appendix A: Theme Model threshold levels

Results on various threshold values using validation data

| Threshold | Accuracy | Precision | Recall | F1.measure |
|-----------|-------------------|-------------------|-------------------|------------------|
| 0.3 | 0.500491835530199 | 0.703502255240117 | 0.739403234802008 | 0.7210061182868 |
| 0.4 | 0.524099940979736 | 0.743642510657063 | 0.705382041271612 | 0.72400715563506 |
| 0.5 | 0.53393665158371 | 0.77415712211647 | 0.669129949804796 | 0.7178221524194 |
| 0.6 | 0.529018296281723 | 0.798985304408677 | 0.636781929726715 | 0.70872129112352 |
| 0.7 | 0.519771788313988 | 0.826897214217099 | 0.600111544896821 | 0.6954835582128 |
| 0.8 | 0.49990163289396 | 0.856865284974093 | 0.55340211935304 | 0.6724839037614 |
| 0.9 | 0.465276411567972 | 0.899328859060403 | 0.485778025655326 | 0.63081658518920 |