# NLP HW2

110550108 施柏江

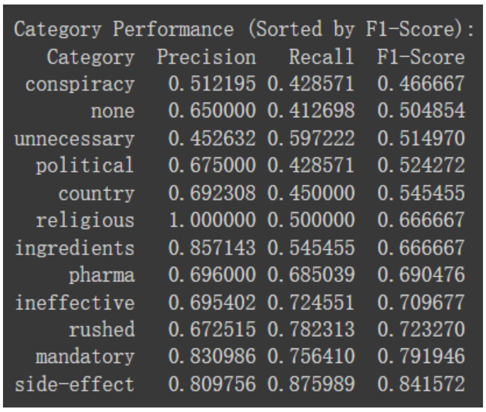
## 1. How did you do to preprocess your data from dataset?

I first loaded the dataset from JSON files. Then I tokenize each tweet using BERT's tokenizer, converting the text into input_ids and attention_mask, with padding or truncation to ensure a consistent length of 256 tokens. For labels, I create a one-hot encoded vector for each tweet, indicating the presence of specific concerns from the predefined label list. Finally, I wrap the tokenized tweets and label vectors into a custom dataset class to load the data in batches for training and evaluation.

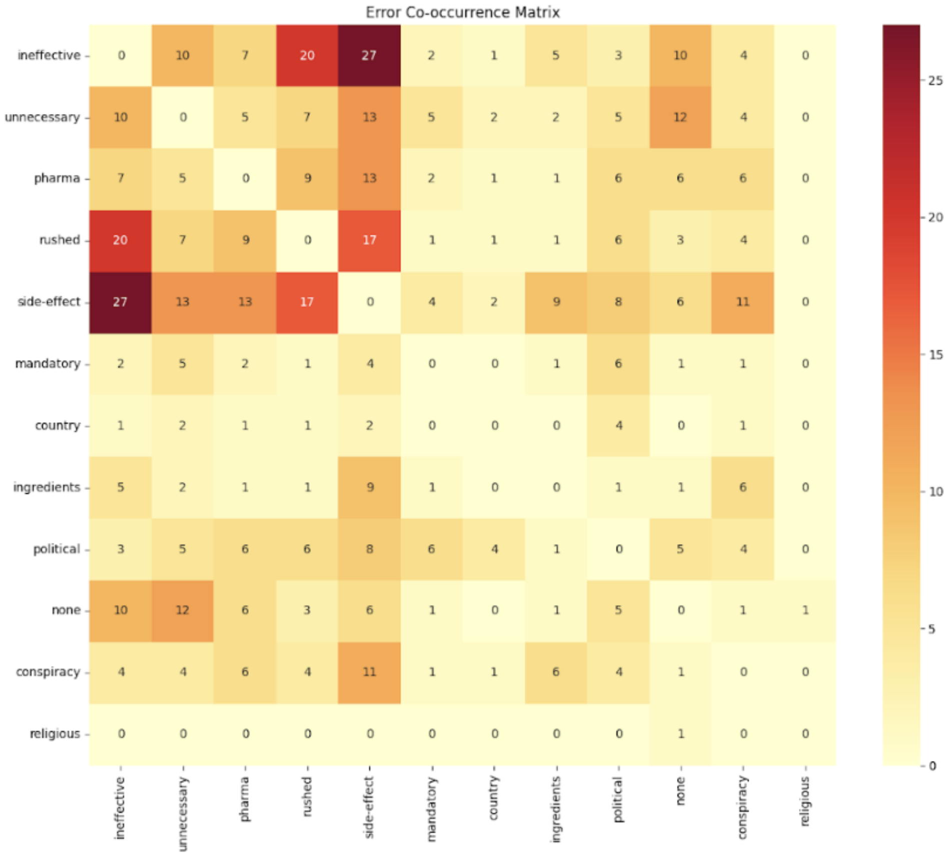
## 2. How were the model's hyperparameters chosen? Did you perform hyperparameter tuning? If so, what were the specific steps and results?

I experimented with different hyperparameters to optimize the model's performance. For the learning rate, I tested values of 5e-6, 1e-5, 2e-5, 5e-5, and 1e-4 to find a balance between training stability and convergence speed. I also experimented with the maximum sequence length, testing values of 64, 128, and 256, to capture the most relevant context from tweets while managing computational resources. For dropout rates, I tried 0, 0.1, and 0.3 to prevent overfitting by adding regularization. These adjustments allowed me to observe how the model performed under different configurations, and I ultimately decided to use a learning rate of 5e-5, a maximum sequence length of 256, and a dropout rate of 0.3 based on the experiments.

### 3. In your experimental results, which categories of concerns were the most difficult to predict? And which categories were these concerns most often misclassified as?



```
Category Performance (Sorted by F1-Score):
   Category  Precision    Recall  F1-Score
 conspiracy   0.512195  0.428571  0.466667
       none   0.650000  0.412698  0.504854
unnecessary   0.452632  0.597222  0.514970
  political   0.675000  0.428571  0.524272
    country   0.692308  0.450000  0.545455
  religious   1.000000  0.500000  0.666667
ingredients   0.857143  0.545455  0.666667
     pharma   0.696000  0.685039  0.690476
ineffective   0.695402  0.724551  0.709677
     rushed   0.672515  0.782313  0.723270
  mandatory   0.830986  0.756410  0.791946
side-effect   0.809756  0.875989  0.841572
```



Error Co-occurrence Matrix

| | ineffective | unnecessary | pharma | rushed | side-effect | mandatory | country | ingredients | political | none | conspiracy | religious |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ineffective | 0 | 10 | 7 | 20 | 27 | 2 | 1 | 5 | 3 | 10 | 4 | 0 |
| unnecessary | 10 | 0 | 5 | 7 | 13 | 5 | 2 | 2 | 5 | 12 | 4 | 0 |
| pharma | 7 | 5 | 0 | 9 | 13 | 2 | 1 | 1 | 6 | 6 | 6 | 0 |
| rushed | 20 | 7 | 9 | 0 | 17 | 1 | 1 | 1 | 6 | 3 | 4 | 0 |
| side-effect | 27 | 13 | 13 | 17 | 0 | 4 | 2 | 9 | 8 | 6 | 11 | 0 |
| mandatory | 2 | 5 | 2 | 1 | 4 | 0 | 0 | 1 | 6 | 1 | 1 | 0 |
| country | 1 | 2 | 1 | 1 | 2 | 0 | 0 | 0 | 4 | 0 | 1 | 0 |
| ingredients | 5 | 2 | 1 | 1 | 9 | 1 | 0 | 0 | 1 | 1 | 6 | 0 |
| political | 3 | 5 | 6 | 6 | 8 | 6 | 4 | 1 | 0 | 5 | 4 | 0 |
| none | 10 | 12 | 6 | 3 | 6 | 1 | 0 | 1 | 5 | 0 | 1 | 1 |
| conspiracy | 4 | 4 | 6 | 4 | 11 | 1 | 1 | 6 | 4 | 1 | 0 | 0 |
| religious | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

The categories "conspiracy," "none," and "unnecessary" were the most challenging to predict, as indicated by their relatively low F1-scores. Conspiracy was often misclassified as side-effect. None was frequently misclassified as unnecessary or ineffective. Unnecessary had misclassifications with side-effect or none.

4. **Building on the previous question, what methods have you tried in your experiment to improve the models' ability to more accurately identify the concerns expressed by users? Please describe both the successful and unsuccessful cases.**

I experimented with different learning rates, sequence lengths, and dropout rates. The final selection helped the model capture more context. This tuning slightly improved F1-scores across multiple categories.

I also attempted data augmentation through paraphrasing tweets, hoping to add diversity in how concerns were expressed. However, this method did not yield significant improvement. Paraphrased tweets occasionally altered the tone or meaning, leading to noisy inputs that confused the model during training.