COMP400 Report

Modeling COVID-19 news reports to predict public health interventions

Jingfu Zhang (Jeff, 260840033)[1]

[1]McGill University, supervised by Professor Yue Li

July, 2024

Abstract

Non-pharmaceutical interventions (NPIs), such as mandating mask wearing and limiting the scale of public gatherings, are frequently more effective in containing the spread of diseases than the conventional medical treatments. However, a thorough investigation of such interventions to fully assess their effectiveness demands long-term monitoring of any news reports that may pertain to the infectious disease in question. Given the absence of automated strategies for tackling this problem, currently the fulfilment of such tasks has heavily relied on the manual effort of the volunteers. This is both challenging to sustain or to scale. Inspired by the recently developed EpiTopics [1] model which enables the automation of such procedures by leveraging probabilistic modelling, we propose the NPIBert which is another improved method of extracting the NPIs from news articles. We pretrain the bidirectional encoder representations from transformers (BERT) on COVID-19 related corpora (the Aylien dataset) to uncover high-quality relationships between COVID-19-related concepts and then fine-tune the pretrained model on a smaller WHO dataset to predict NPI labels from news articles. The NPIBert has outperformed the various baseline models.

1 Introduction

Non-pharmaceutical measures are another key to the success in containing the COVID-19 pandemic. These interventions (often referred to as NPIs), ranging from mandating mask-wearing to enforcing social distancing, are frequently more influential than medical treatments based on anecdotal evidences. This is especially true at the onset of an emerging pandemic due to the absence of a specific remedy against it. However, monitoring these observational evidences often poses a huge challenge for researchers due to the fact that they are often concealed in the enormous volume of other unrelated news reports. Presently as we speak, the fulfillment of such a task has mainly relied on the manual efforts from volunteers across the world. Such a strategy is inevitably difficult to sustain or to scale. To tackle this problem, probabilistic modelling methods such as the EpiTopics model [1] have been introduced. Despite the already-impressive results delivered by this model, the prospects of leveraging the powerful language model BERT seems to point us in the direction of another major breakthrough in terms of prediction performance. The bi-directional

encoder representations from transformers (BERT) is capable of learning deep representation of text from news reports and public health related articles to uncover insights that are previously beyond our attainment. We aim to develop a deep learning framework that could facilitate policy evaluation and potentially guide evidence-based decisions through the usage of BERT. The proposed model is unprecedented in that the representational power of the BERT framework is yet to be explored in the area of NPI monitoring. Additionally, our proposed model should exploit the BERT model's ability to capture high-level information using unsupervised training alone. The bidirectional encoder from transformers' inherent unsupervised training tasks, featuring next-sentence prediction (NSP) and masked language modelling (MLM), enable us to model COVID-19 relevant information using massive sets of unlabelled data. This effectively reduces the demand for labelled data which can be limited in quantity and difficult to obtain. Instead, we focus our scarce labelled assets on the fine-tuning stage of our framework's training to deliver the most optimal results.

2 Related Work

Many characteristics of our framework can find their origin in several other state-of-art applications of neural networks.

2.1 BERT

The NLP community has witnessed a dramatic paradigm shift toward the pre-trained deep language representation model, which achieves the state of the art in question answering, sentiment classification, and similarity modeling. Bidirectional Encoder Representations from Transformers [2] (BERT; Devlin et al., 2019) represents one of the latest developments in this field of study. It has consistently outperformed its predecessors, ELMo [3] (Peters et al., 2018) and GPT [4] (Radford et al., 2018), by a wide margin on multiple NLP tasks. The training of the BERT model highlights two main stages. Firstly, the BERT model is pretrained on large sets of relevant text corpora. Then, the pretrained BERT model is being appended additional linear layers to undergo fine-tuning on labelled and task-specific data. Although not yet fine-tuned for the extraction of NPI labels, the models that belong to the BERT family (BERT, Alberta, Roberta, to name a few) has seen extensive action in many other classification tasks and achieved impressive state-of-art performances.

2.2 DocBERT

DocBERT [5] is the first ever application of BERT to document classification. DocBERT has achieved revolutionary performance gain over several previously popular baseline models with the language modeling ability of the BERT model. The success of DocBERT provides the theoretical foundation for our experiment in that introducing a full-connected layer over the final hidden state of the BERT model's output (i.e. the [CLS] token) is sufficient to adapt BERT to any familiar classification task.

2.3 ClinicalBERT

Hospital readmission negatively affects patients' quality of life and wastes money [6]. It has been estimated that the financial burden of readmission hangs at $17.9 billion and the fraction of avoidable admissions at 76% [6]. Accurately predicting readmission has clinical significance, as it may alleviate the stress experienced by those patients who might be forced to contend with readmission, improve efficiency and reduce the burden on intensive care unit doctors. The ClinicalBERT [6] model aimed to leverage machine learning strategies to model a continuous representation of the potentially sparse, high-dimensional information hid-

#	Model	Reuters		AAPD		IMDB		Yelp '14	
		Val. F_1	Test F_1	Val. F_1	Test F_1	Val. Acc.	Test Acc.	Val. Acc.	Test Acc.
1	LR	77.0	74.8	67.1	64.9	43.1	43.4	61.1	60.9
2	SVM	89.1	86.1	71.1	69.1	42.5	42.4	59.7	59.6
3	KimCNN Repl.	83.5 ±0.4	80.8 ±0.3	54.5 ±1.4	51.4 ±1.3	42.9 ±0.3	42.7 ±0.4	66.5 ±0.1	66.1 ±0.6
4	KimCNN Orig.	–	–	–	–	–	37.6[8]	–	61.0[8]
5	XML-CNN Repl.	88.8 ±0.5	86.2 ±0.3	70.2 ±0.7	68.7 ±0.4	–	–	–	–
6	HAN Repl.	87.6 ±0.5	85.2 ±0.6	70.2 ±0.2	68.0 ±0.6	51.8 ±0.3	51.2 ±0.3	68.2 ±0.1	67.9 ±0.1
7	HAN Orig.	–	–	–	–	–	49.4[3]	–	70.5[3]
8	SGM Orig.	82.5 ±0.4	78.8 ±0.9	–	71.0[2]	–	–	–	–
9	LSTM_{reg}	89.1 ±0.8	87.0 ±0.5	73.1 ±0.4	70.5 ±0.5	53.4 ±0.2	52.8 ±0.3	69.0 ±0.1	68.7 ±0.1
10	BERT_{base}	90.5	89.0	75.3	73.4	54.4	54.2	72.1	72.0
11	BERT_{large}	**92.3**	**90.7**	**76.6**	**75.2**	**56.0**	**55.6**	**72.6**	**72.5**
12	KD-LSTM_{reg}	91.0 ±0.2	88.9 ±0.2	75.4 ±0.2	72.9 ±0.3	54.5 ±0.1	53.7 ±0.3	69.7 ±0.1	69.4 ±0.1

Figure 1: Taking into account performance comparisons on various datasets, the DocBERT paper has established the BERT model as the optimal choice in document classification tasks. This serves as one of the major motivations for us to select the BERT model while constructing our deep learning framework

den in Electronic Health Records (EHRs) to uncover clinical insights and automate the readmission decision process. Given the fact that clinical notes can frequently be long and with their individual words interdependent on one another, earlier attempts to model clinical notes such as the bag-of-words assumptions and Word2Vec have encountered performance bottlenecks. The paper demonstrated that the bidirectional encoder is capable of capturing such long-range dependencies in question and therefore outperform the previous frameworks in terms of accuracy.

Facing similar challenges as in the paper where COVID-19 related new articles could often be sparse and high dimensional (and where long-range dependencies also need to be addressed), we decided that the methodology adapted by the ClinicalBERT paper could prove to be instrumental in fulfilling our task. This eventually led us to opt for the BERT model in selecting our framework's main component.

Various other methodologies presented in the paper also proved to be inspirational in the subsequent development of our model. One technique that deserves the spotlight is as follows. While computing the NPI

$$P(readmit = 1|h_{patient}) = \frac{P^n_{max} + P^n_{mean} n/c}{1 + n/c}$$

prediction, the clinicalBERT model splits a larger piece of text into smaller segments (due to the input size limit of BERT) and outputs a prediction outcome on each one of them. The aforementioned algorithm is used for binning the individual prediction results and consolidating them into one single prediction outcome. Since our model also utilizes segmentation of over-sized input, we discuss the feasibility and effectiveness of the above algorithm later in this report.

Another key aspect of the clinicalBERT paper that we deem enlightening is the pretraining technique. The ClinicalBERT paper has pointed out that the quality of the learned representations of text is highly dependent on the text that the model is pretrained on. [6] Having been trained entirely on the BooksCorpus and Wikipedia dataset,

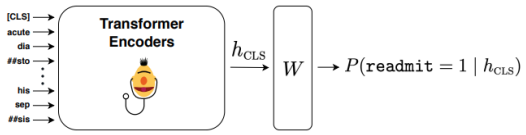

Figure 2: ClinicalBERT's structure which consists of the BERT model itself and an additional linear layer. The [CLS] token is passed between the two components for communication in training. The success of this architecture strongly motivated us to adopt a similar construct in designing our model. The only major difference between our NPIBert model and clinicalBERT is the difference in the sizes of the final output layers. We initialize 15 nodes in the final layer of the NPIBert network in alignment with the 15 harmonized classes of labels available.

the original, google-pretrained BERT model could face huge challenges when it comes to modeling the clinical text, since the clinical text could be full of jargon while having a distinct set of grammatical rules and syntax. The clinicalBERT paper suggested that pretraining the BERT model from scratch using a domain-specific corpus could resolve such a dilemma by allowing the BERT model to learn the domain-specific language from an early stage of training. Similarly, comprehending domain-specific text data (news reports in our case) is also a potential major hurdle for our NPIBert model. We therefore adapt the pretraining solution in the clinicalBERT paper to optimize the performance of our model.

2.4 EpiTopics

Being the first of its kind, the EpiTopics [1] model is a 3-stage machine learning framework for automating systematic NPI tracking. At stage-2 of the model's training process, a pretrained DETM is used to infer topic mixture from the set of 2000 NPI-labelled WHO documents as the input features for predicting NPI labels on each document, which presented theoretical support for us to construct a probabilistic model that could potentially automate the NPI extraction procedures. Moreover, the EpiTopics paper provided extensive description of the significance of the NPI extraction and the central role coordinated, organized, consistent NPI tracking could play in containing the spread of global scale pandemics. The paper overall provided us with strong incentive to explore the area of automated NPI extraction as well as using machine learning frameworks as our means to reach such an objective.

3 Datasets

3.1 WHO Dataset

The WHO Public Health and Social Measures (WHO-PHSM) dataset (https://www.who.int/emergencies/diseases/novel -coronavirus-2019/phsm) represents the collated data on COVID-19 public health and social measures as a result of merging public health and social measures data from several international trackers. The WHO dataset brings together a standardized, unified taxonomy which could be used in COVID-19 research. The content of interest within the WHO dataset are the WHO Summary data which represent news articles related to COVID-19 and WHO measure data which represent the NPI label associated with the aforementioned news article.

To preserve sufficient data points for each individual class of labels, the 44 different categories within the whole WHO taxonomy has been consolidated into 15 distinct classes of labels (For details, see [1]). We take into account both the prevalence of each class of labels and their conceptual similarities as we select the 15 final labels. The dataset is then split into training and testing sets according to the 8:2 ratio. It is also notable that we split the training

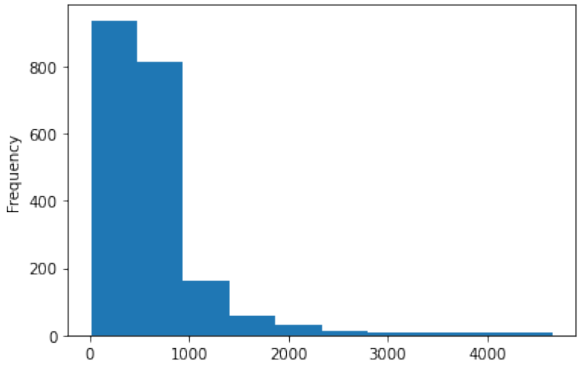

Figure 3: Distribution of WHO documents' lengths

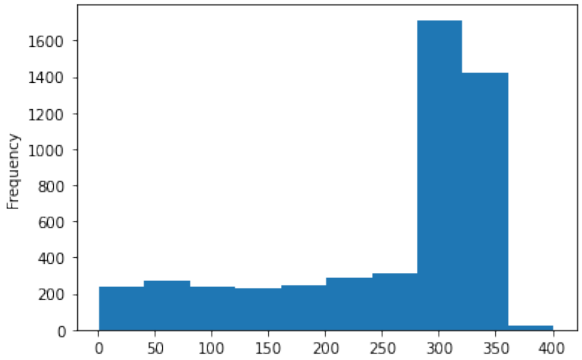

Figure 4: Distribution of split WHO documents' lengths

and testing sets at the document-level to prevent information leakage.

Here, we briefly introduce the WHO document splitting approach, which will be explained in greater detail in the Methodology section on prediction voting. Given the BERT model's input size limit of 512 tokens, we split the on-average 1000-word-long WHO documents into 300 to 350 sized chunks. See illustration for more information on document and chunk sizes.

3.2 AYLIEN dataset

Despite the outstanding quality of the expert-curated WHO dataset which is complete with accurate labelling for each COVID-related news article, the limited size of the WHO dataset would prohibit an neural network architecture as large as BERT to fully develop a meaningful and comprehensive representation of the style of language in the news articles. Such shortage of useful data for the pretraining stage prompts us to pursue another more suitable corpus in the News-Intelligence-Platform-curated dataset called AYLIEN. With its 1.2 million COVID-19 related news articles, the AYLIEN dataset is arguably a more favourable option for pretraining the BERT model from scratch.

4 Methodology

4.1 Data Preprocessing

Various data preprocessing techniques have been applied to the AYLIEN dataset before it is fed to the BERT pretraining pipeline. To start, non-ASCII characters, non-English words and white spaces have been pruned from the dataset. The dataset is then being passed through a sentencizer to be split into a list of sentences. Each of the sentence is subsequently being positioned on an individual line before we collect and store them in several large text files. In total, we tally 1.2 million documents in the AYLIEN dataset in addition to the 2049 articles in the WHO dataset.

4.2 Tokenization

Since the BERT model takes as input tokens instead of regular English words, tokenization the input sentences is necessary. We first train a randomly-initialized tokenizer using the AYLIEN corpus to create a specialized set of vocabulary and word-breaking rules based on the nature of the domain-specific language (news reports in our case). The choice of vocabulary is often based on the frequency of occurrence of the tokens.

In the case of masked language modeling, we need to provide the BERT model with both the masked input ids and its true labels. Following up from the previous step,

we pass each of the individual sentences in the prepared text files to the tokenizer to produce a set of input ids, attention masks and true labels. The labels correspond to the output of the tokenizer right after tokenization. The input ids repersent the labels but with around 30% of its tokens masked using a masking algorithm.

```
# create random array of floats
    ↪ with equal dims to input_ids
rand = torch.rand(input_ids.shape)
# mask random 15% where token is
    ↪ not 0 [PAD], 1 [CLS], or 2 [
    ↪ SEP]
mask_arr = (rand < .15) * (
    ↪ input_ids != 0) * (input_ids
    ↪ != 1) * (input_ids != 2)
# loop through each row in
    ↪ input_ids tensor (cannot do
    ↪ in parallel)
for i in range(input_ids.shape[0]):
    # get indices of mask positions
        ↪ from mask array
    selection = torch.flatten(
        ↪ mask_arr[i].nonzero()).
        ↪ tolist()
    # mask input_ids
    input_ids[i, selection] = 3 #
        ↪ our custom [MASK] token ==
        ↪  3
```

Finally, the attention masks is one means for the BERT model to easily differentiate paddings and the effective input ids that represent segments of actual words. This is given rise by the padding design of the tokenizer where input sequences less than the maximum input size are padded to that length.
In the case of next sentence prediction, the full sentences in the text files will be split into smaller chunks and shuffled with a small probability. Then, we group the input sequences into pairs. Each pair will receive a "1" label if it has not been shuffled or a "0" label if one sequence does not naturally succeed the other.

4.3 BERT Pretraining

As hinted in the above, the BERT model is pretrained on two major tasks: masked language modeling and next sentence prediction.
In masked language modeling, we masked a portion of the input tokens for the BERT model to evaluate a set of possible tokens that could be placed in the masked position. We then compare the predicted masked tokens with the original words that have been disguised to compute the loss.
In next sentence prediction, we supply the BERT model with a pair of input sequences and train the model to predict whether the two sequences are consecutive.
The pretraining of the Roberta model (i.e. another variant of the BERT model which is trained on a 10-fold larger dataset) is distinct from that of the BERT model in that next sentence prediction is deprecated. This is due to the assumption that having masked language modeling alone enhances the model's performance [7]. However, due to the fact that [CLS] tokens are never involved in the pretraining stage of Roberta but needed in the later stages of fine tuning, we opt for the BERT pipeline as our primary pretraining strategy.
Observing the Google research team's official directives, we start with Google's Wikipedia-pretrained BERT checkpoint and further pretrained our model for 1 million steps on 25G AYLIEN news data at batch size 16. The full traning process consumes 14 days on average. We also make available a distributed data parallel version for pretraining speed-ups (see methodology section).

4.4 NPIBert fine tuning overview

After pretraining the BERT model, we fine tune a "BERT + linear layer" combination on the NPI prediction task using the WHO dataset. Our framework computes for each class the probability that a WHO news ar-

ticle belongs to the class in question. The computation observes the below formula

$$P(belongsToClass = 1|h_{CLS}) = \sigma(Wh_{CLS})$$

where σ is the sigmoid function, [CLS] is the output of the BERT model corresponding to the classification token involved in next sentence prediction, and W is the parameter matrix.

4.5 Voting in fine tuning

By design, the BERT model has an upper limit of 512 for the maximum length of a input sequence of tokens. This restricts the lengths of text we are allowed to feed to the BERT model at once. Taking into account the fact that an average-length English word is usually split into 1 to 1.5 tokens, the number of English words one can input to the BERT model is between 300 to 350. This limitation can hamper our classification algorithm to some extent because longer input sequences need to be split into chunks, the training and evaluation will be run at the chunk-level rather than at the document-level. Although chunk-level training does not necessarily translate to bad performance (WHO documents are broken into only 3 chunks on average) because each chunk is frequently representative enough of the whole document, we sought an alternative strategy that could take advantage of aggregation by pooling the predictions results from multiple chunks before delivering a final prediction for the document. This naturally brings ClinicalBERT's voting algorithm into picture, where the chunk-level predictions are summarized via 2.3. The clinicalBERT paper states that a performance enhancement of 3 to 8% can be expected after adopting the strategy. Being aware of the potentially promising improvements, we modified the voting algorithm to a multi-class adaptation, where the sets of predictions logits from each chunk are summed together, fed to a softmax layer and then taken the argMax of. To our dismay, the success of the voting algorithm cannot be replicated in the multi-class scenario.

4.6 Distributed data parallel

Despite the remarkable boost in performance generated by the GPU's parallel computational capabilities, the sheer size of the BERT model induces prohibitively time-consuming training sessions (14 days on average for an epoch on a single rtx-2080 GPU). Such an enormous computational expense can be further complicated by unforeseen errors that occur during the training process, often nullifying a long-running training session and therefore bringing the project's progression to a crashing halt. Admittedly, exception handling algorithms could come in handy in such circumstances. However, unpredictable errors occur and it is often extremely challenging to complete an exhaustive error handling procedures that could tackle every possible problem that could arise. We therefore suggest another distinct way of working around heavy computational burdens —- Distributed Data Parallel, which has demonstrated its effectiveness over the course of our research project. To begin with, the DDP algorithm replicates the model of interest n times (where n is equal to the number of GPUs visible) and installs one for each processor unit. The complete set of training data is then evenly distributed across the GPUs. After consuming each batch, the model collects losses from each GPU to compute the overall gradient to be re-distributed to each model. The usage of DDP, on average, cuts the training time to just a fraction of its previous value which pronouncedly elevates the tractability of our model's training even in the face of massive datasets. This enabled us to accommodate more extensive experimentation as well as time to cover more of the detailed aspects in the optimization phase of our project. We make the

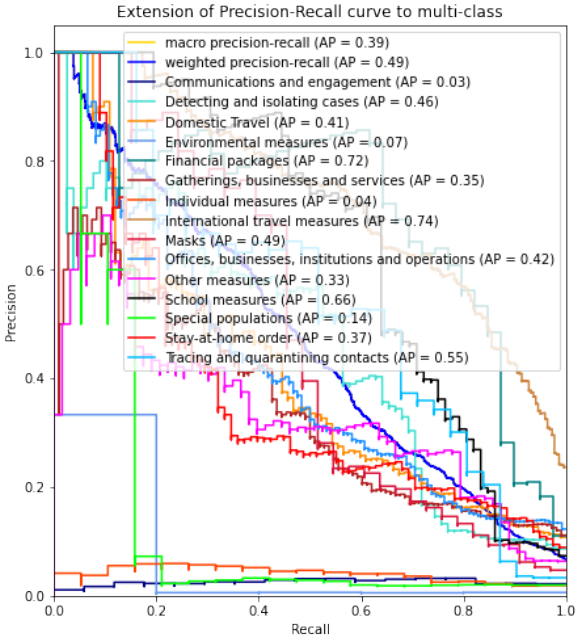

Figure 5: NPIBert reached a 25.8% increase in terms of macro AUPRC and a 19.5% gain in terms of weighted AUPRC when being contrasted against the baseline model

DDP version of our pretraining code available upon request.

5 Results

5.1 Overall result

We present the overall structure of the NPIBert model in (figure) and its detailed performance statistics in the following sections.
Through hyper-parameter tuning, we discovered that a batch size of 24, learning rate of $2e^{-5}$, drop out rate of 0.3 at the fine-tuning phase to have achieved the most performant model. To that end, we obtained a macro AUPRC score of 0.39 and a weighted AUPRC of 0.49 on the WHO dataset, which constitute a notable advancement from our baseline model. (see AUPRC figures 5)

5.2 Roberta v.s. BERT in classification tasks

(See table 1 2) In order to select the most suitable model for pretraining, we compare the performances of the BERT model and the Roberta model on evaluation tasks (masked language modeling and next sentence prediction) after pretraining them on the AYLIEN dataset. We concluded that the BERT model has been the better option out of the two for the WHO document classification task.

5.3 AYLIEN v.s. Wikipedia v.s. AYLIEN & Wikipedia

(See table 3) To construct our NPIBert model, we had 3 options at hand: using the BERT parameters from the google-pretrained checkpoint (that has been exposed to solely Wikipedia corpus), using the BERT model pretrained from scratch on the AYLIEN data and finally using the BERT model that is pretrained on both AYLIEN data and Wikipedia corpus. To guide our decision, we conducted a set of experiment to test out each model's performance and measured the performance metrics on NPI prediction.

5.4 NPI-wise performance comparisons v.s. baseline

In this section, we compare the prediction outcomes between the NPIBert model and the baseline model on each individual NPI. We conclude that although NPIBert is able to achieve better overall performance, the baseline mdoel still holds advantage in various categories of NPI prediction.

model	masked language modeling	next sentence prediction
BERT pretrained on AYLIEN +16G_Wikipedia	0.738	0.993
(Roberta) BERT pretrained on AYLIEN+16G_Wikipedia +144G_additional_data without NSP	0.604	*not supported by Roberta

Table 1: BERT achieves better performance on the native evaluation tasks than Roberta after both models have been pretrained on the AYLIEN dataset. The pretraining of BERT on AYLIEN dataset is based on the BERT-base checkpoint provided by Google Research. The pretraining of Roberta on AYLIEN dataset is based on the Roberta-base checkpoint provided by the Huggingface machine learning library.

model	macro AUPRC
BERT pretrained on AYLIEN +16G_Wikipedia; then fine-tuned and evaluated on WHO	0.39
(Roberta) BERT pretrained on AYLIEN+16G_Wikipedia +144G_additional_data without NSP; then fine-tuned and evaluated on WHO	0.28

Table 2: Fine-tuned BERT outperforms fine-tuned Roberta as well in NPI label predictions

model	macro AUPRC
BERT model pretrained from scratch on AYLIEN data; then fine-tuned and evaluated on WHO data	0.33
BERT model pretrained on Wikipedia corpus; then fine-tuned and evaluated on WHO data	0.30
BERT model pretrained on Wikipedia + AYLIEN data; then fine-tuned and evaluated on WHO data	0.39

Table 3: BERT model pretrained on both Wikipedia data and AYLIEN data achieved the best performance

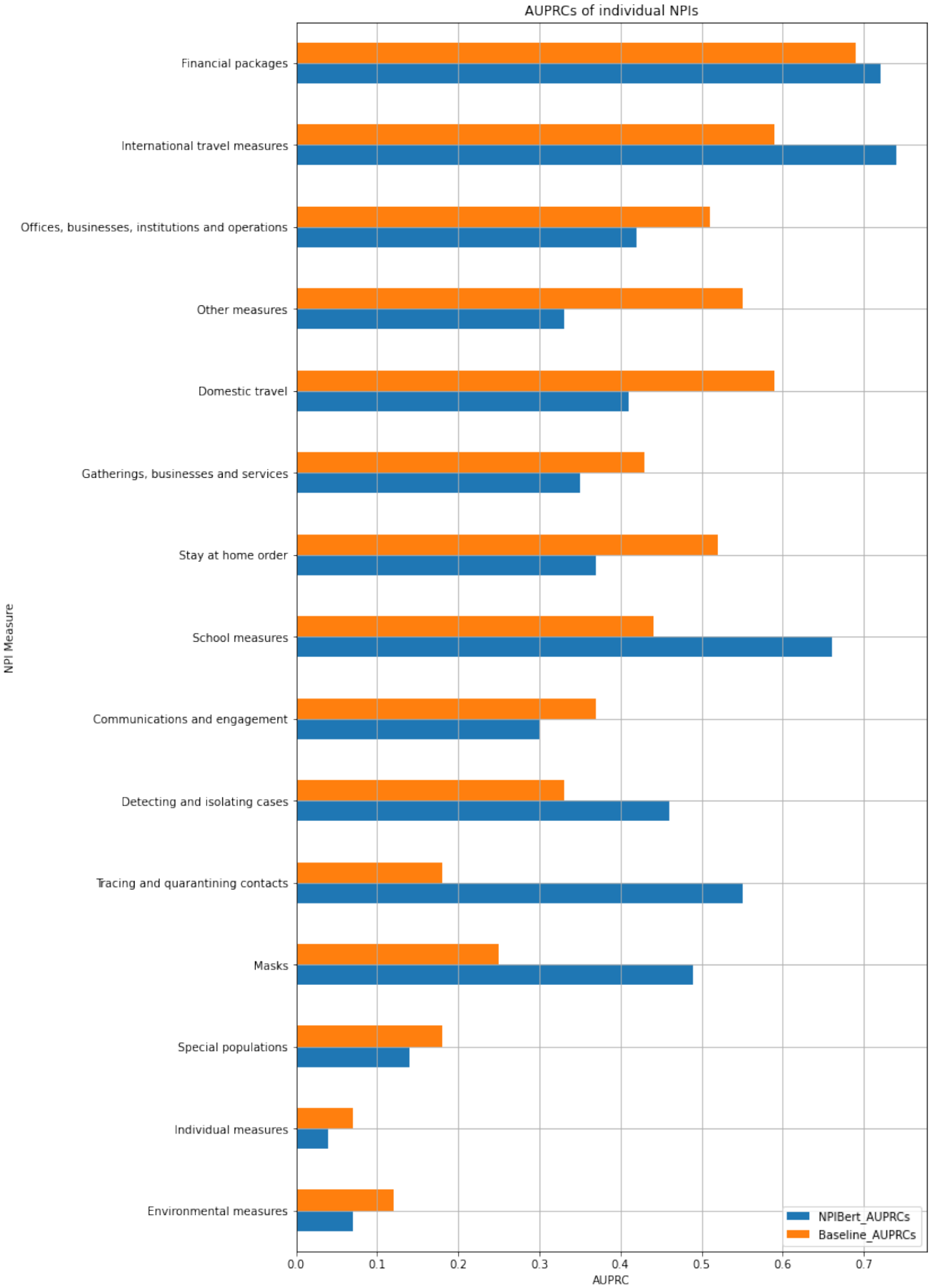

Figure 6: In additional to better overall performance, NPIBert achieves better performance in various individual NPI prediction tasks when being compared against the baseline Epitopics model on a NPI-by-NPI basis

6 Discussion and conclusion

6.1 Roberta v.s. BERT in classification tasks

As being demonstrated in the Results section, BERT brings about stronger performance not only in the native MLM evaluation tasks, but also in the NPI prediction tasks. As a first thought, this could be a phenomenon that is surprising to observe because Roberta is trained on 144G more data compared to BERT. Following the theory that more training data, in general, leads to better performance, it seems natural to conjecture that Roberta is more likely to obtain greater success than BERT. However, due to the fact that [CLS] token is passed between BERT or Roberta to the final linear layer and that [CLS] is never involved in Roberta's training (Roberta is only trained with masked language modeling while BERT is trained with both MLM and next sentence prediction using the [CLS] token), the final layer of the BERT model is able to extract a more comprehensive representation of the input sequence from the [CLS] token it receives. We infer that the [CLS] token's participation in BERT's native next sentence prediction training played a major role in boosting the BERT's model's performance.

6.2 AYLIEN v.s. Wikipedia v.s. AYLIEN & Wikipedia

Once BERT has been selected as the primary component of our framework, we test the impact on performance different training datasets have. As seen in the previous Results section, while having the domain-specific data in AYLIEN significantly aids prediction accuracy, we also observe that adding more general-purpose data can further improve the model's performance in the classification tasks. And we resolved to base our model on a BERT framework that is trained on both Wikipedia and AYLIEN data for optimization.

6.3 Possible directions for future exploration

We discussed in the Methodology section that the voting algorithm fell short of its expectation, adversely affecting the prediction accuracy instead of improving it. This, however, could be linked to the absence of a similar voting strategy in the loss computation stage of training. We strongly favour this direction as a possible way of further improving the framework's performance from both a theoretical and a practical standpoint.

Acknowledgements

Many thanks to Bruce Wen, Professor Yue Li and other fellow graduate and undergraduate researchers at McGill's Li Lab for their assistance and unwavering support over the course of the COMP400 project.

References

[1] Zhi Wen, Guido Powell, Imane Chafi, David Buckeridge, and Yue Li. Inferring global-scale temporal latent topics from news reports to predict public health interventions for covid-19. *medRxiv*, 2021.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[3] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018.

[4] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

[5] Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. Docbert: BERT for document classification. *CoRR*, abs/1904.08398, 2019.

[6] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *CoRR*, abs/1904.05342, 2019.

[7] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.