STAT 946 Deep Learning

Project Title (COVID-19):

Analysis of COVID-19 Measures Impacts on the Trend of New Confirmed Cases Using Deep Learning Models

Group Members:

Surname, First Name	Student ID	Department
Mohammad Zarei	20858700	Civil & Environment Engineering
Alireza Vezvaei	20896050	Mathematics, Computer Science
Mohammad Dehghan	20924172	Mathematics, Computer Science
Arash Mollajafari	20777740	Mathematics, Computer Science

Our most significant contributions are:

1. Two labeling method (binary and continuous) are proposed for quantifying policy effectiveness on COVID-19 new cases trend
2. A classification MLP and a regression MLP are developed based on the proposed labeling methods in order to predict and compare the effectiveness of policies
3. A time series forecasting model has been developed using a recent method called Temporal Fusion Transformer which can estimate the trend of new cases considering historical trend and applied policies.

STAT 946 Deep Learning

Project Title (COVID-19): Analysis of COVID-19 Measures Impacts on New Cases Trend Using Deep Learning Models

Group Members:
Mohammad Zarei, Alireza Vezvaei, Mohammad Dehghan , Arash Mollajafari

Abstract

In this work, we propose three different approaches that can be utilized to model, analyze, and compare the effectiveness of various COVID-19 policies/regulations. In the first approach, we have used a labeling technique and trained a model to predict whether a policy will be effective (label 1) or not (label 0) given policy information and new cases history. In the second approach, instead of having a binary label, an *Efficacy* value has been defined for each policy which quantifies a given policy effectiveness. Finally, in the third approach, a recent model called Temporal Fusion Transformer (TFT) has been employed to build a time-series forecasting model which is able to predict the trend of COVID-19 new cases given historical data and applied regulations.

1 Datasets

There are two available data sets including the cumulative daily number of confirmed COVID-19 new cases for 201 countries from 1/22/2020 to 2/27/2021, and all the policies implemented by governments worldwide in response to the COVID-19 pandemic (23,923 measures for 193 countries) [1]. Cumulative number of confirmed cases are aggregated for all provinces of those few countries that have province-level data. Here are some important features in governments policies data set:

- *LOG_TYPE*: shows whether a measure is introduced/extended (81%) or phased out (19%).
- *CATEGORY*: shows the major category of each policy including Public health measures (33.2%), Social distancing (23.2%), Movement restrictions (21.6%), Governance and socio-economic measures (18.2%), Lockdown (3.7%), Humanitarian exemption (0.1%)
- *MEASURE*: shows the minor category of each policy including 35 items which are summarised in Table 4 in the appendix.
- *COMMENTS*: A brief description of implemented policy (there are 124 (0.52 %) policies without comments)

We have used three different approaches to develop deep models which can relate the regulations/policies to the trend of new cases of COVID-19; binary classification, regression, and time series forecasting model. In the time-series forecasting model, we have also utilized [6], which is a dataset of official holidays in 67 countries in 2020, to improve forecasting of future COVID-19 cases.

2 Binary classification model

In this approach, the goal is to develop a model that can predict the effectiveness of a given regulation/policy on the basis of the new cases trend. For labeling the polices, a binary classification labelling technique from [3] is used. Two periods are defined to study the impact of each policy. N_{P1} is total new confirmed cases during the 14-day period before the date of policy implementation (base on the fact that COVID-19 incubation period is up to 14 days), and N_{P2} is total new confirmed cases during the 14-day period after a 5-day delay from implementation date. The label for each policy is calculated using the following equation:

$$\text{label} = \begin{cases} 1 \text{ (effective), if } \ N_{P1} > N_{P2} \\ 0 \text{ (not effective), if } \ N_{P1} \leq N_{P2} \end{cases} \tag{1}$$

	Accuracy (%)
Naïve Bayes (with comments)	36.69
Naïve Bayes (without comments)	36.74
Classification MLP (with comments)	64.57
Classification MLP (without comments)	66.35

Table 1: Classification MLP vs NB results

After excluding the policies with no comments and null labels (e.g. policies with not enough data for two defined periods), 21317 policies have remained. Table 5 in Section 6.2 shows the label counts for each measure category. Based on those results, for the case of introduction/extension of a measure only about 30% of times a policy considered as effective, and in case of measure phase-out this is about 50%. This observation supports the fact that the trend of new cases is a result of many other causes other than government policies such as people behavior. We have prepared the following features as the model inputs:

- One-hot encoded "Log_type" (2 features)
- One-hot encoded "Category" (5 features)
- One-hot encoded "Measure" (35 features)
- DeBerta [2] embedding of comment reduced by PCA method (20 features)
- 14-day history of daily new cases (14 features)

All features have been scaled using Min-Max scaling method, and the final data is then split into train, test and validation sets with 0.8, 0.1, 0.1 ratio. Using the train set, a multi layer perceptron (MLP) model with 4 hidden layers (each hidden layer has the size of 64 with a SELU activation function [4]) is trained. Weighted data loader is used to address the unbalanced labels. The hyper-parameters and settings for training process are presented in Section 6.1.

A naive Bayes model is also developed as a baseline model. In the similar work [3], the naive Bayes model shows the best performance in most of the experiments. The performance metrics for the test set for each model are presented in Table 1.

Although the MLP model outperform the NB model, the performance is still very low as it was expected. One potential solution to improve the results can be focusing on developing a time series forecasting model. In this part, we ignored the the time dependency of data for simplicity.

In another experiment, we have excluded the embedded comments from the feature vector to see how it affects the model performance. Based on the results in Table 1, we can see that this exclusion has minimal impact on both models' performances.

3 Regression model

In the regression approach, we have used the similar features and model architecture as previous section. But instead of predicting a binary label for each policy, the model is been trained to predict the amount of effectiveness of each policy. This new target value called *Efficacy* has been defined as follows:

$$\textit{Efficacy} = \frac{N_{P1} - N_{P2}}{AVG(N_{P1}, N_{P2})} \tag{2}$$

In this definition, an effective policy will have large positive *Efficacy*, and the *Efficacy* values are between 2 and -2 (Figure 1. The pre-processed dataset with the size of 21291 is then split into train/test/validation sets with the ratio of 0.8/0.1/0.1. Two deep regression model (with/without comments embedding) are been trained using the train set (hyper parameters/settings are available in Section 6.1):

A simple linear regression model is also fitted to the data as a base line model. The performance metrics for each model over test data is presented in Table 2. The results indicate that the regression MLP model significantly outperforms LR model, and its performance is almost independent from comments embedding.

In order to compare the effectiveness of 35 measures in Table 4, we have prepared 35 test sets for each of the measures. In other words, in each of these test sets all features are same (e.g. 14-day history) except the applied measure. Based on the average of model predictions for *Efficacy* values in each of these test sets, we can roughly understand which measures seems to be more effective. Note that for this analysis we ignored comment embedding as it has minimal impact on model performance.

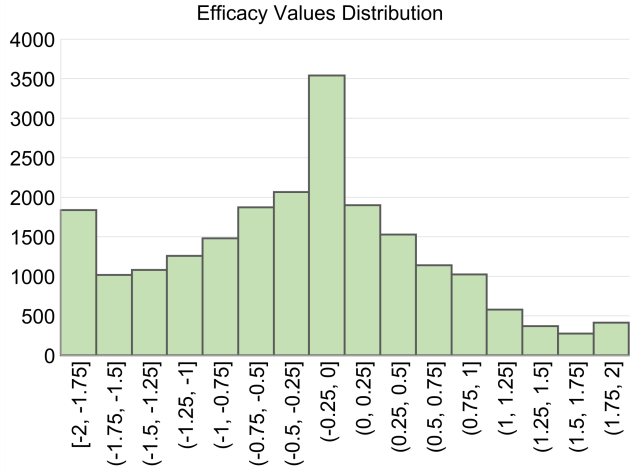

Figure 1: Distribution of Efficacy values for the full data set

	MSE	**MAE**	R^2 **score**
Linear Regression (with comments)	0.793	0.703	0.072
Linear Regression (without comments)	0.795	0.706	0.070
Regression MLP (with comments)	0.681	0.633	0.203
Regression MLP (without comments)	0.651	0.617	0.239

Table 2: Regression MLP vs LR model results

After scaling the averages of predicted *Efficacy* values of each test set over 35 measures using Min-Max method, a relative effectiveness factor is calculated for each measure that are shown in Figure 2. Based on this factor, top four most effective measures are respectively (1) Partial lockdown, (2) Psychological assistance and medical social work, (3) State of emergency declared, (4) Lockdown of refugee/idp camps or other minorities. On the other hand, General recommendations, Schools closure and Amendments to funeral and burial regulations are among the least effective policies.

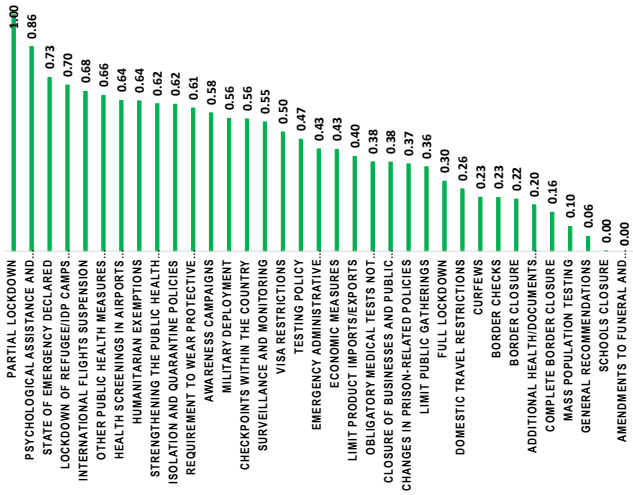

Figure 2: Min-Max scaled effectiveness factor for each measure

4 Time-series forecasting model

One major limitation of the aforementioned approaches in previous sections is that they separately associate the variations in COVID cases with the policies implemented in a specific time period before the variation. However, these changes in the number of reported confirmed cases are the consequence of coinciding numerous policies and tens of other unseen causes, such as people's adherence to the governments' regulations, virus mutations, or other variables related to the spread of COVID-19. Thus, it is extremely challenging to distinguish the effect of each cause from the others and predict the degree in which each of the causes contribute to the final outcome of new COVID-19 cases trend. Therefore, the underlying assumption in the previous section that we can equally relate the variations in confirmed cases to each policy is pretty naive and simplistic.

Another major drawback of previous methods is that they disregard the temporal nature of the problem. While we aim to discover the impact of governments' policies on new confirmed cases over time and forecast the COVID cases time-series in the following weeks, the proposed models simplify the task by assigning a label or score to each policy by comparing the number of cases in two 14-day periods before and after the implementation of the policy. Since the underlying temporal logic may be way more complex, we would lose a great deal of information by converting the time series to a simple classification or regression problem.

To address the aforementioned limitations, we require an interpretable forecasting approach for time-series so that, we could (1) predict the trend of COVID-19 cases in coming weeks and (2) discover the most significant features in the forecasting model to infer the most effective types of policies. More specifically, our time-series forecasting method should be:

- *Flexible to various types of input variables:* Many state-of-the-art forecasting methods such as N-Beats [7] are univariate and forecast a target variable just based on its history (Without using any external information or variables). However, in our task, we have a wide variety of input variables with different types and characteristics. In addition to the history of time-series in each country, we have other time-variant variables that are either known for the future days (like the holidays in each country) or unknown (e.g. all the features related to the policies). Moreover, we have some time-invariant (static) input variables such as country and region (continent). Also, some of these variables are continuous (real-valued) (e.g. the history of confirmed cases and Deberta[2] embedding of policy comments) while some others are categorical (e.g. countries and measures attribute of policies). Therefore, a time-series forecasting model must be designed in a flexible way that would be able to effectively incorporate all different types of input features, unlike some other co-variate models such as DeepAR [8].
- *Applicable on various areas* Many well-known time-series forecasting models are dedicated to specific types of data such as financial time series and the prediction of stock prices. Since this problem is naturally different from most of the popular applications of time-series forecasting, we require a general-purpose model applicable to various types of time series including ours.
- *Explainable and interpretable* As previously mentioned, the ultimate goal of this project is to evaluate the effectiveness of different policies. Therefore, explainability is our essential requirement in the forecasting model. Without an explainable model, we cannot discover which features have a more significant contribution to our time-series forecasting, thus, we cannot evaluate and compare the policies.

4.1 Temporal Fusion Transformer

Temporal Fusion Transformer (TFT) [5] is one of the state-of-the-art models for time-series forecasting. TFT utilizes a unique and complex network consisting of several components (Figure 3) It contains LSTM components as well as self-attention layers for processing time-varying data, unique GRN[1]-based variable selection networks for discovering the most important input features, static covariate encoders for incorporating static (time-invariant) features into the network, and a gating mechanism for skipping unused components of the network.

TFT is a relatively new model[2] which outperformed all of the 9 competing time-series forecasting models on 4 real-world datasets. Our study demonstrated that it is the best fit for our task since it satisfies all three requirements mentioned in the previous section. First of all, TFT is flexible to effectively integrate a complex mixture of inputs of various types including static and time-varying inputs, categorical and real-valued inputs, and known and unknown (in the future) inputs. Secondly, TFT is a multi-horizon time-series forecasting model which is designed to predict and forecast time-series for more than only one step ahead which is suitable for our problem since the impact of imposing any possible regulation would be latent and takes some time to be observed in the daily number of new COVID confirmed cases. In other words, this model makes it possible to forecast the whole time series for a duration of time from the timestamp we are testing. TFT performance has been tested on various real-life datasets such as an electricity consumption dataset, a traffic dataset, and a grocery sales dataset. Finally, despite most of the other popular time-series forecasting models which are used as "black box", TFT provides unique interpretations of the result

[1] Gated Residual Network

[2] TFT was introduced in September 2020

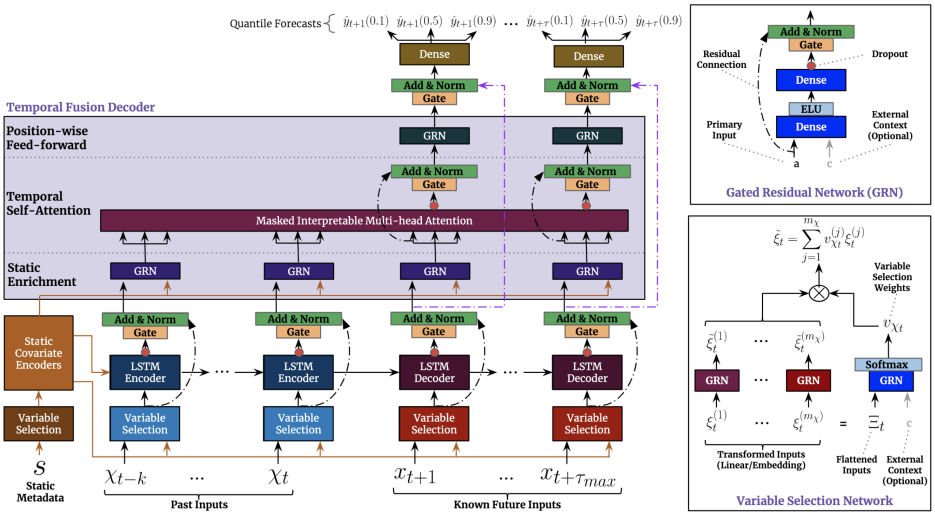

Figure 3: An overview of TFT architecture with its components and inputs/outputs.

and input features. For example, it can analyze the weights and attention in the variable selection component to report a relative importance factor for each input feature, which would be so helpful to investigate policies' effectiveness in our task.

4.2 Input data

In this project, we used pytorch-forecasting implementation of TFT. For utilizing TFT, we needed to convert our datasets into the appropriate time-series format to be consistent with our pytorch-forecasting models. For this purpose, we created a single tuple representing each day in each country[3], thus, we aggregated all the policies belonging to the same day and same country, and filled policy-related features by zero for (country, day) pairs containing no policies. Therefore, we trained and tested the TFT model a dataset of 176 time-series, each containing 424 days (One time-series for each country). The TFT model receives the history of our time-series value and input features in each timestamp (day) back to a certain number of steps in time as for its encoder. Our maximum encoder length was 70, which means we took the history back to 70 days back in time into account. We used the following input features for training TFT:

- *Country and Region (continent).* Both of these features are in the form of categorical inputs which are static for a single time series. As a matter of fact, the country itself was the criteria to define separate time series.
- *History of confirmed COVID-19 cases* . In each country, i.e. time-series, which we are forecasting the confirmed COVID-19 cases (our target) at some point in time, we make use of the history of these cases (target) from the current day (timestamp) to several days back in time.
- *Categories of Policies*. In some experiments, we used *category* as a single categorical feature while in some other experiments we used its one-hot representation (6 features) which is more convenient for aggregation.
- *Measures of Policies*. Similar to the categories, we used *measure* as a single categorical feature in some experiments while we used its one-hot representation (35 features) in some other experiments.
- *Comments of Policies*. We used the last hidden layer of DeBerta to produce the contextualized embedding of aggregated comments (concatenation) of all policies for each day (time step) in each country. We utilized them in three different variants. In some experiments, (1) we used all dimensions of these 768-dimensional representations from our BERT-based model outputs, as input features while in some other experiments, (2) we applied PCA to reduce the dimensionality of the embeddings to 20. In the third type of experiment, (3) we applied k-means to partition the DeBerta embeddings of aggregated comments into 10 clusters. In this case, we used the cluster number as a single categorical feature and fed it into our TFT model. In figure 6 in the appendix, we used the t-SNE method to visualize and demonstrate the effectiveness of this clustering in 2 dimensions.

As mentioned above, each policy has a type which is either introduction/extended (81%) or phaseout (19%). This difference is very crucial, and for training the model, we should consider if each regulation is being introduced/extended or is being removed. We have extended each of the Measures, Categories, and Comments features into two negative and positive sub-classes. For

[3]There are 176 countries (series k and 424 days t_i

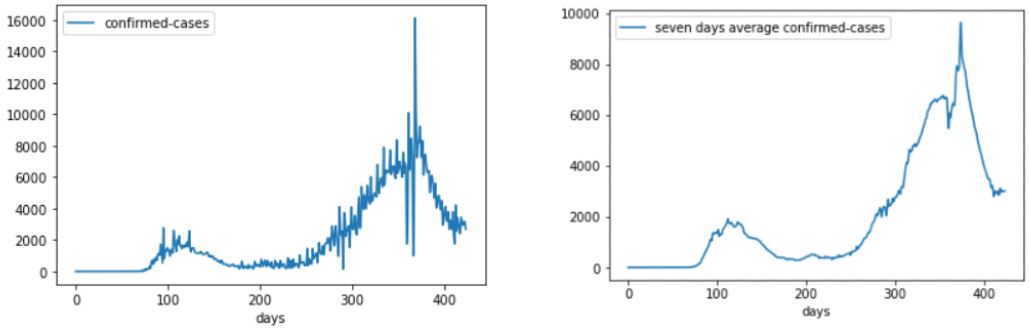

Figure 4: Daily confirmed cases time-series vs Last seven days average confirmed cases time-series in Canada. Seven days average will reduce the noise in the time-series

example, *Social Distancing* is one of the Policy Categories classes that a policy may have been labeled in this class. After prepossessing the data, instead of the *"Social Distancing"* class, we have two classes called *"Positive Social Distancing"* and *"Negative Social Distancing"*. If a policy is introduced/extended, we put it in the positive subclass, and if a policy is phased out, we put it in the negative subclass.

4.3 Experiments

We have used a smoother variant of time-series constructed based on the average number of confirmed COVID-19 cases in the last seven days in all experiments. This would reduce the amount of noise present in daily reported confirmed cases time series. The difference between the two time-series is shown in Figure 4. For each time-series (i.e. each country), we have 424 days, but not all these 424 days help us train the model. In fact, in the first months of the pandemic, almost all countries reported a low number of confirmed cases. Also, we do not have the latest regulations enforced in the last weeks in the dataset. Hence in training, we start the time-series from the second month and omit the last three months. On the other hand, we speculate that statistics from countries with a low number of confirmed cases are not reliable enough to express the impact of imposing government policies and regulations on the number of confirmed cases. Thus we have selected 80 countries from the whole 176 countries that have the most confirmed cases on average. The frequency of the 80 selected countries based on their continent is shown in Figure 7 in Appendix. For testing the performance of TFT models with a different configuration, we have used specific two-week periods of 30 different time-series (i.e. 30 different countries). These 30 countries are also chosen based on the average number of confirmed cases to make sure of the quality of the testing data. These countries are mainly from Europe and the Americas continents.

4.3.1 Different Model Configurations

The TFT model has been deployed and experimented with various settings based on our feature modeling and the network architecture hyper-parameters. In this section, we explain some of these tested model variants and their input features space.

1. *Measures, Categories, and Comment of Policies in form of Categorical data*: In this variant, the history three categorical features are passed to the model besides the Covid cases (real-valued number) which are aiming to forecast for the future time steps; the Measures of Policies (35 classes), the Categories of Policies (6 classes) and the Comments clusters of Policies (10 classes). In the previous sub-section, the process of acquiring these cluster labels for comments to 10 classes has been described. Overall, what our encoder history inputs consist of,
    - 3 categorical inputs,
    - 1 numerical input of the COVID cases, and
    - the 2 static per time-series categorical variables for country and region.
2. *Measures, Categories, and Comment of Policies in form of Numerical data*: In this case, instead of three categorical features, we imposed a different approach to build out feature space. It has 12 features for positive (6) and negative (6) Categories of Policies, each showing the number of policies imposed from that category on a certain day in our time-series for each country; 70 features for positive (35) and negative (35) Measures of Policies, similarly defined as Policy Categories features; and 40 features for positive (20) and negative (20) of Policy Comments embedding. The 20-dimensional embeddings are generated with PCA as mentioned earlier. For instance, for a time step in a time-series, feature i of Categories/Measures of Policies indicated the number of policies of type of Category/Measure, such as the number of policies imposed related to introducing/extending the "Full Lockdown" measure. In this case, we would have,

- 82+40+1 numerical inputs for Policy Cat./Mes., Policy Comment embeddings and the COVID cases, and
- 2 static per time-series categorical variables for country and region.

3. *Measures, Categories, and Comment of Policies in form of binary data*: This version is similar to the previous case *Variation 2*. The difference is that in each feature of positive or negative Policy Category/Measure, we just took the information about the existence of such policy on that specific day of our country-based time series. In other words, if on a certain day, 2 new policies regarding Measure X and phased out of a policy from Measure Y were imposed by their government, the value of Positive-Measure-X would be *'true'* (rather than number 2 discussed in the last model variation) and Negative-Measure-Y would be *'true'* (rather than number 1). The same pattern applies to all Policy Categories and Measures. To summarize,
   - 40+1 numerical inputs for Policy Comment embeddings and the COVID cases,
   - 82 categorical ('true/false') inputs for Policy Cat./Mes., and
   - 2 static per time-series categorical variables for country and region.
4. *Policies Comments Only data*: The input features of this variant include the Policy Category or Measure information and only utilizes the 20-d embeddings of aggregated Policy Comments (Positive and Negative are processed separately like model 2 and 3.
   - 40+1 numerical inputs for Policy Comment embeddings and the COVID cases, and
   - 2 static per time-serie categorical variables for country and region.
5. *Policies Category and Measure Only*: As opposed to Model 4, this variant ignores the comment embeddings and only takes the Categorical ('true/false') version on Policy Category and Measure features (explained in model 3) into account. This compact version is similar to common use cases of TFT which forecast time-series based on some categorical one-hot encoded features, and is beneficial to evaluate the impact of considering the textual information of Policies Comments. Overall, the input features consists of,
   - 1 numerical inputs for the COVID cases,
   - 82 categorical ('true/false') inputs for Policy Cat./Mes., and
   - 2 static per time-series categorical variables for country and region.
6. *Policies Comments Only Pos+Neg data*: This case is very similar to model 4 but instead of having different embeddings for introduction/extension (Positive) and phase-out (Negative) Policies separately we treated them all together aiming to examine if the DeBerta-driven embeddings can have the contextual meaning to forecast the time-series positive and negative effect without exclusively and manually separating them.
   - 20+1 numerical inputs for the Comment embeddings and COVID cases,
   - 2 static per time-serie categorical variables for country and region.
7. *Accumulated Window of Policy Category and Measures History- Numerical* Since the format of our input features for each day were depending on that specific day information, our dataset was extremely sparse. To compensate and diminish this issue, this model takes the Category and Measure of Policies from the last 25 days and accumulates their numbers. In other words, for each time step, all Policy Category and Measure count-features from model 2 were accumulated in from the last 25 days. However, since arithmetic addition of 20-D PCA embeddings does not seem to be interpretive and concatenating all comments text in the last 25 days would almost exceed the BERT architecture, we kept the embedding from the last day just as Model 2, intending to concentrating on the most recent time step policies. Hence, the shape, type, and number of inputs are identical to Model 2.
8. *Accumulated Window of Policy Category and Measures History- Categorical* This version relation with model 7 is similar to the relation between model 2 and 3. It means that 'true/false' classes were derived from the previous model variant, the same way we derived explained for model 3 from model 2. Therefore, the shape, type, and number of inputs are also identical to model 3.
9. *Full-BERT Comments Only* In this case, all of the policies comments regardless of being positive or negative were aggregated and given to DeBerta, and the complete 768-D output embedding for each day were fed into our TFT model. This experiment investigates the possibility of losing information by reducing dimension (PCA) or clustering. However, since the input feature is extremely large for our computational resources and TFT model, we only experimented with this variant on the top 37 time-series (countries) with the most average COVID cases instead of 80 countries in previous models. Therefore, for each time step, we would have,
   - 768+1 numerical real-valued inputs for Policies Comments and the COVID cases,
   - 2 static per time-series categorical variables for country and region.

4.4 Results and Interpretations

In this section, we report and discuss the results of our experiments. To investigate the performance of our models, we utilized a metric called Mean Absolute Percentage Error (MAPE), which is a common measure of prediction accuracy of a forecasting method and defined as follow,

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}|\frac{A_t - F_t}{A_t}|$$

where F_t is the forecast value of time step t while A_t is its actual value, and lower values of this metric mean better and more accurate performance.

Table 4.4 demonstrates the mean of MAPE over all time-series in our test set. As observed, increasing TFT hidden size would generally tend to enhance the performance, which was expected since the model would be more expressive. However, in our experiments, we have seen that this trend would not hold if we increase the hidden size to larger numbers (≥ 64) which can be the result of the small dataset with large numerous variables and reasons that can affect our time series. We generally suspect that due to these reasons and the complex nature of the problem, its variables, hidden indirect and direct factors not taken into account in our data, our models over-fit due to difference we observed between our training and validation Quantile Loss used during training. However, Amongst our TFT model variations, the one with clustering the Policy Comments (model 1) resulted the best, which is promising since it supports the supposition that using textual policies information, even without separating the introduction/extension or phase-out, could be more helpful compared to model 5 (without any comments) and model 3 (with separate positive/negative PCA comment embeddings).

Model Number	Model Hidden Size	MAPE ↓
1	32	0.2286
1	16	0.2356
2	32	0.6356
3	32	0.3342
3	16	0.2396
4	32	0.5552
4	16	0.5688
5	32	0.2321
5	16	0.3025
6	32	0.6014
6	16	0.2906
7	32	0.3538
7	16	0.6251
8	32	0.4276
8	16	0.6327
9	32	0.2749

Table 3: Testing various TFT models with different features inputs. Models are defined in section 4.3.1. ↓ indicated that a lower MAPE score is a better score

Furthermore, it can be seen that model 9 with full 768-embeddings is not outperforming the rest of the models. First, we suspect that it needs a much larger and more complex TFT model, which is preferably fine-tuned end-to-end with the forecasting module. The last attribute can not be easily achieved due to the limited resources and TFT constraints. Secondly, TFT is designed to be interpretable by utilizing its gate encoders and variable selection mechanism. This characteristic is not suitable for vector embeddings since they consist of correlated real-number token values. Furthermore, the contribution and importance of token embeddings, based on their positions, are not interpretable and meaningful. Another observation in comparing model 2 and 6 is that accumulating the number of policies from a certain Category or Measure from the last 25 days might be beneficial since it reduces the sparsity of our input features with interpretable numbers (e.g. informing about more imposed regulations would help it forecast the COVID cases more accurately).

We used Model 5 to investigate and discuss the effectiveness of the policies, which is the most interpretive configuration amongst all with acceptable performance amongst our variety of models (top 3). Figure 5 illustrates the most and least important Policy Category/Measures features used in forecasting our time series. TFT model introduces an approach to quantify the input variables' importance. It is calculated by aggregating selection weights for each variable through the entire test set, by recording the 10th, 50th and 90th percentiles of each sampling distribution, in Quantile Loss settings[5].[4] This is one of the purposes and useful

[4]We refer the readers to the TFT paper for a comprehensive explanation.

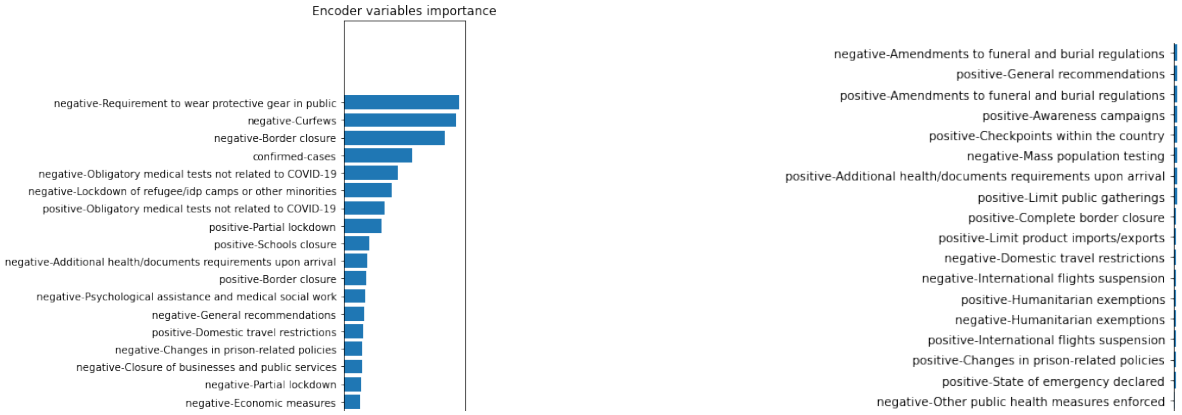

Figure 5: TFT encoder variable Importance: most contribution (left) and least (right) contribution to forecast the time series

functionalities of the TFT model in terms of interpretability that helps us measure the effectiveness of the imposed policies concerning the number of confirmed cases. As illustrated, policies such as related to mandatory mask regulation, lock down, and border closure, etc. seems to be more effective which is reasonable. On the other hand, policies regarding prisons, domestic travels, or humanitarian exemptions seems to be not effective for confirmed cases by our model 5 run.

4.5 Discussion and Future Work

As observed in our model variations results, imposing a mechanism to efficiently include the textual information of Policies' Comments is beneficial. An idea that is worthwhile to execute is training an auto-encoder on DeBERTA to generate latent features from the comments with reduced dimensions. Moreover, the ideal situation would be training both of these networks together as an end-to-end system. However, it demands changes in the original TFT network and how it handles the inputs. Nevertheless, the interpretability and meaning of these features would still be an issue to extract meaningful characteristics for policies' effectiveness. Another field of experiments for the textual comments is to investigate other models than DeBerta, such as newer successors of BERT (Albert, Electra, etc.) or other architectures such as GPT2.

Due to the complicated nature of our problem, an idea to facilitate the problem is to predict and forecast the increasing or decreasing trend of COVID cases or the effectiveness of the policies on the speed and rate of change, instead of forecasting the accurate number of cases in each day as a set of time series. This seems to be consistent with our observations for predicted cases in an upcoming interval because of our model. For instance, our model could predict that the number of cases would decrease or increase on average for the next 14 days compared to the cases before seen up to that time step, just as the actual cases trend on that 14-day interval. In this context, TFT is claimed to have the ability to forecast time-series as a classification problem rather than regression. Therefore formulating our problem more suitably concerning our data limitation and constraints would be beneficial.

One of the major limitations observed in our experiments was the numerous factors that affect our target function. This made our input feature space extremely sparse with some spikes in a few features on the day a policy is imposed. Hence, trying to aggregate data to reduce sparsity seemed to be beneficial. However, our model did not enhance by such a mechanism. Investigating better methods to reduce sparsity would also be a future field of research. Besides, there are other social and clinical factors that directly or indirectly influence the spread and therefore the number of cases of COVID. One of these factors is the holidays in different countries that we suspect would change people's behavior towards spreading the virus with their trips. Unfortunately, we only could find a limited data set of national holidays for a few countries that we did not utilize since we suspect it would add the sparsity to our input features.

Overall, the limited data and sparsity of the policies with a large number of features to consider were some of the main issues that we faced throughout this study. Moreover, using more complex TFTs with larger architectures and more attention heads would require more data that was not available in our case.

5 Conclusion

In this project, we investigated the effectiveness of governments' policies and regulations related to COVID-19 on the number of new confirmed cases. As our first approach, we developed Multi-Layer Perceptron (MLP) models that explicitly predict the effectiveness of individual policies and modeled the problem in both classification and regression fashion. We also studied

the problem as a set of time series (one per country). After looking into several forecasting and auto-regressive models such as NBeats and DeepAR, we concentrated on a model called Temporal Fusion Transformer (TFT) that was best suited for our problem. By defining different architectures and variants from TFT, we created multiple models to forecast and predict the COVID confirmed cases. We also investigated the interpretability of our model and provided the most and least effective group of government policies that contributed to our model.

References

[1] Acaps government measures dataset. *Retrieved April 10, 2021 from https://www.acaps.org/covid-19-government-measures-dataset*.

[2] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

[3] Mukul Jaggi, Priyanka Mandal, Shreya Narang, Usman Naseem, and Matloob Khushi. Text mining of stocktwits data for predicting stock prices. *Applied System Innovation*, 4(1):13, 2021.

[4] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *arXiv preprint arXiv:1706.02515*, 2017.

[5] Bryan Lim, Sercan Ö. Arik, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *arXiv preprint arXiv:1912.09363v3*, 2020.

[6] Vitalii Mokin. Covid-19: Holidays of countries (2021, jan). *Retrieved April 20, 2021 from https://www.kaggle.com/vbmokin/covid19-holidays-of-countries*.

[7] Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis forinterpretable time series forecasting. *arXiv preprint arXiv:1905.10437v4*, 2020.

[8] David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3):1181–1191, 2020.

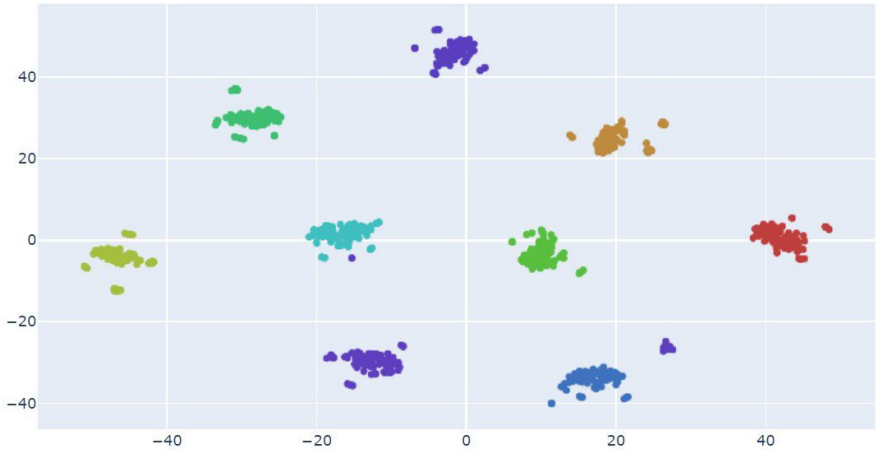

Figure 6: 10-class kmeans clustering of Deberta comment-embeddings represented in 2 dimenisons by t-SNE. Up to 100 samples of comment-embeddings in each class are represented. Points with the same color belong to the same cluster.

6 Appendix

6.1 Hyper Parameters and Settings

Binary Classification MLP:

- Learning rate: 0.001
- Epochs: 1000
- Learning rate decay: 0.001/1000
- Optimizer: ADAM
- Loss function: binary cross entropy

Regression MLP:

- Learning rate: 0.001
- Epochs: 2000
- Learning rate decay: 0.0001/2000
- Optimizer: ADAM
- Loss function: mean square error loss

6.2 Tables

6.3 Graphs and Diagrams

Measure	Percentage
Health screenings in airports and border crossings	1.62%
Awareness campaigns	3.26%
Testing policy	2.50%
Schools closure	3.65%
International flights suspension	3.14%
Strengthening the public health system	7.73%
Other public health measures enforced	3.74%
Domestic travel restrictions	4.26%
Emergency administrative structures activated or established	3.06%
Closure of businesses and public services	9.61%
Additional health/documents requirements upon arrival	1.61%
General recommendations	3.91%
Requirement to wear protective gear in public	2.99%
Isolation and quarantine policies	5.73%
Limit public gatherings	9.32%
Limit product imports/exports	0.29%
Visa restrictions	2.44%
Partial lockdown	2.94%
Border closure	3.75%
Curfews	3.27%
Economic measures	12.46%
State of emergency declared	1.96%
Amendments to funeral and burial regulations	0.57%
Changes in prison-related policies	0.59%
Mass population testing	0.59%
Border checks	0.44%
Military deployment	0.47%
Surveillance and monitoring	2.33%
Full lockdown	0.62%
Checkpoints within the country	0.29%
Lockdown of refugee/idp camps or other minorities	0.10%
Psychological assistance and medical social work	0.60%
Humanitarian exemptions	0.12%
Complete border closure	0.03%
Obligatory medical tests not related to COVID-19	0.01%

Table 4: Percentage of 35 measures

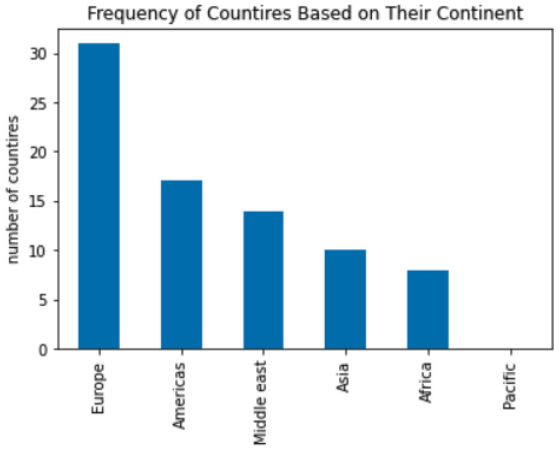

Figure 7: Frequency of the top 80 countries based on the average number of confirmed cases in each continent. These countries are selected for training the models.

Implementation Type	Label	Measure Category	Counts
Introduction/extension	1	Public health measures	2112
		Social distancing	694
		Movement restrictions	1017
		Governance and socio-economic measures	1351
		Lockdown	200
		Humanitarian exemption	7
		Total	5381
	0	Public health measures	4387
		Social distancing	2227
		Movement restrictions	2576
		Governance and socio-economic measures	2308
		Lockdown	443
		Humanitarian exemption	15
		Total	11956
Phase-out	1	Public health measures	231
		Social distancing	992
		Movement restrictions	479
		Governance and socio-economic measures	85
		Lockdown	68
		Humanitarian exemption	0
		Total	1855
	0	Public health measures	251
		Social distancing	1165
		Movement restrictions	606
		Governance and socio-economic measures	96
		Lockdown	114
		Humanitarian exemption	2
		Total	2234

Table 5: Label counts for measure categories