FYP-2021: Final Presentation

3 messages

"葉志立 YIP Chi Lap [Beta]" <clyip@cs.hku.hk> Fri, Apr 22, 2022 at 4:05 PM
To: 方均正 Fong Kwan Ching <u3556490@connect.hku.hk>
Cc: "\"葉志立 Yip Chi Lap [Beta]\"" <clyip@cs.hku.hk>, "\"蔡綺瓊 Choi Yi King [Loretta]\"" <ykchoi@cs.hku.hk>

Hi,

Here it's the notes I jotted down for your final presentation.

Regards,
葉志立 YIP Chi Lap [Beta]
PhD BEng(CompEng) FHKMetSoc MACM MIEEE
Department of Computer Science
The University of Hong Kong

202204221430-FinalPresentation-FcstTCProp-Fong Kwan Ching.txt
2K

Fong, Kwan Ching <u3556490@connect.hku.hk> Sat, Apr 23, 2022 at 5:37 PM
To: "葉志立 YIP Chi Lap [Beta]" <clyip@cs.hku.hk>
Cc: "蔡綺瓊 Choi Yi King [Loretta]" <ykchoi@cs.hku.hk>

Dear Dr Yip,

Thank you for the feedback. I would like to add to the issues raised my responses, which are as follows:

> HKWW model's role
The tropical cyclone (TC) warning signal probability forecast by HKWW is the nearest comparable model/forecaser /product this project has. Therefore, it was used to check whether the baselines are showing appropriate performance such that the comparison with experimental models can be more sound.

> The targets are not the same
Yes, the HKWW models have a simpler task than the models developed in this project. However, there are some overlaps so it may still be suitable. Moreover, the need to include a comparable baseline for the baselines in the analysis is considered important, as it can serve as some rough third-party verification of the project's outcomes.

> Why the 0.87 estimate?
There are several factors taken into account:
(1) The HKWW reported score of 0.845 may represent the best results a traditional approach (no dynamical data) can normally get. This assumes that HKWW is trustworthy.
(2) The experimental models worked better than their baseline counterparts, therefore, it can be argued that if HKWW used dynamical data too, then the scores they would get would be above 0.845.
(3) The score improvements obtained by switching datasets were limited. The obtainable margin (in (2)) above 0.845 would not be as high as 0.1.
(4) The very high scores (0.93-0.96) obtained using 6500 experimental samples may represent overfitting, the actual performance of the ensemble obtained would likely be lower.
(5) If there were more samples, then the high variance issue (in (4)) could be corrected to some extent.
(6) It was found that the models had better scores in general when only 6500 samples were available than when 45000 were.
(7) In fact, the scores the baseline models give should not be much better than HKWW's, because the targets HKWW predict are ostensibly easier to predict.
In conclusion, the best scores the experimental models and the ensemble could obtain would be between 0.845 ((1), (2), (7)) and 0.93 ((3)-(6)). The 0.87 estimate was chosen to be conservative, because TC behaviour is likely more erratic than statistical models can accurately capture.

> Is (the estimated) 0.87 score something you want to achieve?
Yes, but it depends on the interpretations of the question. If it refers to the desirability of a score of 0.87 when there is sufficient data, then I would say that score is indeed satisfactory, though further improvements would be great. If it refers to some quantitative objective set at the beginning of the project (e.g. the project is not considered a success unless a score of 0.87 is reached), then I would say my answer is undefined. At the onset of this project, no estimations about the scores obtainable were ever made, as the correctness of the hypothesis (statistical-dynamical approach is better) had yet been verified, nor was the potential improvement margin understood.

> Classification vs regression problem
The goal is to find some probabilities of an event happening. The intuitive approach would be to consider the continuous probabilities as a target to obtain via regression analysis. Alternatively, given that the event's occurrence is binary and the input data is also labelled with these binary options, the problem could also be treated as a classification problem, where the models should identify whether the event would occur (yes or no). The key here is to also make the classifier produce a confidence score or a probability estimate. Therefore, the task of finding the probabilities can be done both by regression and classification. The disadvantages of either approach is that regressors may predict values outside of [0,1] which cannot be interpreted and classifiers may tend to predict values close to 0 or 1 but nothing in between. The advantages are the inverse of the disadvantages: regressors had slightly better probability calibration, but classifiers were easier to develop, test and optimize. Much of this project focused on classifiers because they were faster to develop and usually had consistently good performance.

> Linear model and generalized additive models (GAMs)
The main concept of GAM is to treat the target variable as a linear combination of some inputs, which may in turn be a function approximated using a number of splines, with some link function bridging between them (i.e. the link function is applied to the linear combination to obtain the predictions; ref.: Wikipedia, pyGAM documentation). It overcomes the weakness of linear models like linear regression and logistic regression where nonlinearities in the data are poorly modelled. In this project, the term "linear model" was applied to the general class of modelling methods that assume the output is some linear combination of the input.

> Time series modelling: exogenous variables used
The consideration was to rearrange the dataset into a time series-like structure, so that the target (endogenous) variables corresponded to the labels (Y) and the feature variables (X) were used directly as exogenous variables without further selection. Nonetheless, very little was actually done, e.g. the DIST00, DIST06, ..., DIST24 columns (TC center radial distance from Hong Kong over the last 24 hours) were simply brought together to give a DIST time series so the models could use them. This was to minimize changes made to the datasets, especially the columns, so that fairer comparisons were possible. The way the time series models function prevents retrieval of feature/exogenous variable importance values to further the analysis (e.g. filter out irrelevant variables), because that option is missing in the API.

> Multilayer perceptron (MLP) training configurations
As many as 1000 epochs were allowed, but when training loss stopped improving for 10 consecutive epochs the training procedure would be halted. This was in accordance with scikit-learn's default MLP training behaviour (documentation links: classifier, regressor).

> MLP regressors seem to be disproportionately bad
This is because MLP regressors tended to overfit under the same hyperparameter settings when compared to MLP classifiers. This may be due to the models having different loss functions (mean squared error vs log loss), so comparing with the same hyperparameter settings could be unfair. Decreasing model complexity and strengthening regularization were considered, but owing to time constraints (each MLP regressor took about 3-6 hours to train, which made meticulous fine-tuning a time-consuming task) little tests could be made.

> Usability
An issue with the developed forecaster is that it is inconvenient for its potential users. The dynamical data must be manually fetched from the data source and manually preprocessed for each forecast, while making the forecast itself involves going into the source code and running the notebook cell by cell. If there were more automation and a simple user interface, where the users may submit input data without having to touch the source code (e.g. file uploads for dynamical data, web form for TC position data, etc.), then it would be much more convenient. The parameters of the models themselves are already fine and do not require further tuning (unless a new dataset is available, thus enabling further analyses, but this is in my opinion beyond the scope of the forecaster's usability).

Finally, the recorded video of the final presentation is here (backup link: here). Additional information about the project, i.e. all source code and reports, are available for your reference at GitHub.

If you have further questions, please do not hesitate to write back at your nearest convenience.

Best regards,
FONG Kwan Ching
Student

[Quoted text hidden]

"葉志立 **YIP Chi Lap [Beta]"** <clyip@cs.hku.hk> Sun, Apr 24, 2022 at 9:22 PM
To: 方均正 Fong Kwan Ching <u3556490@connect.hku.hk>
Cc: "\"葉志立 Yip Chi Lap [Beta]\"" <clyip@cs.hku.hk>, "\"蔡綺瓊 Choi Yi King [Loretta]\"" <ykchoi@cs.hku.hk>

Hi,

Answers received.

Regards,
葉志立 YIP Chi Lap [Beta].
PhD BEng(CompEng) FHKMetSoc MACM MIEEE
Department of Computer Science,
The University of Hong Kong.

[Quoted text hidden]