International Journal of Forecasting 38 (2022) 1400–1404

Contents lists available at ScienceDirect

International Journal of Forecasting

journal homepage: www.elsevier.com/locate/ijforecast

A fast and scalable ensemble of global models with long memory and data partitioning for the M5 forecasting competition

Kasun Bandara [a,*], Hansika Hewamalage [b], Rakshitha Godahewa [c], Puwasala Gamakumara [d]

[a] *School of Computing and Information Systems, Melbourne Centre for Data Science, University of Melbourne, Australia*
[b] *School of Computing Technologies, RMIT University, Melbourne, Australia*
[c] *Department of Data Science and AI, Faculty of Information Technology, Monash University, Melbourne, Australia*
[d] *Department of Econometrics and Business Statistics, Monash University, Melbourne, Australia*

ARTICLE INFO

Keywords:
M5 forecasting competition
Global forecasting models
Sales demand forecasting
LightGBM models
Pooled Regression models

ABSTRACT

This work presents key insights on the model development strategies used in our cross-learning-based retail demand forecast framework. The proposed framework outperforms state-of-the-art univariate models in the time series forecasting literature. It has achieved 17th position in the accuracy track of the M5 forecasting competition, which is among the top 1% of solutions.

1. Model overview

The proposed framework is an ensemble of Pooled Regression model (PR, Gelman & Hill, 2006; Trapero, Kourentzes, & Fildes, 2015) (refer Appendix B) and Light Gradient Boosting Machine (LightGBM, Ke et al., 2017) model, both trained globally across a collection of time series. These forecast models with the cross-learning capability, also known as Global Forecasting Models (GFM), have recently offered numerous possibilities for forecast practitioners, which have never been realised by the traditional univariate forecasting methods that forecast in isolation (Januschowski et al., 2020; Kang et al., 2020).

Recent theoretical and empirical insights (Hewamalage, Bergmeir, & Bandara, 2020; Montero-Manso & Hyndman, 2021) suggest that GFMs are a powerful class of forecasting models that can be designed with a much higher complexity, yet still achieve better generalisation error than the traditional univariate models for larger datasets. This ability of GFMs to control model complexity becomes an important factor when designing forecast models with long memories. Following these theoretical recommendations, the base global models of our ensemble framework are carefully crafted to gain complexity through long model memory and a time series grouping strategy. We achieve the long model memory by using 400 days of time lagged sales observations as input features of our model. To create the sales lags, we apply the moving window strategy to the entire time series, transforming it into pairs of input and output windows. Here, the optimal size of the input window, i.e., 400, is determined by using a grid-search-based hyper-parameter optimisation strategy (refer Appendix A). In addition to the sales lags, the proposed framework also accounts for the static and dynamic exogenous variables available in the dataset. This includes day of week, day of month, weekend or non-weekend, month, snap, event name 1, event name 2, event type 1 and event type 2. The proposed grouping methodology builds separate global models for each identified group of time series from the same product department, i.e., 70 different global models for the total departments (Bandara et al., 2019). On top of the model complexity, the proposed grouping strategy also attempts to address the

[*] Corresponding author.
E-mail address: Kasun.Bandara@unimelb.edu.au (K. Bandara).

https://doi.org/10.1016/j.ijforecast.2021.11.004
0169-2070/© 2021 International Institute of Forecasters. Published by Elsevier B.V. All rights reserved.

Table 1
The WRMSSE values for the base models and the ensemble at all considered hierarchical levels of aggregation. For each row, the results of the best performing method(s) are marked in boldface.

Level	Ensemble	LightGBM	PR
1	0.247	0.340	**0.216**
2	**0.342**	0.397	0.350
3	**0.446**	0.505	0.460
4	0.308	0.384	**0.291**
5	**0.404**	0.482	**0.404**
6	**0.412**	0.458	0.423
7	**0.501**	0.558	0.524
8	**0.520**	0.567	0.539
9	**0.622**	0.684	0.643
10	0.992	1.049	**0.985**
11	0.944	0.983	**0.941**
12	0.892	0.921	**0.890**

issue of data heterogeneity in GFMs (Bandara, Bergmeir, & Smyl, 2020; Bandara et al., 2019; Godahewa, Bandara, Webb, Smyl, & Bergmeir, 2021). In line with the recommendations of Bandara et al. (2020), as a preprocessing step, we stabilise the variance of the grouped time series by applying a mean-normalisation strategy to the grouped time series. Furthermore, to bring model diversity to the forecast combination (Lichtendahl & Winkler, 2020), we employ both linear (PR) and nonlinear (LightGBM) cross-learning models in our forecast framework. To obtain the final predictions of our framework, we compute the simple average of PR and LightGBM model forecasts. Table 1 reports the Weighted Root Mean Squared Scaled Error (WRMSSE) error values of the LightGBM and PR forecasts separately, and their ensemble at all the hierarchical levels of aggregation considered for the final test period of the M5 Competition. The model diversity introduced by the ensemble is evident since the addition of the PR model on top of the LightGBM has made the results better at many of the aggregation levels.

Fig. 1 gives an illustration of the proposed solution. The source code relevant to this framework is publicly available at https://github.com/kasungayan/M5Comp.

2. Comparison with the M5 winning solution

While our method as described above has demonstrated competitive performance at the accuracy track of the M5 Competition, it is different from the best performing solution proposed by In (2021) in a number of ways. These differences are detailed with respect to methodology in Table 2.

According to Table 2, it can be seen that the M5 winning solution performs grouping at multiple levels, and engineer various hand-crafted features, in addition to readily available features from the M5 dataset. Regarding the forecast setup, we see that the winning solution uses the Tweedie loss to train the LightGBM model, which is specifically suited for intermittent data. The final forecasts of the M5 winning solution are obtained by combining the predictions generated from the recursive and direct forecasting strategies.

On the other hand, our proposed solution applies grouping at the department level. Additionally, it performs a mean normalisation strategy to account for scale differences, which the winning solution overlooks. Also, our solution mostly utilises the features already available in the dataset and does not engineer new features to train the models. We use separate loss functions to train LightGBM and PR models, and apply only the recursive strategy to generate the forecasts from the models. Moreover, the M5 winning solution utilises the information available from upper levels of the sales hierarchy by including the historical sales at store and state levels as features to train the LightGBM model. Its better accuracy reflects this at higher levels of the hierarchy than our method, which does not use information from other hierarchy levels.

3. Conclusions

Overall, the methodological differences between the two solutions have resulted in the differences between the final rankings. As for the best performing method, their extensive feature engineering, validation technique, multilevel pooling, and addressing intermittent data with the loss function seem to have contributed to superior performance in the M5 competition. On the other hand, long memory models with data partitioning and model diversity through ensembling have enabled our proposed solution to achieve promising results in the competition. As a possible future work, the two approaches' methodological traits can be combined to explore their effect on forecast accuracy further. Furthermore, incorporating promotional related features to the model and analysing their effect on the final forecasting accuracy are also potential avenues for future research.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Hyperparameter selection & optimization

Table A.1 shows the bounds of the hyper-parameter values used throughout the LightGBM learning process, represented by the respective minimum, maximum, and optimal values.

In the hyper-parameter optimisation process, we first create a parameter grid by dividing the minimum and maximum range of each parameter summarised in Table A.1. Next, the performance of each parameter combination in the grid is evaluated on the validation dataset, i.e., the last 28 days prior to the forecast horizon and the same 28 days as the forecast horizon from the previous year, using the WRMSSE error measure. The parameter combination that gives the lowest WRMSSE on the validation dataset is chosen as the optimal parameter combination to train the LightGBM model.

The winning LightGBM solution uses nine hyperparameters to train the model, namely: Tweedie variance power, bagging fraction, bagging frequency, learning rate, number of leaves, minimum number of instances per leaf, feature fraction, maximum number of bins, and number

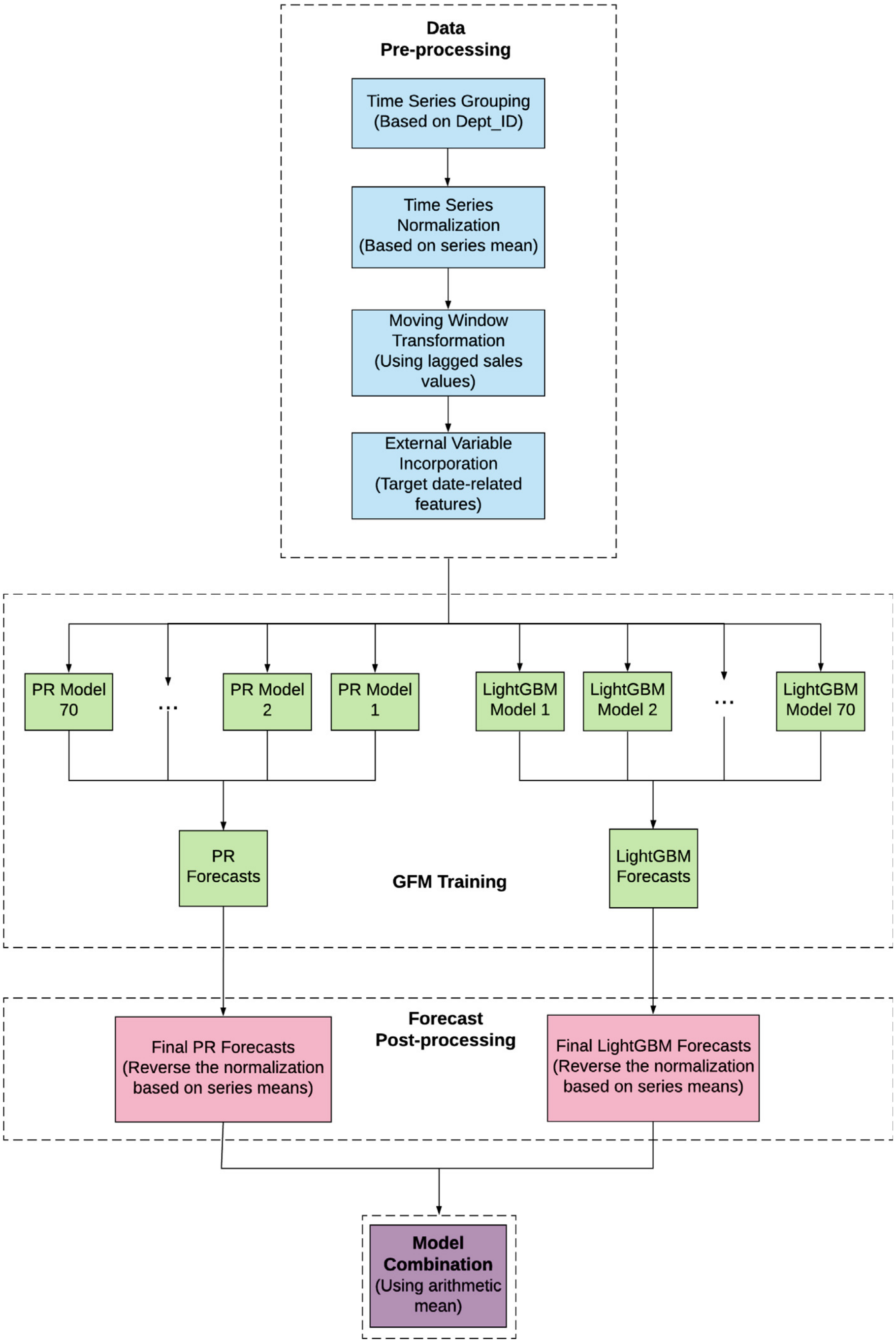

Fig. 1. The overall framework is comprised of three components, namely: (1) the preprocessing layer, which consists of a grouping phase, a normalisation phase, moving window transformation, and an exogenous variable embedding phase, (2) the GFM training layer, which trains GFMs for each group of time series in parallel and generates forecasts from the PR and LightGBM base models, (3) the forecast post-processing layer where the forecasts from the two models are separately processed to reverse the initial normalisation based on series means and (4) the model combination layer, which ascertains the final forecasts by computing the simple average of the final forecasts generated from the two base models.

of estimators. In contrast to our LightGBM implementation, the winning LightGBM solution use Tweedie variance power, maximum number of bins, feature fraction, and number of estimators, as additional hyperparameters to optimise the model.

Appendix B. Functionality of pooled regression model

The PR model that we employ in this work is a global version of an Auto-Regression (AR) model of order 400 (lags of sales) along with the other date-related exogenous variables (day_of_week, day_of_month, is_workingday,

Table 2
Comparison of the proposed method against the M5 winning solution.

Methodology	M5 Winning Method	Proposed method
Data Pre-processing	• At store level (10 groups), store-category level (30 groups) and department level (70 groups)	• Only at the department level (70 groups) • Mean normalisation to account for sales scale differences
Feature Engineering	• State_id, Store_id, Cat_id, Dept_id and Prod_id • Hand-crafted price features such as price, price_norm, price_max, price_min, and price_mean • day, month, year, day_of_week, week_num, month_week, is_workingday, is_weekend, snap and events as date related features • Two weeks of historical sales and sales mean, standard deviations at the store and state level for the entire training period • Sales mean and the sales standard deviation for different window sizes of one week, two weeks, one month, two months and half a year	• day_of_week, day_of_month, month, is_workingday, is_weekend, snap and events as date related features • 400 days of sales lags
Forecasting Setup	• LightGBM as the prediction model • Tweedie loss as the loss function • 9 LightGBM hyperparameters (see Appendix A) • Both recursive and direct strategies to generate multi step-ahead forecasts	• A combination of LightGBM and PR as the prediction models • Poisson loss as the loss function of LightGBM models and Re-weighted Least-Squares as the loss function of PR models • 7 LightGBM hyperparameters (see Appendix A) • The recursive strategy to generate multi step-ahead forecasts
Validation splits	• 13 validation splits corresponding to the last 13 28-day periods constructed through the rolling-origin mechanism	• Last 28 days prior to the forecast horizon and the same 28 days as the forecast horizon from the previous year

Table A.1
Parameter value ranges used to train the LightGBM models in our experiments.

Parameter	Min. value	Max. value	Opt. Value
Bagging frequency	1	5	1
Bagging fraction	0.25	1	0.75
L2 regularization	0	0.5	0.1
Learning rate	0.025	0.1	0.075
Number of leaves	30	320	128
Minimum number of instances per leaf	50	150	100
Number of iterations	500	1500	1200

is_weekend) as well as events and SNAP information. The term pooling indicates that one model is built using many series; 70 models in our case with a single model containing the trained parameters to predict all the series under a specific product department. This model can be formulated mathematically as in Eq. (B.1).

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \phi_3 y_{t-3} + \cdots + \phi_p y_{t-p} + \sum_{i=1}^{n} \alpha_i d_i + \sum_{i=1}^{m} \beta_i e_i + \epsilon_t \tag{B.1}$$

In Eq. (B.1), y_{t-1} to y_{t-p} denote the lags of sales ($p = 400$ in our case) of a particular bottom-level product. ϕ_1 to ϕ_p indicate the corresponding coefficients of the model trained by pooling across series. The variables d_i and e_i refer to the date-related and events (including SNAP) variables used in our model, where α_i and β_i are the respective coefficients. n and m are the total number of date-related variables (4 in our case) and events related variables (3 in our case for event_1, event_2 and SNAP) used in the model. ϵ_t is the error term at time step t. As seen in Eq. (B.1), the PR model can only capture linear relationships between the independent variables.

Our PR model used in this work has a similarity to the concept of pooling used by Trapero et al. (2015) specifically to predict SKUs having lesser historical promotional events to train a model from. Therefore, a pooled model is built by normalising the series to learn across many SKUs, similar to our approach. However, the independent variables used in the two works are slightly different since those authors use mainly promotional variables and their lagged values to predict the sales at future promotions. Apart from that, PR models have been used even earlier in other domains such as social sciences (Gelman & Hill, 2006). The partial pooling at the department level used in our work is the same as the multilevel models proposed by Gelman and Hill (2006). As those authors state, partial pooling is a middle ground that helps avoid issues from both extremes; model underfitting issues resulting from complete pooling (minimal series-specific information) and model overfitting issues due to no pooling (maximum series-specific information) at all.

References

Bandara, K., Bergmeir, C., & Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: a clustering approach. *Expert Systems with Applications*, *140*, Article 112896.

Bandara, K., Shi, P., Bergmeir, C., Hewamalage, H., Tran, Q., & Seaman, B. (2019). Sales demand forecast in E-commerce using a long Short-Term memory neural network methodology. In *Neural information processing* (pp. 462–474). Springer International Publishing.

Gelman, Andrew, & Hill, Jennifer (2006). *Analytical methods for social research*, *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, http://dx.doi.org/10.1017/CBO9780511790942.

Godahewa, R., Bandara, K., Webb, G. I., Smyl, S., & Bergmeir, C. (2021). Ensembles of localised models for time series forecasting. *Knowledge-Based Systems*, *233*, Article 107518.

Hewamalage, H., Bergmeir, C., & Bandara, K. (2020). Global models for time series forecasting: A simulation study. CoRR, abs/2012.12485, URL https://arxiv.org/abs/2012.12485.

In, Yeonjun (2021). M5 winning method. In *Kaggle*. URL https://www.kaggle.com/c/m5-forecasting-accuracy/discussion/163684. (Accessed: 8 october 2021).

Januschowski, T., Gasthaus, J., Wang, Y., Salinas, D., Flunkert, V., Bohlke-Schneider, M., et al. (2020). Criteria for classifying forecasting methods. *International Journal of Forecasting*, *36*(1), 167–177.

Kang, Y., Spiliotis, E., Petropoulos, F., Athiniotis, N., Li, F., & Assimakopoulos, V. (2020). Déjà vu: A data-centric forecasting approach through time series cross-similarity. *Journal of Business Research*.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., et al. (2017). LightGBM: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 3149–3157). Red Hook, NY, USA: Curran Associates Inc..

Lichtendahl, K., & Winkler, R. (2020). Why do some combinations perform better than others? *International Journal of Forecasting*, *36*(1), 142–149.

Montero-Manso, P., & Hyndman, R. J. (2021). Principles and algorithms for forecasting groups of time series: Locality and globality. *International Journal of Forecasting*.

Trapero, J. R., Kourentzes, N., & Fildes, R. (2015). On the identification of sales forecasting models in the presence of promotions. *Journal of the Operational Research Society*, *66*(2), 299–307.