Deep Learning on Administrative Tabular Data: A Comparative Study

Chiung Ching Ho and Chi-Ken Shum[1]

Data Analytics and Research Unit, Monetary Policy Department, Bank Negara Malaysia, Jalan Dato Onn, 50480 Kuala Lumpur, Malaysia

Abstract

Deep learning has traditionally been applied to perform analytics on large unstructured data such as videos, audio, images, and text. Initially, deep learning research on tabular data was not performed much, as the fixed structure inherent in tabular data was seen to negate the ability of deep learning techniques to elicit useful representation of tabular data. Recent advances in tabular deep learning have seen applications of the self-attention and transformer architecture as well as transfer learning which has improved the performance of deep learning on tabular data. Some of these approaches has improved on the results attained using traditional machine learning models such as gradient boosted trees for classification and regression tasks.

While these results are promising, the datasets used in these works were datasets which were typically used for bench-marking machine learning algorithms. This raises the question on how extensible the results would be if it were to be applied to administrative tabular data. To answer this question, we will curate a selection of administrative tabular datasets from open-sourced Malaysian administrative data. The curated dataset will be used to perform classification tasks using both traditional machine learning approaches as well as deep learning approaches. Classification is a machine learning technique which has been used to support policy decisions. As an evaluation, we will evaluate the results of the classification tasks as a measure of feature representation.

Feature representation is an important measure, as feature selection of administrative data by subject-matter-experts is a manual and laborious task. It is believed that automatic feature elicitation via deep learning approaches will reduce this dependency. The results of experiments conducted in this study have shown that deep learning tabular algorithms can achieve comparable results with optimised traditional machine learning when applied on open administrative data, without the need for extensive feature selection and feature engineering.

Keywords

Machine learning; deep learning, administrative data

[1] This paper was prepared for the IPS for the 64th World Statistics Congress to be held on 16 – 20 July 2023. All authors are from the Central Bank of Malaysia. Emails: hocc@bnm.gov.my, and shumchiken@bnm.gov.my respectively. The views expressed in this paper are those of the authors and should not be interpreted as reflecting the views of the Central Bank of Malaysia or of anyone else associated with the Central Bank of Malaysia. The authors would like to thank Syahirah Ismail for her assistance in developing the script to assess data used in this paper.

1. Introduction

The increasing rate of digitization of data has led to corresponding increases in the granularity (Bender et al., 2022) of administrative data that is being collected world-wide. Moreover, the usage of alternative and complementary datasets for supporting policy decision (Dessaint et al., 2021) has also contributed to increasingly complex granular data that supplements administrative data. These developments suggest that administrative data will become increasingly 'bigdata' like in terms of volume, velocity, and variety.

Granular administrative data is useful for informing policy decisions as it can present patterns and trends that is difficult to be detected from smaller aggregated datasets. The patterns detected can be useful for identifying correlations and causation, which are important components of any predictive task. Users of complex granular administrative data such as central banks (Araujo et al., 2022) has begun to develop strategies and uses-cases for leveraging these data for tasks including anomaly detection, outlier-detection, restoration of omissions in financial statements, inflation perception, economic forecasting, inflation forecasting, employment vulnerabilities as well as supervisory letter tone consistency.

Such tasks are frequently achieved using application of machine learning algorithms. Machine learning algorithms can extract patterns useful for predictions that is made based on data. For example, classification has been used to inform policy decision for Covid-19 (Roy & Ghosh, 2020) and hate speech (Burnap & Williams, 2015). Machine learning algorithms can be divided into two main categories: traditional machine learning and deep learning. Traditional machine learning algorithms are based on statistical models and typically work well on structured data (e.g. tabular data) and includes sub-categories such as supervised machine learning (models trained on labelled data), unsupervised machine learning (models are trained on un-labelled data), semi-supervised learning (models are trained using a mixture of labelled and un-labelled data) and reinforcement learning (models learn through trial and error with performance feedback). Deep learning (DL) algorithms are based on artificial neural networks and are typically designed to be used on unstructured data such as images, audio and text. DL can recognise complex patterns and features which makes them effective in tasks such as image recognition, speech recognition, natural language processing and increasingly image, audio and video generation. Examples of DL algorithms include Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Generative Adversarial Networks (GAN), Autoencoders and Transformer Networks.

Given that the majority of administrative datasets are in the form of structured or tabular data (Sun et al., 2019), it would seem to be a strange choice to consider DL algorithms as a technique for performing analytics. For instance, Malaysia's national open data platform (Malaysia Administrative Modernisation and Management Planning Unit (MAMPU), 2023) which contains open public sector datasets comprises 95% structured or tabular data. However, there is an advantage of using DL algorithms over traditional ML algorithms on tabular data in terms of representation of data that is being modelled. The effectiveness of both traditional and DL machine learning algorithms relies on accurate representation of the data that is being modelled. Accurate representation of the data, often also called a feature, requires domain knowledge and human expertise. This process is often manual and labour intensive, especially for supervised machine learning tasks. DL algorithms can learn feature representation by passing input data through a network of interconnected layers, which will elicit the features and representation of the input data.

The development of tabular-oriented DL models such as AutoInt (Song et al., 2019), NODE (Popov et al., 2019), TabNet (Arik & Pfister, 2020) , TabTransformer (Huang et al., 2020), FT Transformer (Gorishniy et al., 2021) and GATE (Joseph & Raj, 2023) has made the application of DL on tabular data more accessible. While the application of tabular DL models has yielded comparable results to traditional machine learning models, there are criticism of the suitability of using DL on tabular data (Shwartz-Ziv & Armon, 2022). In this paper, we aim to investigate the suitability of DL on administrative tabular data relative to traditional machine learning model, in terms of the effects of the size and complexity of the data as well as the effects of feature generation versus that of feature selection and engineering.

2. Methodology

2.1 Data

The dataset used in the experiments was curated from the datasets available on OpenDOSM (Department of Statistics Malaysia, 2023). OpenDOSM is a platform that catalogues, visualises and analyses the administrative data collected and collated by the Department of Statistics Malaysia. The data on OpenDOSM comprises primarily economic related data, as well as national healthcare data (primarily Covid-19 related). OpenDOSM was chosen as the source of data as data is updated regularly and is easily accessible using programming languages.

The 263 datasets available (as of February 17, 2023) was web-scrapped and accessed for both size (number of rows) and complexity (number of columns). Three datasets were selected for analysis, representing varied sizes and complexities, as shown in **Table 1.**

Table 1: Description of the datasets selected from OpenDOSM

Dataset	Size and complexity	Description	Target variable
Price Catcher	266435* rows and 4 columns	Documents the prices of key items according to the Ministry of Domestic Trade and Cost of Living for the month of August 2022	Price
Covid-19	19040 rows and 27 columns	Documentation of Covid-19 cases according to the Ministry of Health Malaysia with various breakdowns	Cluster cases
Exchange Rate	7019 rows and 27 columns	Exchange rates of various currencies	Ringgit Malaysia to Vietnamese Dong exchange rate

Commented [OLM1]: Why Aug 2022 - is this suggesting the last update/reference dataset is in Aug while other datasets like exchange rate doesnot contain reference on mm/yy? thought the price catcher documents daily prices

Commented [DHCC2R1]: It was the largest dataset at the point of writing

*During experimentation, the Price Catcher dataset was reduced to ten-percent of its actual number of rows of 2664356 to reduce the experimentation run time.

Commented [OLM3]: its

The rationale for selecting the three datasets above were to investigate the effects of the dataset size (number of rows) and complexity (number of columns). The columns chosen to be used as a target or class label were chosen both for utility (e.g. the price of items in price catcher) and their statistical properties. In this study, columns with the highest standard deviation were chosen as these columns provide valuable information and variation for machine learning models, but also requires careful pre-processing and model selection to ensure accurate and robust predictions.

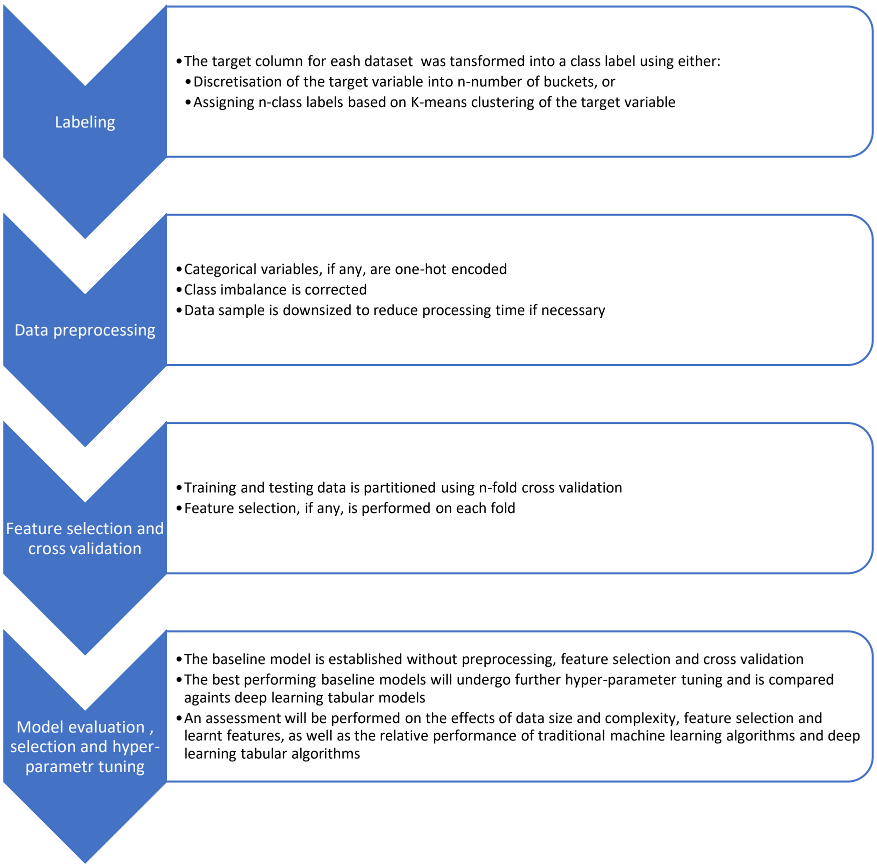
2.1 Empirical Design

A total of 169 experiments were conducted in the course of this study. Experiments were separated into distinct types, including baseline experiments for both traditional machine learning and followed by a comparison between optimised traditional machine learning and tabular deep learning models, and finally experiments conducted using reduced number of classes. **Table 2** shows the detailed description of the types of experiments conducted in this study.

Table 2: Description of each of the experimental type and the corresponding setup

Experiment type	Number of classes	Pre-processing and optimisation	Target variable
Baseline	30	No hyper-parameter tuning Imbalanced classes are not addressed Encoding of categorical variables Imputation of numerical variables	Price Class labels: K-means and discretised
Optimised	30	Imbalanced classes are addressed Optimised hyper-parameter Cross-validation	Cluster cases Class labels: K-means and discretised
Reduced classes	5	Imbalanced classes are addressed Optimised hyper-parameter Cross-validation	Ringgit Malaysia to Vietnamese Dong exchange rate Class labels: K-means and discretised

The baseline experiments were conducted to identify the best performing traditional machine learning algorithms, even without extensive pre-processing, hyper-parameter tuning and correction for class imbalance. For a fairer comparison, the optimised set of experiments were then conducted in order to optimise the machine learning's performance while taking into consideration the effects of class imbalance. The optimised experiments can be a good proxy for good feature selection that is performed by subject matter experts. Thus, the ability of deep learning tabular models to create useful features can be evaluated effectively. Finally, the number of classes were reduced from thirty to five to reduce effect of class imbalance, as well as to represent real-world decision support more accurately.
The detailed process flow of this study is shown in **Figure 1.**
Figure 1 shows the process flow of this study.



Labeling

- The target column for eash dataset was tansformed into a class label using either:
  - Discretisation of the target variable into n-number of buckets, or
  - Assigning n-class labels based on K-means clustering of the target variable

Data preprocessing

- Categorical variables, if any, are one-hot encoded
- Class imbalance is corrected
- Data sample is downsized to reduce processing time if necessary

Feature selection and cross validation

- Training and testing data is partitioned using n-fold cross validation
- Feature selection, if any, is performed on each fold

Model evaluation , selection and hyper-parametr tuning

- The baseline model is established without preprocessing, feature selection and cross validation
- The best performing baseline models will undergo further hyper-parameter tuning and is compared againts deep learning tabular models
- An assessment will be performed on the effects of data size and complexity, feature selection and learnt features, as well as the relative performance of traditional machine learning algorithms and deep learning tabular algorithms

3. Results

3.1 Baseline experiments

Table 3: Baseline experiment results

Dataset	Classifier	Accuracy	Balanced Accuracy	F1 Score
Price Catcher K-means Label	BaggingClassifier	1.00	1.00	1.00
Price Catcher K-means Label	XGBClassifier	1.00	1.00	1.00
Price Catcher K-means Label	RandomForestClassifier	0.99	0.96	0.99
Price Catcher K-means Label	Average	0.55	0.51	0.51
Price Catcher Discretise Label	DecisionTreeClassifier	1.00	1.00	1.00
Price Catcher Discretise Label	BaggingClassifier	1.00	1.00	1.00
Price Catcher Discretise Label	RandomForestClassifier	1.00	0.98	1.00
Price Catcher Discretise Label	Average	0.94	0.60	0.94
Covid 19 Cases K-Means Label	XGBClassifier	1.00	0.99	1.00
Covid 19 Cases K-Means Label	DecisionTreeClassifier	1.00	0.99	1.00
Covid 19 Cases K-Means Label	BaggingClassifier	1.00	0.99	1.00
Covid 19 Cases K-Means Label	Average	0.74	*0.34*	0.71
Covid 19 Cases Discretise Label	BaggingClassifier	1.00	0.94	1.00
Covid 19 Cases Discretise Label	DecisionTreeClassifier	1.00	0.91	1.00
Covid 19 Cases Discretise Label	LinearDiscriminantAnalysis	0.97	*0.48*	0.97
Covid 19 Cases Discretise Label	Average	0.92	*0.25*	0.91
Exchange Rate K-Means Label	XGBClassifier	1.00	0.97	1.00
Exchange Rate K-Means Label	BaggingClassifier	1.00	0.96	1.00
Exchange Rate K-Means Label	RandomForestClassifier	0.97	0.93	0.97
Exchange Rate K-Means Label	Average	0.67	0.64	0.66
Exchange Rate Discretise Label	DecisionTreeClassifier	1.00	0.94	1.00
Exchange Rate Discretise Label	LabelPropagation	0.95	0.93	0.95
Exchange Rate Discretise Label	LabelSpreading	0.95	0.93	0.95
Exchange Rate Discretise Label	Average	0.76	0.67	0.74

Table 3 shows the results of baseline experiments conducted. For each dataset, between 28-25 traditional machine learning algorithms were used to perform classification. The algorithms evaluated comprises tree-based models, linear models and Bayesian models (Pandala, 2023) .The result for the top three performing machine learning algorithms is shown together with the average performance of all the other machine learning algorithms. The three worst performing results are underlined and italicised.

Although the top three machine learning algorithms shows excellent results, these results could be the results of not controlling for class imbalance. It can be observed that the average performance for all classifiers is lower than the performance of the top three classifiers, indicating that not every machine learning algorithm is able to predict well in a scenario where class imbalance exists. Class imbalance is a situation where the distribution of class labels in the training data is unequal, with one or more class having a much smaller number of examples than the others. Machine learning algorithms will be biased towards classes with more examples and will perform more accurately for predicting such classes. This will skew the overall accuracy of the machine learning algorithm as it shows the proportion of correct predictions made by the model out of all the predictions made. The balanced accuracy considers the imbalanced nature of the dataset by calculating the average of the per-class accuracies. The F1-score harmonic mean of precision and recall, two metrics that are used to measure the accuracy of the model's predictions. Precision is the proportion of true positives (TP) out of all the positive predictions (TP + FP). It measures the model's ability to correctly identify positive examples. Recall is the proportion of true positives (TP) out of all the actual positive examples (TP + FN). It measures the model's ability to correctly identify all positive examples.

The other contributing factor to be considered would be the high degree of collinearity present in the datasets chosen, especially the Exchange Rate and Covid-19 datasets. As such, our discussion shifts to the impact of size of dataset on the average performance of machine learning classifiers across datasets. As the size of the dataset increases from the Exchange Rate dataset to the Covid 19 dataset and eventually the Price Catcher dataset, it can be observed that the average results across all measures have increased with size. It can also be observed that the datasets with class labels that resulted from discretised binning returned better results as compared to datasets with class labels that resulted from the clustering technique.

3.2 Optimised Experiments

In Optimised Experiments, the top performing traditional machine learning models were chosen to be further optimised using data pre-processing and hyper-parameter tuning. We intend to compare these models to deep learning tabular models which can create its own features versus that of features chosen by the machine learning model which are then further optimised using hyper-parameter tuning.

Figure 2 & **Figure 3** shows the Accuracy and F1-score for the Optimised set of machine learning algorithms and a Category Embedding Model, which is one example of a deep learning tabular model.

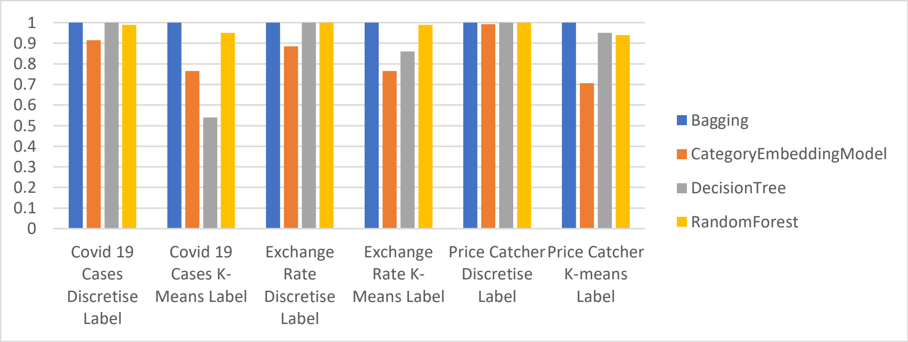


Figure 2: F-1 score for optimised traditional machine learning and a Category Embedding Model

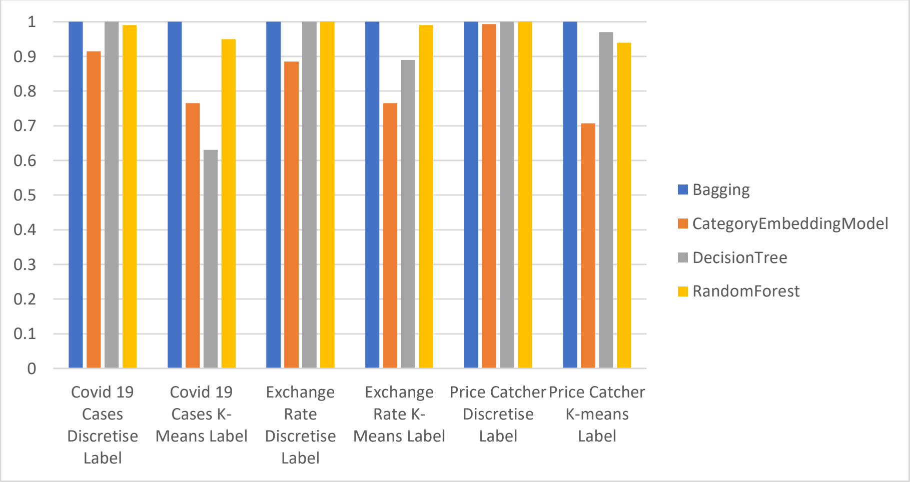


Figure 3: Accuracy score for optimised traditional machine learning and a Category Embedding Model

A Category Embedding Model (CEM) is a feed-forward neural network with embedding layers for the categorical columns. The configuration used is a three-layer neural network with a leaky relu activation function, with a small learning rate of 1e-3. Batch size is set at 32, with maximum epochs of 100. The number of nodes in each layer is 4096-4096-512.

We can observe that optimised traditional machine learning algorithms has out-performed the Category Embedding Model (CEM) in five out of six set datasets (for both F1-score and Accuracy), while in one dataset the CEM performed better than the Decision Tree model. This set of results is due to the relative lack of categorical features in the dataset chosen as the dataset are mostly numerical in nature. An improvement might be achieved by one-hot encoding of the categorical columns in the Price Catcher dataset.

3.3 Reduced classes

The experiments conducted in this phase can be thought of as an ablative step. The initial choice of choosing 30 target classes was done deliberately as a means of coercing an artificial state of class imbalance. In the reduced classes set of experiments, the number of classes was reduced to five classes to reduce class imbalance. In addition to that, we evaluated a Gated Additive Tree Ensemble (GATE) (Joseph & Raj, 2023) tabular deep learning model, as it learns how to represent features using a gating unit like Gated Recurrent Units (GRU), which can also select the most relevant features. It also uses an ensemble of non-linear trees that can be differentiated and adjusted with self-attention to make the final prediction. The GATE model used in this study consisted of 10 trees used in a bagging mode.

The same set of optimised machine learning algorithms used in the previous set of experiments were evaluated. The results of the experiments performed is shown in **Table 4.**

Table 4 : Results of applying Optimised traditional machine learning algorithms and GATE on reduced number of classes in dataset

Dataset	Classifier	Accuracy	F1 Score
Price Catcher K-means Label	Bagging	1.00	1.00
Price Catcher K-means Label	RandomForest	1.00	1.00
Price Catcher K-means Label	DecisionTree	1.00	1.00
Price Catcher K-means Label	GatedAdditiveTreeEnsemble	0.95	0.95
Price Catcher K-means Label	CategoryEmbeddingModel	0.98	0.98
Covid 19 Cases K-Means Label	Bagging	1.00	1.00
Covid 19 Cases K-Means Label	RandomForest	1.00	1.00
Covid 19 Cases K-Means Label	DecisionTree	1.00	1.00
Covid 19 Cases K-Means Label	GatedAdditiveTreeEnsemble	0.96	0.96
Covid 19 Cases K-Means Label	CategoryEmbeddingModel	0.94	0.94
Exchange Rate K-Means Label	Bagging	1.00	1.00
Exchange Rate K-Means Label	RandomForest	1.00	1.00
Exchange Rate K-Means Label	DecisionTree	1.00	1.00
Exchange Rate K-Means Label	GatedAdditiveTreeEnsemble	1.00	1.00
Exchange Rate K-Means Label	CategoryEmbeddingModel	0.97	0.97

Both the CEM and GATE is able to produce results which matches or nearly matches the performance of optimised traditional machine learning models. As a comparison between tabular deep learning models, it can observed that GATE performed better than CEM for two out of tree datasets.

4. Discussion and Conclusion

The experiments conducted during this study has shown that tabular deep learning models has potential to be used in performing classification tasks as a complement to traditional machine learning models. In scenarios where granular alternative data is available without subject matter expertise, deep learning tabular models may be useful.

In terms of assessing the effects of size and complexity of the dataset on model performance , specific examples are observed. An optimised Decision Tree model will benefit from more data , as its accuracy improved when comparing the K-means labelled Exchange Rate, Covid 19 and Price Catcher datasets. For the Optimised scenario, the CEM showed improvement as the data set size increased for both accuracy and F1 score. In the Reduced classes scenario, the GATE model also showed improvement as the dataset size increased for both accuracy and F1 score. This, however, cannot be taken as a generalisation that increasing the size of data will guarantee performance improvement as it is highly dependent on the model chosen.

In conclusion, the usage of deep learning tabular models on administrative data is a viable solution in lieu of subject matter expertise for feature selection and engineering. For future work, usage of deep learning tabular models for well-defined problems may be explored, as well as the usage of deep learning tabular models for multi-modal problems (AutoGluon, 2023) (combining text and images for example) as administrative data becomes more varied.

Commented [OLM4]: Assessing?

Commented [DHCC5R4]: Indeed, changed

Commented [OLM6]: Specific examples are observed - not sure what is meant? Does it means that results are mixed across different models?

Commented [DHCC7R6]: Yes that is accurate. I did put a disclaimer in the last sentence to not generalize this finding

Commented [OLM8]: its

Commented [DHCC9R8]: Changed!

Commented [OLM10]: Suggest to standardise - ie dataset or dataset or data-set instead of usage of all terms in the same paper

Commented [DHCC11R10]: Noted!

References

Araujo, D., Bruno, G., Marcucci, J., Schmidt, R., & Tissot, B. (2022). Machine learning applications in central banking. *IFC-Bank of Italy Workshop on 'Data Science in Central Banking', Part 1: Machine Learning Applications*. IFC-Bank of Italy Workshop on 'Data Science in Central Banking', Part 1.

Arik, S. O., & Pfister, T. (2020). *TabNet: Attentive Interpretable Tabular Learning* (arXiv:1908.07442). arXiv. https://doi.org/10.48550/arXiv.1908.07442

AutoGluon. (2023). *Multimodal Data Tables: Tabular, Text, and Image—AutoGluon Documentation 0.7.0 documentation*. https://auto.gluon.ai/stable/tutorials/tabular_prediction/tabular-multimodal.html

Bender, S., Blaschke, J., & Hirsch, C. (2022). Data Production in a Digitised Age: The need to establish successful workflows for micro data access. *Technical Report*.

Burnap, P., & Williams, M. L. (2015). Cyber Hate Speech on Twitter: An Application of Machine Classification and Statistical Modeling for Policy and Decision Making. *Policy & Internet*, *7*(2), 223–242. https://doi.org/10.1002/poi3.85

Department of Statistics Malaysia. (2023). *OpenDOSM*. https://open.dosm.gov.my

Dessaint, O., Foucault, T., & Frésard, L. (2021). *Does Alternative Data Improve Financial Forecasting? The Horizon Effect* (SSRN Scholarly Paper No. 3784024). https://papers.ssrn.com/abstract=3784024

Gorishniy, Y., Rubachev, I., Khrulkov, V., & Babenko, A. (2021). *Revisiting Deep Learning Models for Tabular Data* (arXiv:2106.11959). arXiv. https://doi.org/10.48550/arXiv.2106.11959

Huang, X., Khetan, A., Cvitkovic, M., & Karnin, Z. (2020). *TabTransformer: Tabular Data Modeling Using Contextual Embeddings* (arXiv:2012.06678). arXiv. https://doi.org/10.48550/arXiv.2012.06678

Joseph, M., & Raj, H. (2023). *GATE: Gated Additive Tree Ensemble for Tabular Classification and Regression* (arXiv:2207.08548). arXiv. https://doi.org/10.48550/arXiv.2207.08548

Malaysia Administrative Modernisation and Management Planning Unit (MAMPU). (2023). *Data.gov.my*. https://www.data.gov.my/data/ms_MY/dataset

Pandala, S. R. (2023). *Lazy Predict* [Python]. https://github.com/shankarpandala/lazypredict (Original work published 2019)

Popov, S., Morozov, S., & Babenko, A. (2019). *Neural Oblivious Decision Ensembles for Deep Learning on Tabular Data* (arXiv:1909.06312). arXiv. https://doi.org/10.48550/arXiv.1909.06312

Roy, S., & Ghosh, P. (2020). Factors affecting COVID-19 infected and death rates inform lockdown-related policymaking. *PLOS ONE*, *15*(10), e0241165. https://doi.org/10.1371/journal.pone.0241165

Shwartz-Ziv, R., & Armon, A. (2022). Tabular data: Deep learning is not all you need. *Information Fusion*, *81*, 84–90. https://doi.org/10.1016/j.inffus.2021.11.011

Song, W., Shi, C., Xiao, Z., Duan, Z., Xu, Y., Zhang, M., & Tang, J. (2019). AutoInt: Automatic Feature Interaction Learning via Self-Attentive Neural Networks. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 1161–1170. https://doi.org/10.1145/3357384.3357925

Sun, B., Yang, L., Zhang, W., Lin, M., Dong, P., Young, C., & Dong, J. (2019). *SuperTML: Two-Dimensional Word Embedding for the Precognition on Structured Tabular Data* (arXiv:1903.06246). arXiv. https://doi.org/10.48550/arXiv.1903.06246