Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

A STUDY TO DETERMINE THE RESILIENCE OF THE OPTIMAL ALGORITHM UNDER VARIOUS CONDITIONS

JIWEI YAN
Research Scholar of Lincoln University College Malaysia.

Dr. MIDHUN CHAKKARAVARTHY
Associate Professor, Dean, Faculty of Computer Science and Multimedia, Lincoln University College.

Dr. SANDEEP SHRESTHA
Professor of Lincoln University College Malaysia.

Abstract

E-commerce forecasting relies on categorising whether or not a customer's visit to an online store results in a purchase. When a big German clothing shop categorises its clients and displays gift cards in front of those who aren't buying anything to nudge them into making a purchase, its increased turnover. To make such forecasts, a wide range of prediction models and data sources are available. For the purpose of determining how consumers may best be categorised as purchasing or not buying, this study seeks to retrieve well-suited prediction models and compare their performances across diverse data kinds, like static and dynamic data Research conducted using the Cross Industry Standard Process for Data Mining (CRISP-DM) formed the basis of this paper. There were several suitable models identified through literature study, including boosted trees such as the RF and RNN, as well as Support Vector Machines such as the SVM and the FNN, as well as Logistic Regression and Recurrent Neural Networks. Algorithms were then trained on three separate datasets, the sequential session data, the static customer data, and a combined dataset, before being assessed and contrasted using several performance criteria, including prediction latency and comprehensibility.. After that, the RNN was trained on datasets varied in the amount of feature engineering that was necessary. Python was used to create all algorithms, evaluations, and comparisons. According to the acquired data, the RF was the most effective while exhibiting a tolerable prediction delay. There was no difference between the algorithms in terms of comprehensibility. Customer information has a little impact on the performance of a combined dataset, which yields the best outcomes. ROC AUC values for various datasets and methods are included in a table for comparison's sake. The RNN also showed a promising effect in terms of time-consuming feature engineering, where fewer and less designed features produced better results than a greater number of more extensively engineered features utilised in the other algorithms.

Keywords: RF, evaluations, comparisons, Logistic Regression, Recurrent Neural Networks

1. Introduction

It was suggested in the literature to run the models on a variety of datasets: Research has demonstrated that a combined dataset of customer and clickstream data is more accurate.

When it comes to estimating the likelihood of a customer making a purchase, data is more accurate than independent datasets. The dynamic clickstream data's already excellent performance can only be marginally improved by using static customer data (Bogina, Kuflik, & Mokryn, 2016; Lee, Ha, Han, Rha, & Kwon, 2015; Poggi, Moreno, Berral, Gavald,

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

& Torres, 2007). Using an exploratory data analysis, this thesis tries to replicate the findings of prior literature while also giving new insight into the predictive value of the various datasets. It is hoped that this study will explain why static customer data performs so poorly when compared to dynamic clickstream data.

Lang and Rettenmeier (2017) used an RNN to predict the likelihood of a purchase occurring during a web-shop session. Research argues that RNNs, as stateful models with a memory, are capable of decreasing labor-intensive feature engineering when applied to sequential data sets. As a result, Lang and Rettenmeier fail to demonstrate whether or not less feature engineering produces as excellent results as features that have been substantially engineered. This is a major omission from their paper. This work seeks to train an RNN on datasets with different levels of feature engineering, in order to determine how well RNNs perform on less engineered features, as sequential clickstream data appears to be a good fit for RNNs. Using RNNs, it is possible to decrease feature engineering, which is critical for e-commerce because it is a time-consuming operation that takes a great deal of expertise.

A last step is to assess the models' resilience, which shows how varied situations (such the type of device being utilized) might impact their performance. Even while it's crucial to know how a model would perform in real-world scenarios, no previous research has looked at how these results would change under different circumstances, even though this is critical for understanding how the model would work in practice.

In light of the information presented in Section 1.1, the purpose of this thesis is to identify and apply machine learning models for classifying web store visitors as aborting or not aborting, referred to as "no buying" and "purchasing sessions," respectively.

Different data sets, including clickstream data generated by each online store visitor and customer information if a visitor could be recognized, are used to apply these models. This is done in order to determine which model and data are most suitable for forecasting whether an online shopping session will result in a purchase.

It's all about performance, latency, and ease of use. Use cases require real-time prediction and model comprehensibility since the demand for explaining machine learning models' judgments is increasing. Latency is significant. The German clothes retailer's boosted tree model will serve as a benchmark for the comparison of the various algorithms. An exploratory data analysis of clickstream and static customer data is performed to give more information and reasoning about the differing performances on different datasets.

Stateful machine learning models, like RNNs, are trained on datasets that require less feature engineering in order to prove if stateful models can produce decent results while lowering the need to feature engineer.

Once all datasets have been analyzed, the models are evaluated for their ability to withstand a variety of circumstances. The gender of the visitor or the gadget on which the

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

web shop was accessed is examples of such criteria. Using this analysis, we can predict how well the models will perform in the field once they've been put to use in the real world.

If a tested model outperforms the baseline model, then deployment is just an inference of this study.

Chapter 3 begins with a review of the literature, focusing on comparable tasks to the one at hand in Chapter 2, before moving on to examine the conclusions of this study. Chapter 3 goes into detail on the methods, including the study methodology, the algorithms and software that were used, and the metrics that were used to gauge their effectiveness. The data Chapter 4 discusses data preprocessing, a first data analysis, and the utilized characteristics following the approach Chapter 3. Chapter 5 goes into detail about the implementation, including hyper-parameter tweaking. Chapter 6 presents the findings, and Chapter 7 discusses them. In Chapter 8, the thesis concludes by answering all of the above-mentioned questions.

2. Literature Review

Diffusion techniques (DTs) use a series of split conditions to break down a large population into smaller, more homogenous subpopulations. Create the most diverse subgroups possible. The best splits may be found using a variety of algorithms, such as the Hunts algorithm, which uses a greedy method to discover the best splits (P. Tan et al., 2005). Converting DTs into categorization rules that are easy to grasp is a benefit of simple DTs (Han et al., 2011). As the models become larger and more imbalanced, this comprehensibility declines (Rokach & Maimon, 2014). DTs can learn and predict things quite quickly. In spite of their differing comprehensibility, black box models like FNNs and SVMs are still more difficult to comprehend than the diverse types of models (Rokach & Maimon, 2014). Features that must be developed, and the time it takes to implement them.

Temporal sequences cannot be implicitly modeled and the complexity of trees with multiple category variables increases. In noisy data, methods that simply use solitary DTs, referred to as "non-ensemble DTs," tends to over-fit and be unstable (Hop, 2013).

This approach is part of an ensemble of DTs, which includes more than one DT. Using the preceding tree's prediction residuals, a chain of trees is constructed (Friedman, 2002). The baseline model employed in this study is an example of this type of approach. Numerous Kaggle machine learning competitions (Kaggle – The Home of data science and machine learning, 2017) have shown that boosted DTs are an extremely strong tool for predictive analytics, but they are less understandable than plain DTs since they consist of many trees.

This is another example of ensemble trees, in which enormous trees are fitted to bootstrap re-sampled data and are voted on in a majority vote (Breiman, 1996). De-correlating the trees improves RF over Bagging. During each split, a random subset of characteristics is selected and used exclusively in the subsequent split. In this case, the outcomes are

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

based on a simple majority of the trees' votes. Improve on non-ensemble DTs' drawbacks, such as robustness and overfitting, by utilising a large number of classifiers, such as Bagging and RFs (Bagging). Training is quicker, but they take longer to make predictions (Breiman, 2001). Bagging and RFs, on the other hand, still rely on feature engineering and can't account for time-dependent variables.

DTs have been used extensively in the literature to solve problems that are quite similar to this one. Many different algorithmic approaches have been developed to identify trading sessions as either buying or non-buying. Additional item sales information were employed in addition to clickstream data, which improved forecast accuracy. Bagging RepTree was the best option for our use case when just clickstream data was available. Accuracy was 0.824%, with a recall rate of 0.808, an F1-score of 0.806%, and an AUC of 0.889, using Weka, an open source machine learning tool (Frank, Hall, and Witten, 2016), which facilitates data analysis using several machine learning approaches.

Poggi et al. (2007) used clickstream data translated into HMCs to train several machine learning models. Fewer than 14,000 transactions.

This model was trained using tions. Static session information and dynamic session information were examined separately by the authors. Static and dynamic session data only marginally improved the prediction performance. J48's DT classification outperformed an LLR classifier with an accuracy of 0.708%, whereas the LLR classifier had an accuracy of 0.681%, according to the data.

It was a task for participants in the Prudsys Datamining Cup 2013 to categorise clickstream data (DMC 2013, 2013) into purchasing vs no-purchase sessions using a dataset including 429,013 data points. In the end, a bag of 600 C4.5 DTs let the University of Dortmund's winning team reach an accuracy of 0.972. (Hop, 2013). An FNN and an SVM were also tested at the Prudentia Datamining Cup 2013 by Hop. An accuracy of 0.903 surpassed the other two techniques, with the SVM in second place with an accuracy of 0.868, and the FNN in third with an accuracy of 0.755. Hop also cites the RF method's other advantages, such as its great precision. As compared to SVMs, the training effort is low and the hyper-parameter adjustment is modest, making it simple and rapid to implement.

3. Research Gap

For machine learning models, feature engineering is the process of identifying and extracting certain features from data that describe the underlying problem. But even if it is not a well-defined procedure, it is nevertheless a key phase for a machine learning model's growth. It doesn't matter how sophisticated a model is, if the input features aren't representational of the job, it can't forecast accurately. An issue with feature engineering is that it is typically over-specified and incomplete (Castanedo et al., 2014).

A total of 28 features from the clickstream data and a total of 20 features from the customer data were used in this study to capture session characteristics. Table 4.1

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

displays all 48 characteristics, broken down by sequence and client. There are feature names, a brief description of the data type, and an option to allow a feature to be null able in the table below. In addition, it is specified which properties remain constant across a session. Features with non-constant values describe events that happened in the preceding stage. A feature set variable cannot include the green variable since it is the target variable.

Aside from the RNN, none of the algorithms that will be built can simulate time sequences. A number of specific features were built to capture the time-dependent data, such as the sessiondauer and the delta last page features. As a result, the feature set for these algorithms is far more complex than the RNN's, and includes sums, averages, maximum and minimum computations. Each following step in a session comprises information on the preceding stages, such as the total number of visited pages, the total duration up to this point, the total number of viewed article detail pages, and so forth. All of this is unnecessary with an RNN because it takes in full sequences as input. This led to the creation of a second simplified feature set that still includes all of the customer's desired features, but with a reduced number of sequence features. Only the type of visitor interaction, the visited page, and a few more characteristics providing page information like the article price are included. A lot of the time, these features don't exist since they're only relevant to a specific sort of website. AdaP, an article's price, is only calculated if the article's detail page is accessed. From these, you can get all the other features. Because the feature set is limited in scope, it is possible to get closer to the real clickstream data by employing minimum feature engineering, hence avoiding the need to explicitly model variables. Training and testing on normal and RNN datasets will be done in order to evaluate if RNNs can learn all other features on their own, reducing feature engineering work.

This dataset's sequence and completeness yielded extremely strong performance results for most algorithms, providing a solid foundation on which management and decision makers may base future business choices. One can notice that certain algorithms performed better on all datasets than others while analyzing the performance of the three datasets. Algorithms in Chapter 3 highlighted precision and ROC AUC as most significant metrics, and the Random Forest (RF) fared best on both datasets with the second-highest precision and the greatest ROC AUC. As a result, it is the best algorithm in terms of performance, achieving the highest scores across all datasets. Other studies, such as Hop (2013) and Niu et al. (2017), found that the RF outperformed other methods, as discussed in Chapter 2, and these results support those findings. This approach was the second best in terms of accuracy and ROC AUC on the entire and sequence dataset.

4. Research Objective & Methodology

Owing to the fact that this research was conducted on a desktop computer rather than a server due to the lack of infrastructure and data protection requirements, this section outlines the study's limitations.

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

The quantity of training data, the creation of the padded sequence data for RNN input, training iterations, and hyper-parameter tuning were all affected by the memory and computing limits of a local machine. The employed training data, notably for the FNN and RNN, which require vast quantities of data to correctly discover patterns in the data and carry out excellent predictions, was insufficient. Training durations were too long since additional data would have led to higher prediction outcomes. The RNN sequence was created in accordance with strict parameters. Aside from being too big to retain in the computer's memory, the padded sequences were saved on disc after processing every 5,000 sequences because of this limitation. Since the RNN could no longer handle larger sequences, there was no way to use sequences longer than 20 time steps. As a result, it was not able to investigate whether or not including lengthier sequences had resulted in more accurate forecasts. While additional training iterations yielded higher predictions, they also required longer training durations for all algorithms. Since FNN and RNN require a large number of training epochs, this had the most impact on them.

To enable for comparison with the baseline model, all of the study's data required to be identical to that of the baseline model. As a result, we were unable to look at new feature engineering options. Additional feature engineering may have enhanced the outcomes, as is conceivable. One may have used more data, such as click locations and hover durations, like in the work of Niu et al (2017). In addition, because the customer dataset's training set was likely too small for all algorithms, it would have been preferable to use the same number of data samples as were used in the complete and sequence dataset, even if this meant utilizing weeks of additional data that were not used in the complete and sequence dataset.

If the sequence dataset had only included sequential, time-dependent data, the comparison between the sequence and static data would have been clearer. Session data with both static and sequential properties were characterized by the sequence dataset. A separation between session and customer data is possible, although this does not clearly differentiate sequential and static data. For this reason, it is necessary to further delineate the effects of sequential and static data. There is a potential that findings on the sequence dataset might have been just as excellent or even better if hyper-parameter tweaking had been applied to this dataset.

Finally, the present model's implementation, training, and deployment need predictions based on a single session. As a result, the scope of this investigation was limited to session-based predictions alone. It's a shame, but this strategy has significant drawbacks. For example, purchases that could be made at a later time can be perfected by showing a gift card with session-based predictions. Therefore, the system accurately recognizes a no-buying session if someone is now merely looking because he expects to make one purchase over the weekend. Thus, a gift card is presented, but it can only be used during the current session it is displayed on. Rather than wait until the weekend, the visitor is likely to make a purchase right now. In spite of what appears to be a successful forecast, profit was taken away from the weekend when a visitor would have purchased

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

without needing a gift card, even though that day's sales soared. Identifying visitors who do not intend to make a purchase during the current session or in the near future can help prevent the so-called "front-loading" effect from occurring.

5. Data Analysis & Findings

Recommendations for further research are derived from the above-mentioned constraints. As a first step in any follow-up research, it is important to conduct the different tests instead of locally on a regular desktop PC on a more computationally intensive machine. Parallelization is a feature of all algorithms, which may also be used on a more powerful system. Using additional data, tweaking hyper-parameters, and performing more training rounds can already assist to generate more powerful models, which can then be used to better understand and utilize the algorithms' potential.

It would be interesting to see if incorporating additional characteristics, such as those that take into account things like click information, improves the accuracy of forecasts. It is also possible to focus on utilising more information from the data that is already available to determine whether performance can also be improved by enhancing the data basis rather than adjusting and experimenting with various methods. In addition, future research should discriminate between sequential and static customer data in order to truly establish the distinction between sequential and static consumer data in the long run.

Finally, it would be fascinating to explore how stateful models like RNNs might further reduce feature engineering and the use of static customer data. Furthermore, the findings already acquired, which suggest RNNs function well on less designed characteristics of a sequential nature, were already quite encouraging. An experiment might be conducted exclusively on this study's comparison of RNNs trained on data containing the same information as the RF method but with a different amount of features, as was done. A server is needed here to train the RNN, so that more data and more epochs may be utilized. It is possible that conserving time and human brain capacity during feature engineering would result in longer training durations and more training data to extract essential features implicitly and obtain similar results to the RF trained on the whole dataset. In order to do this, a more extensive and perhaps simultaneous training on more powerful equipment is needed.

6. Conclusion

The performance of the models is influenced by the quality of the input data. The poorest outcomes were shown in the carefully manufactured datasets, which were used as input for the stateless machine learning models. Even when paired with sequencing data, it might only have a little impact on performance. When it comes to forecasting online purchases, sequence data is the most essential. As a result, future work on feature engineering should concentrate on sequence data instead. If privacy issues and computational complexity are a problem, it is possible to limit the feature set to only the sequence data in order to lessen the computing burden. Training the RNN on qualities

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

that were less designed made the relevance of sequential data much more apparent. RNN fared better on the sequential data than the combined dataset in this situation It is clear that stateful machine learning models should only be trained on sequential data in this scenario, as the customer data actually hampered the model's performance.

The best-suited algorithm, as mentioned in the discussion of the findings, was assessed based on the days of the week, the used device, and the gender of the consumer. The algorithm was shown to perform well across a wide range of situations, with no significant changes in performance amongst the investigated variables. Even if you don't know your customers' gender, you can still get the greatest results by omitting it.

Boost the accuracy of predictions. Furthermore, it is beneficial for decision-makers to understand how different parameters might affect the accuracy of forecasts.

References

- Jain, A. (2016, 3). Complete guide to parameter tuning in xgboost (with codes in python). Retrieved from https://www.analyticsvidhya.com/blog/2016/03/ complete-guide-parameter-tuning-xgboost-with-codes-python/
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112). New York: Spring.
- Jolliffe, I. (1982). A note on the use of principal components in regressio. Journal of the Royal Statistical Society, 31(3), 300 – 303.
- Kaggle – the home of data science and machine learning. (2017). Retrieved from https://www.kaggle.com/
- Korpusik, M., Sakaki, S., Chen, F., & Chen, Y. (2016). Recurrent neural networks for customer purchase prediction on twitter. In Proceedings of the CBRecSys.
- Kotu, V., & Deshpande, B. (2014). Predictive analytics and data mining: concepts and practice with rapidminer. Morgan Kaufmann, 562–572.
- Lang, T., & Rettenmeier, M. (2017). Understanding consumer behavior with recur- rent neural networks. In Proceedings of the 3rd International Workshop on Machine Learning Methods for Recommender Systems.
- Lee, M., Ha, T., Han, J., Rha, J., & Kwon, T. (2015). Online footsteps to purchase: Exploring consumer behaviors on online shopping sites. In Proceedings of the ACM Web Science Conference.
- Liu, Y., an, A., & Huang, X. (2006, April). Boosting prediction accuracy on imbal- anced datasets with svm ensembles. PAKDD, 6, 107 – 118.
- Lo, C., Frankowsik, D., & Leskovec, J. (2014). Understanding behaviors that lead to purchasing: A case study of pinterest. In Proceedings of the KDD 16, San Francisco, CA, USA.
- Magrabi, A. (2016, 11). Top 5 machine learning applications for e- commerce. Retrieved from https://techblog.commercetools.com/top-5-Machine-learning-applications-for-e-commerce 268eb1c89607
- Manning, C., Raghavan, P., & Schuetze, H. (2008). Introduction to information retrieval. Cambridge University Press.

Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 55 Issue: 11: 2022
DOI10.17605/OSF.IO/W3XS2

- Moe, W. (2003). Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream. Journal of Consumer Psy- chology, 13(1–2), 29–39.
- Mulder, W. D., Bethard, S., & Moens, M.-F. (2015). A survey on the application of recurrent neural networks to statistical language modeling. Computer Speech & Language, 30(1), 61 – 98.
- Nakayama, Y. (2009). The impact of e-commerce: It always benefits consumers, but may reduce social welfare. Japan and the World Economy, 21(3), 239–247.
- Neural network hyper parameters. (2015, 12). Retrieved from http://colinraffel.com/wiki/neural network hyper parameters
- Niu, X., Li, C., & Yu, X. (2017). Predicitive analytics of e-commerce search behavior for conversion.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O. . . . Duchesnay, E. (2011). Scikit-learn: machine learning in python. Journal of Machine Learning Research.
- Piatetsky-Shapiro, G. (2014). Kdnuggets methodology poll.
- Poggi, N., Moreno, T., Berral, J., Gavald, R., & Torres, J. (2007, 7). Web customer modeling for automated session prioritization on high traffic sites. International Conference on User Modeling.
- A practical introduction to deep learning with caffe and python. (2016, 6). Re- trieved from http://adilmoujahid.com/posts/2016/06/introduction-deep-Learning-python-caffe/Python. (2017). Python website. Retrieved from https://www.python.org/
- Random forest classifier. (2017). Retrieved from http://scikit-learn.org/stable/ modules/generated/sklearn.ensemble.RandomForestClassifier.html
- Retail ecommerce in germany: A major digital market growing in size and sophistication. (2017, 7). Retrieved from https://www.emarketer.com/ Report/Retail-Ecommerce-Germany-Major-Digital-Market-Growing-Size-Sophistication/2002102
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should i trust you?: Explaining the predictions of any classifier.
- Rnn. (2018). Retrieved from https://keras.io/layers/recurrent/
- Rokach, L., & Maimon, O. (2014). Data mining with decision trees: theory and applications. World scientific.
- R: The r project for statistical computing. (2017). Retrieved from https://www.r-project.org/
- Russell, S., & Norvig, P. (1995). Artificial intelligence - a modern approach. Prentice- Hall, Englewood Cliffs: Artificial Intelligence.
- Salehi, F., Abdollahbeigi, B., Langroudi, A. C., & Salehi, F. (2012). The impact of website information convenience on e-commerce success of companies. Procedia Social and Behavioral Sciences, 57, 381–387.
- Scalable and flexible gradient boosting. (2016). Retrieved from http://xgboost.readthedocs.io/en/latest/
- The sequential model api. (n.d.). Retrieved from https://keras.io/models/ sequential/ Sk-learn. (2017). Sckit-learn website. Retrieved from http://scikit-learn.org/ stable/
- Srivastava, T. (2015, 6). Tuning the parameters of your random forest model. Retrieved from https://www.analyticsvidhya.com/blog/2015/06/tuning-random-forest-model/