PROCEEDINGS OF
INTERNATIONAL CONFERENCE ON ADVANCED TECHNOLOGIES
https://proceedings.icatsconf.org/
11th International Conference on Advanced Technologies (ICAT'23), Istanbul-Turkiye, August 17-19, 2023.

The Effects of Data Standardization and Normalization Techniques in Click Through Rate Prediction

Alper Yıldız[1], Müslüm Ersel Er[1], Ahmet Bursalı[1], Talha Çolakoğlu[1], Ekin Can Erkuş[1]

[1] *Intelligent Application Department DC, Huawei Technologies Turkey R&D Center Istanbul, Turkey*

alper.yildiz@huawei.com, ORCID: 0009-0002-0646-591X
muslum.ersel.er1@huawei.com, ORCID: 0000-0002-8232-6965
ahmet.bursali@huawei.com, ORCID: 0000-0001-7050-769X
talha.colakoglu2@huawei.com, ORCID: 0000-0002-4524-862X
ekin.can.erkus2@huawei.com, ORCID: 0000-0002-2445-5929

***Abstract*— Accurate ranking is critical for the user experience as well as applications such as information retrieval, recommender systems, and decision-making. To transform data into a common scale or distribution, standardization and normalization techniques are used. The purpose of this paper is to look into the effects of various data standardization and normalization techniques on ranking performance in order to improve performance or reduce computational complexity. It examines methods such as z-score standardization, min-max scaling, and robust scaling in existing literature and experimental studies. The paper assesses their impact on various ranking algorithms and models using benchmark datasets and discusses the benefits, limitations, and trade-offs associated with each technique, taking into account factors such as data distribution characteristics, outliers, and interpretability. The results can aid in the selection of the best normalization and standardization techniques for ranking tasks, particularly in recommender systems.**

***Keywords:* learning to rank, search optimization, standardization, normalization**

I. Introduction

The importance of effective ranking in shaping the user experience across a wide range of areas, including information retrieval, recommendation systems, decision making processes, is paramount in an era of massive digital in- formation. The reliability of ranking models is becoming key to delivering suitable and personalized results as traffic on search engines and digital platforms with recommender systems through the increasing number of data sources. Therefore, learning to rank studies have become a research and practical application area. Standardization and normalization techniques have gained a great deal of attention in data analysis and machine learning, aiming at facilitating the

This study compares the major methods of data standardization and normalization in order to reach our research objectives. In particular, we are examining standardization methods such as z-score, Manhattan, Peldschus, standard deviation, vector, Zavadskas Turskis log standardization, and scaling methods such as min-max, maximum absolute, robust scaler. We analysed the current literature with one of the benchmarking datasets. Various deep ranking models evaluated in this study to ensure the effect of standardization methods in neural network models. The findings of these studies are expected to have benefits for data modelling, neural networks and ranking applications.

Our study aims to provide practical advice for researchers and practitioners in selecting the appropriate approaches when performing standardization and normalization. In the area of recommender systems, where personalized and precise recommendations are essential in order to improve user satisfaction and engagement, this guidance is especially useful.

II. Methods

Methods used in this study uses both scaling and standardization strategies. While some of the methods do not change the distribution but only scale the data with a ratio, other methods try to convert the data to normal distribution.

A. *Min-Max Scaling*

Also called as Weitendorf’s linear standardization, scales values between 0 and 1.

$$x_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \tag{1}$$

B. *Maximum Absolute Scaling*

Maximum absolute scaling is dividing each value in the series to the maximum absolute value. By this method, maximum value of scaled series either become 1 or -1.

$$x_i = \frac{x_i - \min(x_i)}{|max(X)|} \tag{2}$$

C. *Z-score Standardization*

This method is named after its robustness to outliers which is provided by squeezing values between first and third quantile values of the series. Therefore, this method works better when outliers and skewness appear in the data.

$$x_i = \frac{x_i - \text{median}(X)}{Quantile(0.75) - Quantile(0.25)} \tag{3}$$

D. *Z-score Standardization*

The most common standardization method where the data approximates to the standard normal distribution with mean value equal to 0 and variance value equal to 1 after the method is applied.

$$x_i = \frac{x_i - \text{mean}(X)}{std(X)} \tag{4}$$

E. *Manhattan Standardization*

The Manhattan Standardization, also called as Manhattan Norm, convert vectors to a form that, if all the values are positive, sum of instances be equal to 1.

$$x_i = \frac{x_i}{\sum_{j=1}^{n} |x_j|} \tag{5}$$

F. *Standard Deviation Standardization*

This approach is the process of converting the values into a specific standard form by dividing the instances by standard deviation of the series. Therefore, method transforms series' standard deviation to 1 and facilities the analysis of data by making them comparable regardless of their scales.

$$x_i = \frac{x_i}{std(x_i)} \tag{6}$$

G. *Vector Standardization*

It is the process of dividing the values by square root of the sum of square values. This method is a kind of scaling but the scale ratio depends on the values of the series.

$$x_i = \frac{x_i}{\sqrt{\sum_{j=1}^{n} x_j^2}} \tag{7}$$

H. *Peldschus' Nonlinear Standardization*

Peldschus' nonlinear standardisation is used for purposes such as correcting the distribution of data, making it more resistant to outliers or highlighting certain features. It transforms data points not with a linear relationship, but with a certain non-linear transformation and changing the series' distribution [1].

$$x_i = \left(\frac{x_i}{max(X)}\right)^2 \tag{8}$$

İ. *Zavadskas Turskis Log Standardization*

This method standardises the values in a dataset using a logarithmic transformation. The transformation is a data preprocessing method that corrects the distribution of data by compressing values into a generally more restricted range and making them more robust to outliers [2].

$$x_i = \frac{\ln x_i}{\ln \prod_{j=1}^{n} x_j} \tag{9}$$

J. *Altman Z-score Standardization*

Altman standardization method uses the root mean square deviation as denominator to handle skewness in the data and approximate the data to a normal distribution [3].

$$x_i = \frac{x_i - \text{mean}(X)}{\sqrt{\frac{1}{n-1}\sum_{j=1}^{n}(x_i - \text{mean}(X))^2}} \tag{10}$$

In general, X shows the data vector and x_i shows the instance i of the series X. Functions given in the formulas such as max, min, and std are references for maximum, minimum, median and standard deviation functions respectively.

III. EXPERIMENTS

A. *Dataset*

Since the continuous variables does not exist most of the benchmarking ranking datasets, we only considered Criteo dataset. We use % 5of the Criteo dataset, about 2.2 million rows of data, due to computational time concerns of the experiments. To sample the Criteo dataset we apply a proper approach that ensures keeping the density distribution of the sample data same as the original data. We split %80 of this %5 part of data as training set. Remaining %20 of the data are split into two equal parts as the validation set and test set. Thus, we ensure that we use same splits of the data for each model training and evaluation. Additionally, in order to have reproducible, consistent, and comparable experimental results, we also set the random seed to 2023, which is the year that this work is submitted.

B. *Parameters*

For each experiment run we tune hyper-parameters on validation set and use same hyper-parameters to evaluate models on test set. In order to perform hyper-parameter tuning, we apply a manual binary search; learning rate is selected from the set [$1e^{-3}$, $1e^{-4}$, $5e^{-5}$, $1e^{-5}$], while the batch size is initially set as 1000 and is increased gradually taking values from the set [1000, 5000, 10000, 20000].

C. *Training*

We considered 3 deep ranking models which are DCN [4], DeepFM [5] and FiBiNET [6]. The model implementations obtained from BARS benchmark [7]. In order to prevent overfitting, an early stopping criterion is considered, which stops the iteration when the validation loss drops two consecutive epochs. Additionally, we use the Adam optimizer [8] with a weight decay rate of 1e-6 that is chosen from the set of [1e-4, 1e-5, 1e-6].

IV. RESULTS AND DISCUSSION

Table I, II and III shows DCN, DeepFM and FiBiNET results respectively. All scaling and standardization methods listed with corresponding F1, AUC and log loss scores obtained from Criteo dataset. The reason behind multiple model usage is to ensure the performances obtained from standardization methods.

TABLE I

DCN RESULTS WITH F1, AUC AND LOG LOSS METRICS ON CRITEO DATASET

Standardization Method	**DCN**		
	F1	**AUC**	**Log-loss**
AltmanZscoreStandardization	**48.0**	**77.6**	47.7
ManhattanStandardization	40.5	74.4	49.4
MaxAbsScaler	42.8	76.0	48.4
MinMaxScaler	43.0	76.1	48.4
PeldschusNonlinearStandardization	41.1	75.3	48.9
RobustScaler	42.3	75.5	53.6
StandardDeviationStandardization	44.0	77.4	47.6
ZScoreStandardization	45.9	77.5	**47.5**
VectorStandardization	40.2	74.7	49.3
ZavadskasTurskisLogStandardization	40.2	74.7	49.3

TABLE II

DEEPFM RESULTS WITH F1, AUC AND LOG LOSS METRICS ON CRITEO DATASET

Standardization Method	**DCN**		
	F1	**AUC**	**Log-loss**
AltmanZscoreStandardization	**48.0**	**77.8**	47.4
ManhattanStandardization	40.2	74.6	49.3
MaxAbsScaler	43.5	75.9	48.6
MinMaxScaler	43.0	76.0	48.4
PeldschusNonlinearStandardization	40.7	75.0	49.1
RobustScaler	43.6	77.4	47.3
StandardDeviationStandardization	44.2	77.5	47.3
ZScoreStandardization	45.9	77.6	**47.2**
VectorStandardization	39.8	74.5	49.4
ZavadskasTurskisLogStandardization	39.8	74.5	49.4

TABLE III

FIBINET RESULTS WITH F1, AUC AND LOG LOSS METRICS ON CRITEO DATASET

Standardization Method	**DCN**		
	F1	**AUC**	**Log-loss**
AltmanZscoreStandardization	**48.1**	**77.6**	47.7
ManhattanStandardization	39.8	74.4	49.6
MaxAbsScaler	43.4	76.1	48.5
MinMaxScaler	43.3	76.0	48.6
PeldschusNonlinearStandardization	41.3	75.2	49.1
RobustScaler	44.8	77.4	**47.3**
StandardDeviationStandardization	43.7	77.4	47.4
ZScoreStandardization	45.7	77.3	47.4
VectorStandardization	40.6	74.4	49.6
ZavadskasTurskisLogStandardization	40.7	74.4	49.7

We observe that in each model, Altman Z-score Standardization method performs better in F1 and AUC metrics. Since we initialize weights with normal distribution, we expect a normally distributed input to yield better results. Consequently, we can assume that the Altman Z-score standardization approximate the data distribution to normal distribution closer than any other standardization and scaling method. Additionally, z-score standardization performs better in the manner of log loss metric. We conclude that methods with standard deviation as denominator performs better with respect to other methods. Min-Max and Max-Abs scaling methods perform slightly worse than methods that use standard deviation. Robust scaler also performs well especially in FiBiNET and DeepFM models, shows that these two models are more sensitive to the outliers with respect to DCN model.

V. CONCLUSION

We aim to analyse the performance benefits of standardization and scaling methods in deep neural network models in learning to rank domain. In order to ensure the performance measures of the methods, we considered three popular models and evaluated the results in f1, AUC and log loss metrics since class imbalance exists in the dataset. Generally, standardization algorithms performs better than scaling methods while methods which are robust to outliers and skewness performs close to the standardization methods. This behaviour shows sensitivity of neural networks to the outliers and skewness in the dataset.

We consider this study as preliminary for future studies on the performances of standardization and scaling methods in the manner of training time and the number of approximation iterations. We also plan to conduct hyper-parameter search with bayesian methods to ensure models yield best performances.

REFERENCES

[1] W. Gardziejczyk and P. Zabicki, "Normalization and variant assessment methods in selection of road alignment variants–case study," *Journal of civil engineering and management*, vol. 23, no. 4, pp. 510–523, 2017.

[2] E. K. Zavadskas and Z. Turskis, "A new logarithmic normalization method in games theory," *Informatica*, vol. 19, no. 2, pp. 303–314, 2008.

[3] E. I. Altman, M. Iwanicz-Drozdowska, E. K. Laitinen, and A. Suvas, "Financial distress prediction in an international context: A review and empirical analysis of altman's z-score model," *Journal of International Financial Management & Accounting*, vol. 28, no. 2, pp. 131–171, 2017.

[4] R. Wang, B. Fu, G. Fu, and M. Wang, "Deep & cross network for ad click predictions," in *Proceedings of the ADKDD'17*, 2017, pp. 1–7.

[5] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "Deepfm: a factorization-machine based neural network for ctr prediction," *arXiv preprint arXiv:1703.04247*, 2017.

[6] T. Huang, Z. Zhang, and J. Zhang, "Fibinet: combining feature importance and bilinear feature interaction for click-through rate prediction," in *Proceedings of the 13th ACM Conference on Recommender Systems*, 2019, pp. 169–177.

[7] J. Zhu, Q. Dai, L. Su, R. Ma, J. Liu, G. Cai, X. Xiao, and R. Zhang, "BARS: towards open benchmarking for recommender systems," in *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, E. Amigó, P. Castells, J. Gonzalo, B. Carterette, J. S. Culpepper, and G. Kazai, Eds. ACM, 2022, pp. 2912–2923. [Online]. Available: https://doi.org/10.1145/3477495.3531723

[8] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.