KeAi
CHINESE ROOTS
GLOBAL IMPACT

International Journal of Innovation Studies

DEEP LEARNING APPROACH FOR PREDICTING AD CLICK

Mrs. J. Thamaraiindhu [a]*

[a] Post Graduate Student, Department of Computer Science and Engineering, School of Engineering and Technology, Jeppiaar University, Chennai, Tamilnadu, India

Mrs. S. Kayalvizhi [b]

[b] Assistant Professor, Department of Computer Science and Engineering, School of Engineering and Technology, Jeppiaar University, Chennai, Tamilnadu, India

Mrs. T. Subha Rathi Priya [c]

[c] Assistant Professor, Department of Computer Science andEngineering,School of Engineering and Technology, Jeppiaar University, Chennai, Tamilnadu, India

*Corresponding Author: Mrs. J. Thamaraiindhu
Post Graduate Student, Department of Computer Science andEngineering,School of Engineering and Technology, Jeppiaar University, Chennai, Tamilnadu, India Email: thamaraiindhu1997@gmail.com

Abstract

In the digital advertising landscape, predicting user behaviour and user interest impact in response and action based on the advertisement. This study used a deep learning approach for predicting ad clicks, leveraging a comprehensive dataset that includes user demographics, ad attributes, and contextual information. To date, existing methods for Ad click prediction, or click-through rate prediction, mainly consider representing users as a static feature set and train machine learning classifiers to predict clicks. In this paper we propose a deep learning approach for predicting ad click by BI-LSTM method and evaluate the model performance metrics such as precision, recall and f1-score.our goal is to accurately predict the past and future behaviour of the user based on the ads. To achieve the goal, we collect page information displayed to the users as a temporal sequence and use bi direction long short-term memory (BI-LSTM) network to learn features of both forward and backward direction that represents user interests as latent features. on real-world data show that, compared to existing approaches, considering bidirectional long and short-term sequences, user requests results in improvements in user Ad response prediction and campaign specific user Ad click prediction.

Keywords— Deep Learning, Bi-lstm network, Ad click prediction, User behaviour, forward and backward direction.

1.INTRODUCTION

1.1 Definition

Ad click prediction, also known as click-through rate (CTR) prediction, is theprocess of predicting whether a user will click on an ad. Ad click prediction refers to the use of algorithms and statistical models to forecast the likelihoodthat a user will click on a specific online advertisement for user interest and response by deep learning. This involves analysing various factors such as user behaviour, demographics, ad characteristics, and contextual information. The goal is to optimize ad placement and targeting to improve click-through rates and ultimately enhance the effectiveness of digital advertising campaigns.

1.2 Text Processing

Text processing is the process of automatically analysing, manipulating, or generating electronic text. It involves using computer commands to perform tasks like searching and replacing, formatting, and filtering. Unstructured text data can be automatically analysed and sorted by text processing to obtain useful information. Text processing systems can automatically comprehend human language and derive value from text data usingNatural Language Processing (NLP) and deep learning and communicate with words, not numbers, companies receive much raw text data through email, chat conversations, social media. Text processing methods is used for evaluating and sorting textual data now that we are more familiar with text processing. The methods used to process and analysetext from a frequency distribution, collocation, concordance, and TF-IDF. Text classification takes a text dataset and it is used to extract valuable data from customer reviews and customer service logs.

1.3 Deep Learning

Deep learning is a subset of machine learning that uses neural networks with many layers (hence "deep") to model complex patterns in data. It performs tasks in natural language processing. Neural Networks are composed of interconnected nodes (neurons) organized in layers: an input layer, one or more hidden layers, and an output layer. Deep learning models are trained on large datasets, adjustingthe weights of connections between neurons to minimize the difference between predicted and actual outputs. Backpropagation is the algorithm used to update weights during training, allowing the model to learn from its errors. Activation

Functions introduce non-linearity into the model, enabling it to learn complex patterns. Deep learning has gained popularity due to its ability tohandle large amounts of data and its success in achieving state-of-the-art results in many AI applications. DNNs are neural networks that consist of multiple hidden layers. Abstract features the network can learn, enabling it to make predictions with greater accuracy.

1.4 Deep Learning For Text Processing

word Embeddings like Word2Vec, GloVe, FastText convert words into dense vector representations that capture semantic meanings. Similar words are closer together in the vector space, enabling better understanding and context. Recurrent Neural Networks (RNNs) are designed for sequential data, making for text processing tasks. They can remember previous inputs, for understanding context in sentences. Transformers in text processing used to understand the relationshipsbetween words in a sentence. They excel in a variety of tasks, from text to Natural Language Processing (NLP) Tasks**.** Text classification assigning categories to text, used in sentiment analysis or spam detection. Named Entity Recognition (NER) Identifying entities (people, organizations, locations) in text. Machine Translation translating text from one language to another. Text generation creating coherent and contextually relevant text, as seen in chatbots and content creation tools. Attention mechanisms allow on specific parts of the input text when making predictions, enhancing understanding and performance on tasks involving context. pre-trained models that can be easily adapted for various NLP tasks, making deep learning more accessible. While CNNs are used to text processing, especially in tasks such as sentence classification or sentiment analysis and LSTM handles long-range dependencies using memory gates for user interest and probability of user click.

1.5 Ad Click Prediction

Ad click prediction, also known as click-through rate (CTR) prediction, is the process of predicting whether a user will click on an ad. The ad clicks advertisement process using deep learning involves several stages, to improve the prediction of click-through rates (CTR) and predicting online advertisement extracting user interest and modelling evolution from user historical behaviour to overfit a method used factorization (FM), Deep (FM) and Gated Recurrent Unit Neural Network (GRU). Analyse factors such as userbehaviour, demographics, ad characteristics, and contextual information. The goal is to optimize ad placement and targeting to improve click-through rates and ultimately enhance the effectiveness of digital advertising campaigns, the interest based deep model trace dynamic of interest for positive and negative samples by Deep based dynamic Interest Perception Network (DIPN). In this paper deep learning techniques are often employed to refine these predictions based on historical data. Ad click prediction using deep learning involves employing neural networks to estimate the probability that a user will click on an advertisement based on various input features. The click-through rate is the ratio of the number of clicks to the number of ad impressions. Predicting online shopping behaviour and target marketing in real time a RNN compares to SML benchmarks and in this performance are standard classifiers for RNN clickstream modelling. Log loss measures the accuracy of probabilistic predictions. A lower log loss indicates better performance. For precision the proportion of true positives (correct clicks predicted) out of all predicted positive cases (all predicted clicks). In the existing system deep user segment interest, a Tao Bao data is real data in advertising platform.

2. LITERATURE SURVEY

2.1 Deep Learning for User Interest and Response Prediction in Online Advertising.

Two deep learning-based frameworks, LSTMcp and LSTMip, for user click prediction and user interest modelling. Our goal is to accurately predict (1) the probability of a user clickingon an Ad and (2) the probability of a user clicking a specific type of Ad campaign. To achieve the goal, we collect page information displayed to the users as a temporal sequence and use long short-term memory (LSTM) network to learn features that represents user interests as latent features (Gharib shah, X Zhu, A Hainline, M Conway - Data Science and …,2020).

2.2Deep Learning for Click Through Estimation.

CTR estimation performance and now deep CTR models have been widely applied in many industrial platforms. In this survey, we provide a comprehensivereview of deep learning models for CTR estimation tasks. First, we take a reviewof the transfer from shallow to deep CTR models and explain why going deep isa necessary trend of development. Second, we concentrate on explicit feature interaction learning modules of deep CTR models. Then, as an important perspective on large platforms with abundant user histories, deep behaviour models are discussed (W Zhang, J Qin, W Guo, R Tang,2021).

2.3AHybrid Neural Network Architecture to Predict Online Advertising Click rate behaviour.

A new hybrid neural network, DGRU, which integrates Factorization-Machine Based Neural Network (Deep FM) and Gated Recurrent Unit Neural Network (GRU). The Deep FM component is responsible for performing the autonomic feature combination, and the GRU component is designed to model user interests and evolutions. The GRU component is fed with a series of 1 and 0 representing user click behaviours. It contains information on what users like and dislike. Moreover, the conciseness of the format is helpful to avoid the problem of overfitting (R Zhou, C Liu, J Wan, Q Fan, Y Ren,J Zhang, N Xiong,2021).

2.4 CTR Prediction models Considering the Dynamics of user interest.

To Click-through rate (CTR) prediction is one of the key areas in industrial bidding advertising. To improve prediction performance the interest-based deep models that learn the user's latent interest fromhistorical click behaviours. the interest-based deep models ability to trace the dynamics of interest from both the positive and negative samples, which leading to prediction errors. DIPN model introduces three new parts to the interest-based deep model: a gated recurrent unit (GRU); an attention mechanism; an interest degree feature (H Zhang, J Yan, Y Zhang IEEE Access, 2020).

2.5Predicting Online Shopping Behaviour from Clickstream data using Deep learning.

Click stream data is an important source to enhance user experience and pursue business objectives in e-commerce. The paper uses clickstream data to predict online shopping behaviour and target marketing interventions in real-time. a methodology capable of unlocking the full potential of clickstream data using the framework of Recurrent Neural Network (RNNs). To this end, the paper proposes an approach to measure the revenue impact of a targeting model (D Koehn, S Lessmann, M Schaal,2020).

2.6 Deep user segment interest network modelling for click-through rate prediction of online advertising

Online advertisement (Ads) can show great marketing ability by processing data from multiple channels to convey information, understanding what users want, and approaching them easily. Moreover, predicting the click- through rate (CTR) can increase advertisement revenue and user satisfaction. This can be alleviated through the segmentation of users with similar interests a novel model, the Deep User Segment Interest Network, to improve CTR prediction. three novel layers for improving performance: i) an individual interestextractor, ii) a segment interest extractor, and iii) a segment interest activation (K Kim, E Kwon, J Park IEEE Access, 2021).

3.RESEARCH GAP ANALYSIS

A research gap analysis for deep learning approaches for ad click prediction in text processing involves identifying the key areas where existing research falls short or where further exploration could significantly advance the field. Lack of methods for automatic feature extraction and then representation learning from text data in the context of ad click prediction.

More research is needed to design context-aware deep learning models that can account for how text should be dynamically interpreted based on a user's context. There's potential for research into online learning algorithms or real-time deep learning models for ad click prediction that can continuously adapt based on new textual data (like recent user queries or newly launched ads).

4.PROPOSED METHODOLOGY

Ad click prediction using a Bidirectional Long Short-Term Memory (Bi-LSTM) model is an effective approach for analysing sequences of user interactions or adexposure over time. Here's a high-level overview of how to implement this by Data Preparation, Data Preprocessing, Model Architecture, Training the Model, Evaluation and Deployment.

In first step Data Preparation: Collect a dataset that includes features such as userID, ad ID, timestamp, ad characteristics (e.g., type, category), user demographics,and whether the ad was clicked (target variable). Feature Engineering will Createrelevant features such as, User's historical click-through rate (CTR), Time-based features (e.g., hour of day, day of week), Encoding categorical variables (e.g., one-hot encoding). Sequence Generation: Construct sequences for each user, where each sequence contains a series of past interactions with ads (e.g., timestamps, ad IDs) leading up to a potential click. In second step Data Preprocessing by Normalization: Normalize continuous features to help the model converge, Padding: Ensure all sequences are of the same length by padding shorter sequences with zeros, Train-Test Split: Split the dataset into training, validation, and test sets. In third step Model Architecture done Input Layer: Accept sequences of features, Bi-LSTM Layer: Use a Bi-LSTM layer to capturecontext from both past and future interactions. This helps the model understand patterns in sequences, Dense Layer: Follow the Bi-LSTM with one or more dense layers for further processing, Output Layer: Use a sigmoid activation function forbinary classification (clicked or not clicked).In forth step Training the Model Loss Function: Use binary cross-entropy as theloss function. Optimize**r**: Choose an optimizer (e.g., Adam) and define the learning rate. Training: Fit the model on the training data, using the validation setto tune hyperparameters. After training Evaluation done by Metrics: Evaluate themodel using metrics like accuracy, precision, recall, and F1-score.AUC-ROC: Analyse the receiver operating characteristic curve and area under the curve to assess performance. Finally in Deployment Real-time Prediction: Once trained, the model can be deployed to predict ad clicks in real-time. Monitoring: Continuously monitor model performance and retrain with new data as necessary. Advantages are Experiment with hyperparameters like the number of LSTM units, batch size, and dropout rates to improve performance. Use techniques likecross-validation to validate the model's effectiveness. Consider more advanced features like attention mechanisms for enhanced sequence processing. This should give you a solid starting point for implementing ad click prediction using Bi-LSTM! Bi-LSTM models can adapt to changing user interests over time, as they are designed to learn patterns in sequential data. This adaptability is crucial for online advertising platforms where user preferences can shift quickly based on factors such as seasonality, trends, and personal preferences.

Forward pass in a Bidirectional Long Short-Term Memory (BiLSTM) network involves in forward direction (typically left-to-right in text), the BiLSTM uses the standard LSTM equation with slight modifications. For each time step t, the LSTM in the forward direction updates the hidden state $h_t^{forward}$ and cell state $c_t^{forward}$ using the following equations.

1.Forget Gate: $f_t^{forward}=\sigma\ (W_f\cdot\ [h_{t-1}^{forward},\ x_t]\ +b_f)$. Where, $h_t-1^{forward}$ is the hidden state from the previous time step. X_t is the input at the current time step, W_f and b_f are the weight matrix and bias term for the forget gate.

2. Input Gate: $i_t^{forward}=\sigma\ (W_i\cdot\ [h_{t-1}^{forward},\ x_t]\ +b_i)$ where, σ is the sigmoid activation function.

3. Candidate Cell State: $C_t^{forward}=\tanh\ (W_C\cdot\ [h_{t-1}^{forward},\ xt]\ +b_C)$ where, tanh is the hyperbolic tangent activation function.

4. Update Cell State: $C_t^{forward}=f_t^{forward}\cdot C_{t-1}^{forward}+i_t^{forward}\cdot C_t^{forward.}$combining the previous cell state,

weighted by the forget gate, and the candidate cell state, weighted by the input gate.

5. Output Gate: $o_t^{forward}=\sigma\ (W_o\cdot\ [h_{t-1}^{forward},\ x_t]\ +b_o)$.

6. Update Hidden State: $h_t^{forward}=o_t^{forward}\cdot\tanh\ (C_t^{forward})$.

Final Output of the Forward LSTM at Time Step t. The output at each time step t is $h_t^{forward}$ which represents the hidden state of the forward LSTM at that point in time as represented in figure 1 and followed by the equations from 1 to 6.

Backward pass in a Bidirectional Long Short-Term Memory (BiLSTM) network involves in backward direction (typically right-to-left in text), operates similarly to the forward pass but in reverse direction. For each time step t, the LSTM in the backward direction updates the hidden state $h_t^{backward}$ and cell state $c_t^{backward}$ using the following equations.

7. Forget Gate: $f_t^{backward} = \sigma\ (W_f^{backward} \cdot [h_{t+1}^{backward}, xt] + b_f^{backward})$ where, $h_{t+1}^{backward}$ is the hidden state from the next time step (we're going backwards) x_t is the input at the current time step, $W_f^{backward}$ and $b_f^{backward}$ are the weight matrix and bias term for the forget gate in the backward direction.

8. Input Gate: $i_t^{backward} = \sigma\ (W_i^{backward} \cdot [h_{t+}{}^{1backward}, x_t] + b_i^{backward})$

9. Candidate Cell State: $C^{\sim tbackward} = \tanh\ (W_C^{backward}. [h_{t+1}^{backward,}\ xt_{]} + b_C^{backward})$

10.Update Cell State: $C_t^{backward} = f_t^{backward} \cdot C_{t+1}^{backward} + i_t^{backward} \cdot C^{\sim}{}_t^{backward} \cdot$ combining the next cell state, weighted by the forget gate, and the candidate cell state, weighted by the input gate.

11. Output Gate: $o_t^{backward} = \sigma\ (W_o^{backward} \cdot [h_{t+1}^{backward,}\ x_{t]} + b_o^{backward})$

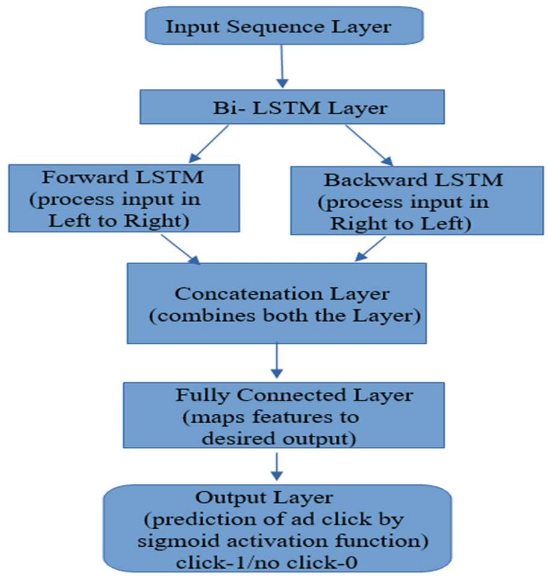
12. Update Hidden State: $h_t^{backward} = o_t^{backward} \cdot \tanh\ (C_t^{backward})$

The output for the backward LSTM at time step t is h_t backward, which represents the hidden state of the backward LSTM at that point in time followed by the equations 6 to 12.

The final output at each time step t for the BiLSTM $h_t^{BiLSTM} = [h_t^{forward}, h_t^{backward}]$.

This structure allows the BiLSTM to capture both past and future information by using both the forward and backward directions.

Figure 1: Represents the flowchart of proposed system.



4.1PROPOSED ARCHITECTURE

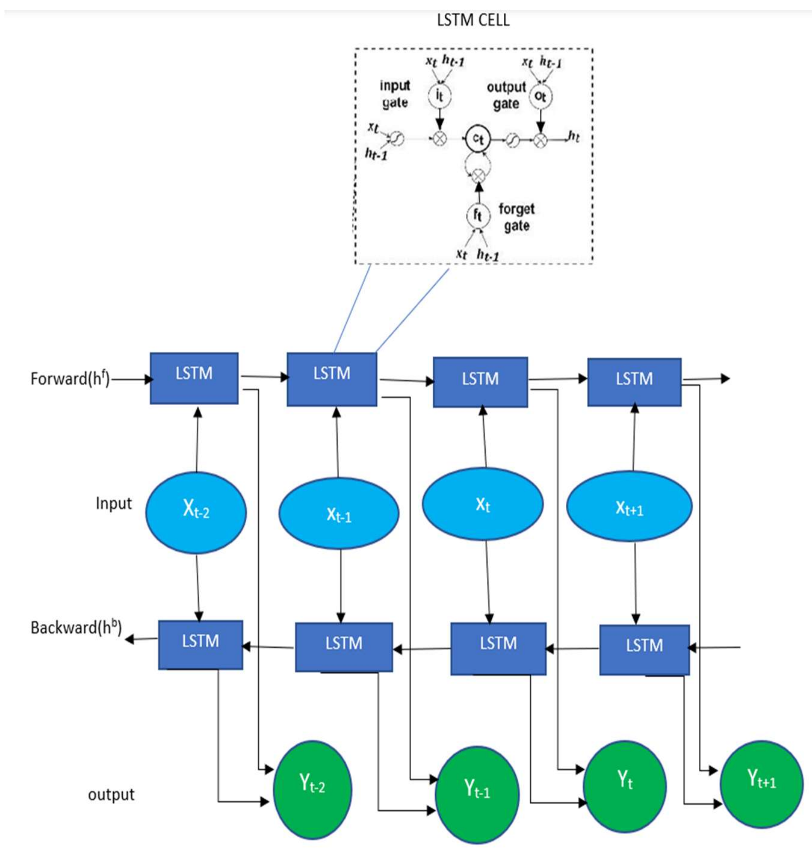


Figure 2: Shows the Architecture of the proposed system

5. IMPLEMENTATION

Step by step process to build a model for Ad click dataset by using Bi LSTM.

5.1 Data Collection and Data Description

The data is collected from the Kaggle website.
https://www.kaggle.com/code/marius2303/ad-click-prediction-dataset-data-analysis/notebook

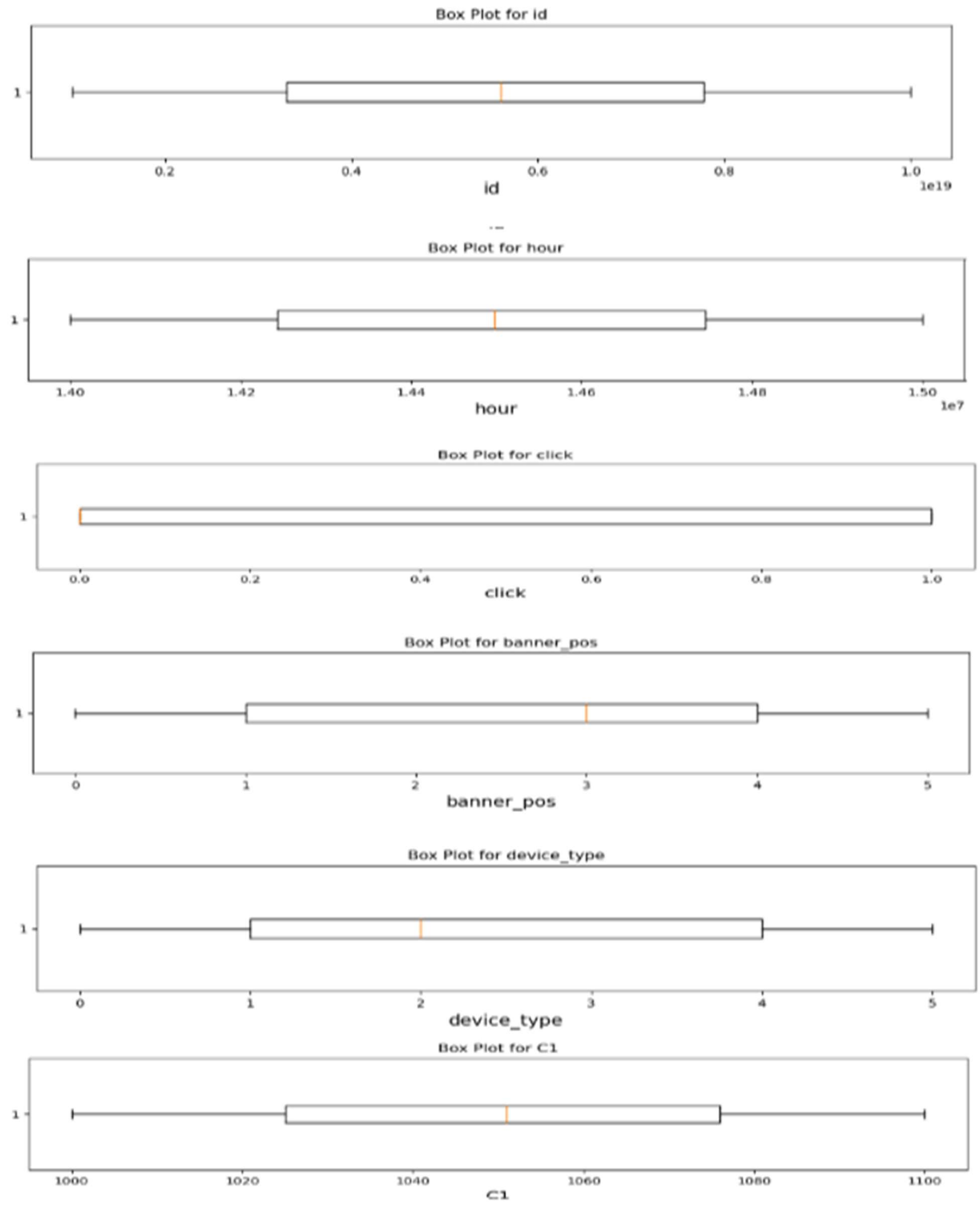
Data description number of rows and columns in data frame as shows in table 1.

Table 1: Shows the description values in the data frame.

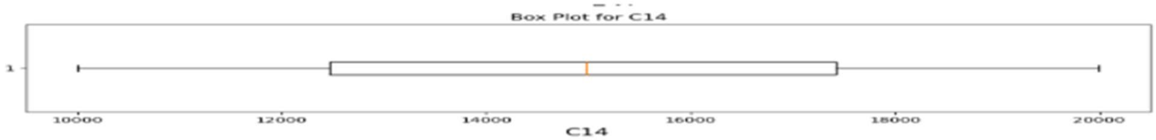
	id	click	hour	C1	banner_pos	device_type	device_conn_type
count	1.000000e+03	1000.000000	1.000000e+03	1000.000000	1000.000000	1000.000000	1000.000000
mean	5.513154e+18	0.533000	1.451437e+07	1048.052000	2.537000	2.537000	2.449000
std	2.605000e+18	0.499159	2.872798e+05	28.894501	1.704563	1.687447	1.687082
min	1.006930e+18	0.000000	1.400020e+07	1000.000000	0.000000	0.000000	0.000000
25%	3.234060e+18	0.000000	1.427409e+07	1022.750000	1.000000	1.000000	1.000000
50%	5.500480e+18	1.000000	1.451148e+07	1047.000000	2.500000	3.000000	2.000000
75%	7.948051e+18	1.000000	1.476755e+07	1073.000000	4.000000	4.000000	4.000000
max	9.985903e+18	1.000000	1.499978e+07	1100.000000	5.000000	5.000000	5.000000

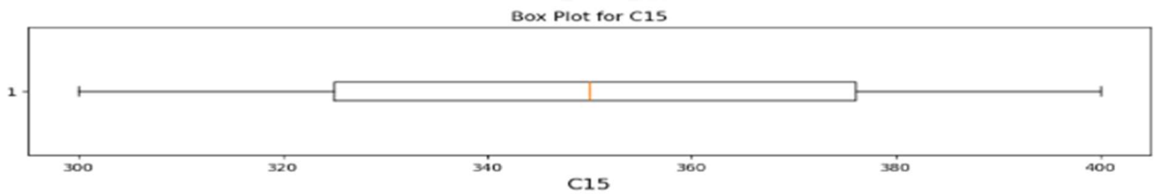
	C14	C15	C16	C17	C18	C19	C20	C21
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000
mean	14895.606000	348.812000	45.059000	2000.708000	2.951000	1103.423000	0.334000	39.961000
std	2906.995528	29.251877	9.054032	56.940064	1.370576	58.399851	3.151885	6.078357
min	10001.000000	300.000000	30.000000	1900.000000	1.000000	1000.000000	-5.000000	30.000000
25%	12325.000000	323.000000	37.000000	1952.000000	2.000000	1053.000000	-2.000000	35.000000
50%	14729.500000	347.000000	45.000000	2002.000000	3.000000	1105.000000	1.000000	40.000000
75%	17414.500000	374.250000	53.000000	2050.000000	4.000000	1155.250000	3.000000	45.000000
max	19995.000000	400.000000	60.000000	2100.000000	5.000000	1200.000000	5.000000	50.000000

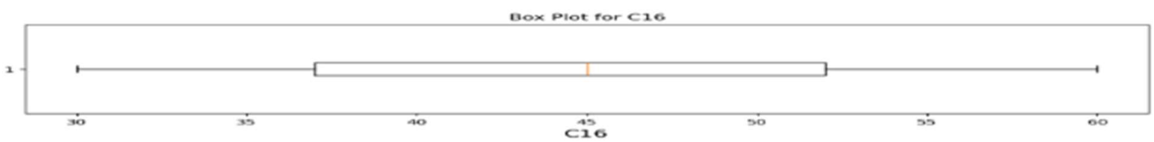
Box plot displays the range within the variables measured between the Minimum quartile, median and maximum quartile as shown in fig 3.



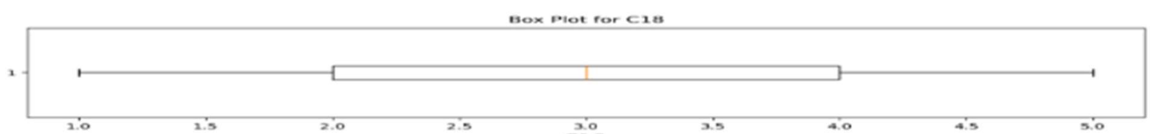

Box Plot for device_conn_type
1
0
1
2
3
4
5
device_conn_type


Box Plot for C14
1
10000
12000
14000
16000
18000
20000
C14

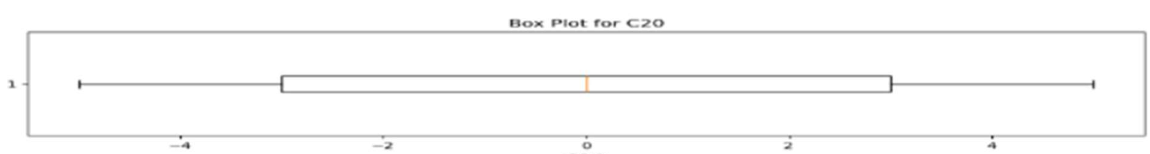

Box Plot for C15
1
300
320
340
360
380
400
C15

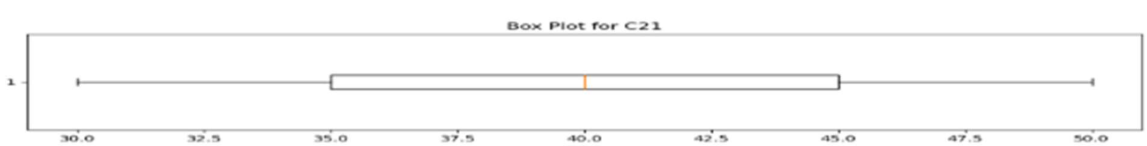

Box Plot for C16
1
30
35
40
45
50
55
60
C16


Box Plot for C17
1
1900
1925
1950
1975
2000
2025
2050
2075
2100
C17


Box Plot for C18
1
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
C18


Box Plot for C19
1
1000
1025
1050
1075
1100
1125
1150
1175
1200
C19


Box Plot for C20
1
−4
−2
0
2
4
C20


Box Plot for C21
1
30.0
32.5
35.0
37.5
40.0
42.5
45.0
47.5
50.0
C21

5.2 Data Generation

Since deep learning model requires more data have been generated using random package with the same columns and rows likewise 1000 rows are generated. After generation our dataset was extended to 5000 rows and 24 columns.

5.3 Data Preparation

In data preparation the following steps were done.

Step1: Load the data (replace with your file path or data frame)

Step2: Handle missing values

We can fill numerical columns with the mean and categorical ones with most frequent value.

Step 3: Preprocess the data by combining the required values.

5.4 Model Building

A Bi LSTM model was build with three layers namely bidirectional, dropout and dense.

K-fold validation 10 folds was done with 10% as the test data size.

5.5 Model Training

The model was trained for 10 epochs with 64 as batch size the validation was done.

5.6 Model validation

The model was evaluated using the accuracy from epochs.

Table 2: Represents the comparison of test accuracy for epochs 10 and 20.

Epochs	Test Accuracy
10	0.5100
20	0.960

From the epochs 20 our model performed better accuracy as 0.960 as shown in table 2 and fig 3.

The model accuracy and model loss for the epochs 10 and epochs 20 for the test data and train data as shows in fig 3 and fig 4.

Figure 3: Shows the graph for accuracy and loss for epochs 10.

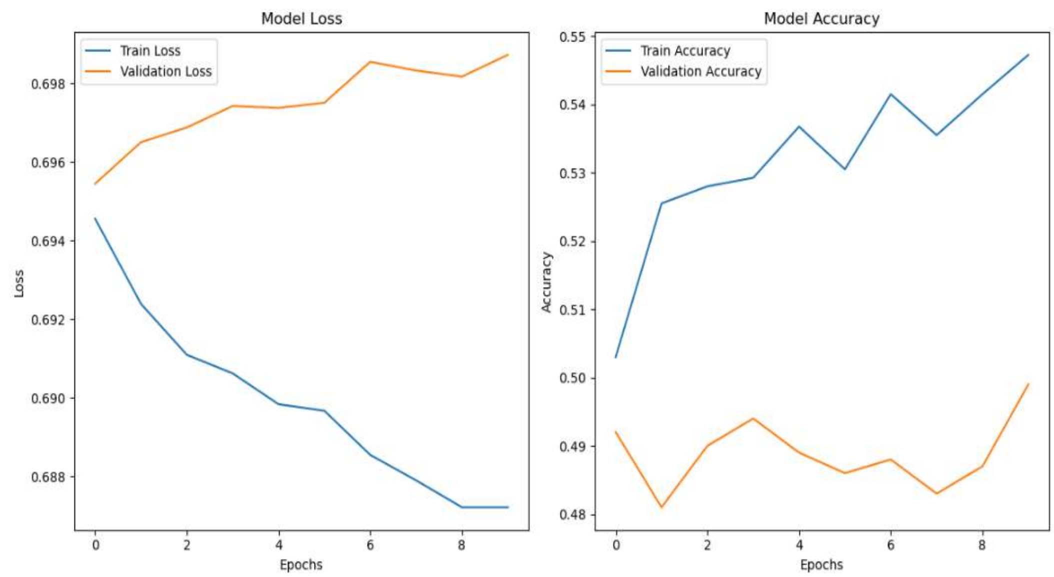
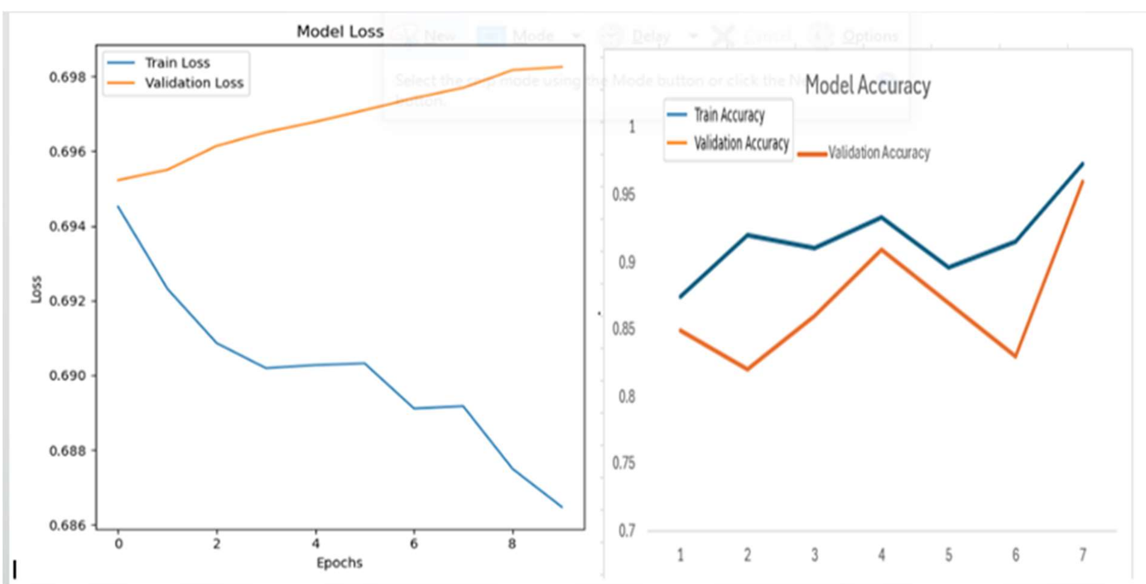


Figure 4: Shows the graph for accuracy and loss for epochs 20.



After the process of test and train data the model was evaluated using the accuracy from 10 cross validation.

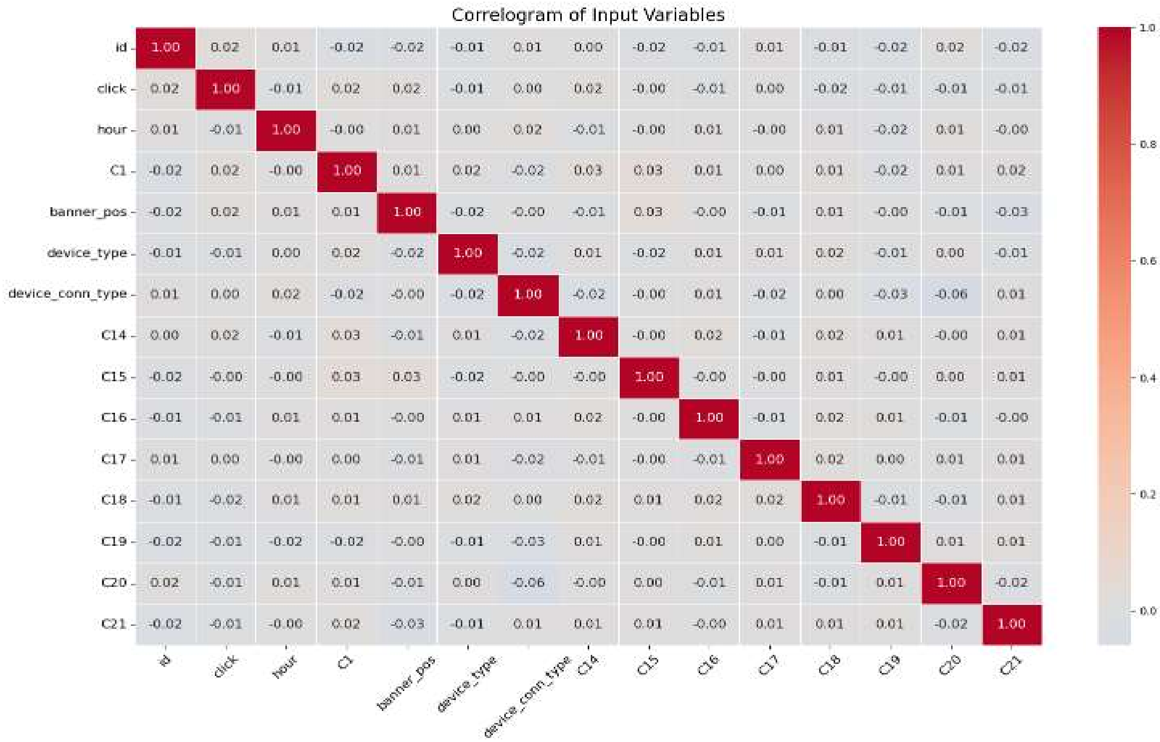


Figure 5: Shows the correlogram of input variables for every feature

Table 3: Represents the comparison of test accuracy for fold 5 and fold 10.

10-cross validation	Accuracy
Fold 5	0.5950
Fold 10	0.9623

By comparing the fold 5 and fold 10 as shown in table 3 our model performed better accuracy as 0.9623 as shown in fig 4. The correlogram of input variables for every feature is represented by the correlation matrix of positive values and negative values which are shown in fig 5.

6. RESULT

Our model performed well with accuracy of 0.96. The values of other metrics is shown in table 4 and table 5.

Table 4: Shows the values of actual and predicate for click and not click in confusion matrix.

TRUE POSITIVE 146	FALSE NEGATIVE 165
FALSE POSITIVE 545	TRUE NEGATIVE 300

In our model the true positive values that are predicted for the actual value which are click as 146 and

true negative values that are predicted for the actual values as not click as 300.

The other evaluation metrics are Accuracy, Precision, Recall, F1-Score.

Table 5: shows the values of other evaluation metrics.

Accuracy	0.96
Precision	0.96
Recall	0.98
F1-Score	0.97

our proposed system well performed with the better performance of accuracy, precision, recall

and f1-score when compared to the other existing system as shown in table 6.

Table 6: shows the comparison of existing system to our proposed system.

MODEL	ACCURACY	PRECISION	RECALL	F1-S
MLP	0.714	0.75	0.75	0
KNN	0.95	0.89	0.86	0
CNL	0.69	0.70	0.65	0
Naives Bayes	0.8708	0.2969	0.2730	0.2
Random Forest	0.8709	0.2991	0.2770	0.2
SVM	0.8949	0.3938	0.2176	0.2
$LSTM_{CP}$	0.8481	0.3140	0.5183	0.3
Proposed System BI-LSTM	**0.962**	**0.96**	**0.988**	**0**

7. CONCLUSION

In this study utilizing deep learning approaches for predicting ad clicks has shown significant promise in enhancing advertising effectiveness and optimizing resource allocation. By leveraging large datasets and complex models, deep learning can capture intricate patterns and user behaviours that traditional methods might overlook. This leads to more accurate predictions, improving targeting strategies and increasing conversion rates.

In conclusion we explore Deep learning approach for predicting ad clicks demonstrates potential to enhance the effectiveness of digital advertising. And application of Bidirectional Long Short-Term Memory (Bi-LSTM) networks forpredicting ad clicks, leveraging their ability to capture sequential patterns in user behaviour. By analysing user interactions over time, Bi-LSTM models effectively incorporate both past and future context. These models improve prediction accuracy and enable more personalized ad targeting to higher click-through rates and better return on investment. The proposed Bi-LSTM-based system aims to enhance CTR prediction by effectively modelling the dynamic, time-varying nature of user interest. By leveraging both past and future user behaviour, the Bi-LSTM model improves the accuracy of click predictions and adapts to evolving user preferences, providing more personalized and relevant ad recommendations. Our proposed system outperformed the existing system with an Accuracy of 0.962, Precision of 0.96, Recall of 0.988 and F1-Score of 0.97.

REFERENCES

1. Gharibshah, Z., Zhu, X., Hainline, A., & Conway, M. (2020). Deep learning for user interest and response prediction in online display advertising. *Data Science and Engineering,* 5(1), 12–

26. https://doi.org/10.1007/s41019-019-00115-y

2. Zhang, W., Qin, J., Guo, W., Tang, R., & He, X. (2021, April 21). *Deep Learning for Click-Through rate estimation.* arXiv.org. https://arxiv.org/abs/2104.10584
3. Zhou, R., Liu, C., Wan, J., Fan, Q., Ren, Y., Zhang, J., & Xiong, N. (2021). A hybrid neural network architecture to predict online advertising Click-Through rate behaviors in social networks. *IEEE Transactions on Network Science and Engineering,* 8(4), 3061–3072. https://doi.org/10.1109/tnse.2021.3102582
4. H Zhang, J Yan &Y Zhang (2020). CTR prediction models considering the dynamics of user interest. *IEEE Journals & Magazine | IEEE Xplore.* https://ieeexplore.ieee.org/abstract/document/9068250
5. Koehn, D., Lessmann, S., & Schaal, M. (2020b). Predicting online shopping behaviour from clickstream data using deep learning. *Expert Systems With Applications,* 150, 113342. https://doi.org/10.1016/j.eswa.2020.113342
6. Kim, K., Kwon, E., & Park, J. (2021). Deep user segment Interest Network Modeling for Click-Through Rate Prediction of Online advertising. *IEEE Access,* 9, 9812–9821. https://doi.org/10.1109/access.2021.3049827
7. Sangsawang, T. (2024). Predicting Ad Click-Through Rates in Digital Marketing with Support Vector Machines. *Journal of Digital Market and Digital Currency.,* 1(3), 225–246. https://doi.org/10.47738/jdmdc.v1i3.20
8. Bratus, O., & Bidyuk, P. (2024). TOWARDS CLICK-THROUGH RATE PREDICTION IN ONLINE ADVERTISING. *Problems of Applied Mathematics and Mathematic Modeling,* 3–17. https://doi.org/10.15421/322301
9. Zhang, Y., Dai, H., Xu, C., Feng, J., Wang, T., Bian, J., Wang, B., & Liu, T. (2014). Sequential Click Prediction for Sponsored Search with Recurrent Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence,* 28(1). https://doi.org/10.1609/aaai.v28i1.8917
10. Chen, J., Sun, B., Li, H., Lu, H., & Hua, X. (2016). Deep CTR Prediction in Display Advertising. *Proceedings of the 30th ACM International Conference on Multimedia.* https://doi.org/10.1145/2964284.2964325
11. Samel, K. (2017). Predicting advertisement clicks using deep networks: Interpreting deep learning models. *Journal of Purdue Undergraduate Research,* 7, 50–56. https://doi.org/10.5703/1288284316397

12. Xu, R., Wang, M., & Xie, Y. (2018). Optimally connected deep belief net for click through rate prediction in online advertising. *IEEE Access,* 6, 43009–43020. https://doi.org/10.1109/access.2018.2861429
13. Jiang, Z., Gao, S., & Li, M. (2018). An improved advertising CTR prediction approach based on the fuzzy deep neural network. *PLoS ONE,* 13(5), e0190831. https://doi.org/10.1371/journal.pone.0190831
14. Su, Y., Jin, Z., Chen, Y., Sun, X., Yang, Y., Qiao, F., Xia, F., & Xu, W. (2017). Improving click-through rate prediction accuracy in online advertising by transfer learning. *Proceedings of the International Conference on Web Intelligence,* 1018–1025. https://doi.org/10.1145/3106426.3109037
15. Xi, X., Leng, S., Gong, Y., & Li, D. (2024). An accuracy improving method for advertising click through rate prediction based on enhanced xDeepFM model. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2411.15223
16. Chaitanya, K., Ramisetty, H. L., Shaik, R., Gorli, T., & Katta, L. (2024). Ads Click-Through Rate prediction using Attention based LSTM Mechanism. *2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS),* 1–6. https://doi.org/10.1109/sceecs61402.2024.10481965
17. Vieira, A. (2015). Predicting online user behaviour using deep learning algorithms. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.1511.06247
18. Zou, D., Wang, Z., Zhang, L., Zou, J., Li, Q., Chen, Y., & Sheng, W. (2021). Deep Field Relation Neural Network for click-through rate prediction. *Information Sciences,* 577, 128–139. https://doi.org/10.1016/j.ins.2021.06.079
19. Livne, A., Dor, R., Mazuz, E., Didi, T., Shapira, B., & Rokach, L. (2020). Iterative boosting deep neural networks for predicting Click-Through rate. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2007.13087
20. Baqapuri, A. I., & Trofimov, I. (2014). Using neural networks for click prediction of sponsored search. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.1412.6601