ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

Box-Office Revenue Predictions based on XGBoost and Sentiment Analysis

Mengtong Xu[1], Demin Wei[2, a, *], Tianrui Zhu[3], Yubo Zhang[4]

[1]School of Natural Sciences, The University of Manchester, Manchester M13 9PL, UK.

[2]School of Mechanical Engineering & Automation, BUAA, Beijing 100191, China.

[3]College of Computing, Wuhan university, Wuhan 430072, China.

[4]Immaculate High School Danbury, Connecticut 06810, US.

[a]E-mail: 824423093@buaa.edu.cn

Abstract

Sentiment analysis of movie comments can obtain a lot of valuable information, which could be used to predict the Box-Office Revenue (BOR). In this study, LSTM model is used to analyze comment datasets extracted from IMDb. Then a model based on XGBoost is carried out to predict BOR. This work combines the two models as a new approach. The experimental results show that this model has high prediction accuracy. This approach will help film-makers to predict the success of their movies.

Keywords

Sentiment analysis, XGBoost, LSTM.

1. INTRODUCTION

In this paper, a machine learning-based system is demonstrated to predict box office revenue. In 2019, worldwide box office receipts topped $42.5 billion. With such a vast market, it makes sense to predict box-office. This work use machine learning as tool.

The essence of movie prediction is to assess whether a movie's attributes will entice people to go to the theater. The cast is a measure of how much star power the film can bring. Enough star power can make fans overlook shortcomings of a film and to bring in the proceeds [1]. When the release season is taken into account, the impact of competitors on the film's box office can be compared horizontally.

Except objective variables, some subjective factors, such as word-of-mouth (WOM), also impact revenues [2]. A McKinsey & Company study illustrates that more than two-thirds of sales volumes are based on WOM, showing the influence of it to purchase intent [3]. Moreover, extracting underlying sentiment from film reviews is vital to study costumer behavior [4, 5]. Liu and other researches developed ways to measure the popularity of films and to predict consumer intentions and box-office revenues [6].

Sentiment analysis contributes to the accuracy of Box-Office Revenue, BOR, predictions and users purchase intention could result in more accurate BOR prediction with the use of Support Vector Regression and Linear Regression [6]. A random forest model which is based on sentiment score of the movie tweets and movie expert reviews gives great result in BOR predictions [7]. The BOR prediction considering movie consumption intention is higher than without considering it. SVM and CIMM models have yielded high accuracy on identifying consumer intentions [8]. Combining comments, the BP neural network model could predict BOR accurately [9]. LSTM-based sentiment models provide strong support for the accuracy of

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

BOR. It incorporates emotional characteristics to traditional features. Sentiment analysis combining CNN with LSTM produces a model whose BOR prediction accuracy is 88.94% [10]. Among these studies, many good models have been proved. A Deep-DBP model was proposed. Only using temporal components, the prediction accuracy is high [11].

In this research, XGBoost is used as the main model. In predicting BOR, this model is less studied. The novelty of this research highlights the coupling of sentiment analysis with general film information. This research presents a model based on XGBoost to make BOR predictions. Specifically, a LSTM model is built to analyze sentiment characteristics. Then, combine the LSTM model and XGBoost model, presenting a new approach to make accurate BOR predictions. This approach will help film-makers to predict the success of their movies before the release, and helps people to understand the factors' influence on BOR.

2. DATA

From the Internet Movie Database website (imdb.com), two datasets for 1000 movies, choosing to predict those box offices, are obtained. One is the general information about the films, ranging from the title, genre, descriptions, director, actors, year of release, runtime of film, rating, votes to the revenues. The other is the top 25 reviews on the website for each film.

In addition, movies reviews (from github.com), sorted as positive and negative, are downloaded to train the sentiment analyzing model. There are 25,000 sorted comments, in which positive and negative comments are evenly split.

For sifting data, first delete the missing data and convert quantitative value into float data type. Table 1 shows the resulting data:

Table 1. Examples of samples

	Title	Genre	Description	Director	Rating	Votes	Review 1th
1	Guardians of the Galaxy	Action	A..universe.	James Gunn	8.1	757074	Guardians…
2	Prometheus	Adventure	Following …alone.	Ridley Scott	7	485820	It appears…
3	Split	Horror	Three …24th.	M. Night Shyamalan	7.3	157606	A fantastic..

Then, get rid of special symbols and other special characters like "?!;:#$ and convert data into vector form by Word2Vec.

3. METHODOLOGY

3.1. Research Strategy

This research's aim is to predict movie BOR. To achieve this, movie comments are used to train a LSTM model, perform sentiment analysis, and get the movie sentiment score. Then a XGBoost model for BOR prediction is built. The data is combined with the sentiment analysis results as the model's input to train the model.

Dichotomy is used to perform sentiment analysis, dividing the comments into two categories, positive and negative, with score of 1 and 0 respectively. Then, the number of comments of each category multiplies by the corresponding comment score, divided by the total number of comments, which is the sentiment score.

The model uses xgboost library to make regression. This work focus on 9 variables: genre, description, director, year, runtime, rating, votes, metascore, and sentiment score.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

3.2. Long Short-term Memory (LSTM)

Long short-term memory, LSTM, is a model with several gates on the standard recurrent neural network usually used in deep learning [12]. LSTM model can remember information for a long time, avoid long-term dependency problem, and can learn independently. A common LSTM unit contains 3 gates: forget gate, input gate and output gate [13].

3.2.1 Variables

$x_t \in R^d$: input vector to the LSTM unit

$f_t \in R^h$: forget gate's active vector

$i_t \in R^h$: input/update gate's activation vector

$o_t \in R^h$: output gate's activation vector

$h_t \in R^h$: hidden state vector also known as output vector of the LSTM unit

$\tilde{C}_t \in R^h$: cell input activation vector

$C_t \in R^h$: cell state vector

$W \in R^{h \times d}$, $W \in R^{h \times h}$ and $b \in R^h$: weight matrices and bias vector parameters which need to be learned during training

σ: $sigmoid$ function

[13]

3.2.2 Detailed structure

3.2.2.1The forget gate

Forget gate shown in Figure 1 determines what information is discarded. h_{t-1} and x_t are first spliced together and then passed to a sigmoid function. Then, generate a value f_t between 0 and 1, which directly determines how much state information is retained [14].

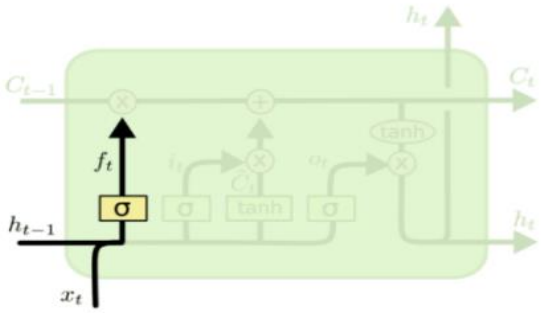

Fig 1. The Forget Gate

3.2.2.2 The input gate

Input gate determines what new information to add. As the left picture in Figure 2 illuminated, this step contains two layers: a tanh layer and a $sigmoid$ layer. The former outcome, from -1 to 1, is used to produce the updated value candidate for $\tilde{C}_t$. Then it is multiplied by the outcome of the $sigmoid$ layer to provide a scaling effect. If the $sigmoid$ output is 0 in the extreme case, then the cell state on the corresponding dimension does not need to be updated [15].

Another picture in Figure 2 demonstrates the step to multiply the old cell state C_{t-1} to f_t to discard some information and add the new information part $i_t \times \tilde{C}_t$ to get the new cell state C_t.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

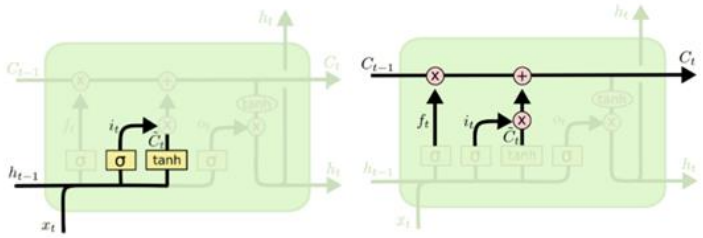

Fig 2. The Input Gate

3.2.2.3 The output gate

Figure 3 elucidate the last step of deciding what to output. The output value is related to the cell state, and C_t is given to a candidate by a $tanh$ function. Eventually, which parts of the candidate are output is determined by a $sigmoid$ layer.

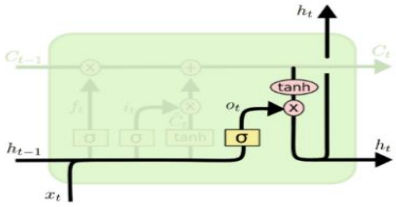

Fig 3. The Output Gate

3.2.3 Key equation and realization of the LSTM

LSTM use a particular initial state h_0 prior to start the spread from the first data (t = 1) to the last (t =N). Each time, update equation for $f_t, i_t, o_t, \tilde{C}_t, C_t$ and h_t are shown as follow:

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{1}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{3}$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{4}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{5}$$

$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

The equation parameters are updated by means of the back propagation of LSTM by gradient descent.

3.3. XGBoost

XGBoost is an essential Gradient Boosted Decision Tree, GBDT, but strive for maximum speed and efficiency [16]. The original GBDT algorithm is based on the negative gradient of the empirical loss function [17] to construct the new decision tree. While XGBoost adds regular terms during the decision tree construction phase.

$$L_t \sum_i l\left(y_i, F_{t-1}(x_i) + f_t(x_i) + f_t(x_i)\right) + \Omega(f_t) \tag{7}$$

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

$F_{t-1}(x_i)$ represents the optimal solution of the existing t-1 tree. The regular term is:

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{8}$$

T is the number of leaf nodes, and w_j is the predicted value of the J leaf node. The second order Taylor expansion of the loss function at F_{t-1} can be derived [18].

$$L_t \approx \tilde{L}_t = \sum_{j=1}^{T} \left\{ G_j \omega_j + \frac{1}{2}\left(H_j + \lambda\right)\omega_j^2 \right\} + \gamma T \tag{9}$$

T is the number of leaf nodes in the F_t of the decision tree,

$$G_j = \sum_{i \in I_j} \nabla F_{t-1} l(y_i, F_{t-1}(x_i)) \tag{10}$$

$$H_j = \sum_{i \in I_j} \nabla^2 F_{t-1} l(y_i, F_{t-1}(x_i)) \tag{11}$$

It represents the combination of the indexes of all the samples belonging to leaf node j [17].

XGBoost sorts the data in advance before training, then saves it as a block structure, which makes parallelism possible. In the process of node splitting, it is necessary to calculate the gain of each feature, and finally choose the largest feature to split, so that the computation of each feature can be carried out with multi-threading.

When the tree node is split, it turns to calculate the gain corresponding to segmentation point of each feature, aiming to enumerate all possible segmentation points by the greedy method [19]. Figure 4 shows the exact greedy algorithm for split finding:

Algorithm 1: Exact Greedy Algorithm for Split Finding

Input: I, instance set of current node
Input: d, feature dimension
gain $\leftarrow 0$
$G \leftarrow \sum_{i \epsilon I} g_i, \quad H \leftarrow \sum_{i \epsilon I} h_i$
for $k = 1$ **to** m **do**
  $G_L \leftarrow 0, \ H_l \leftarrow 0$
  for j *in sorted (I, by* x_{jk}*)* **do**
    $G_L \leftarrow G_L + g_j, \ H_L \leftarrow H_L + h_j$
    $G_R \leftarrow G - G_L, \ H_R \leftarrow H + H_L$
    $score \leftarrow \max(\text{score}, \frac{G_L^2}{H_L+\delta} + \frac{G_R^2}{H_R+\delta} - \frac{G^2}{H+\delta})$
  end
end
Output: Split with max score

Fig 4. Extract Algorithm for Split Finding

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

To make up for the inefficient of Greedy algorithm, XGBoost proposes a parallel approximate histogram to generate candidate segmentation points [20]. Figure 5 illustrates the approximate algorithm for split finding:

Algorithm 2: Appriximate Algorithm for Split Finding

for $k = 1$ to m do

  Propose $S_k = \{s_{k1}, s_{k2}, \cdots s_{kl}\}$ by percentiles on feature k.

  Propose can be done per tree (global), or per split (local).

end

for $k = 1$ **to** m **do**

  $G_{kv} \leftarrow= \sum_{j \in \{j | s_k, v \geq X_{jk} > s_{k,v-1}\}} g_j$

  $H_{kv} \leftarrow= \sum_{j \in \{j | s_k, v \geq X_{jk} > s_{k,v-1}\}} h_j$

end

Follow same step as in previous section to find max score only among proposed splits.

Fig 5. Approximate Algorithm for Split Finding

4. RESULT

After training the LSTM model with the data-set of positive and negative reviews, the accuracy (F1-score) is of 88.0%, indicating the model can analyze sentiment implicit in comments quite well. Then, use the model to get sentiment score of specific movies through the top 25 reviews each. In Table 2, examples and summary of the outcomes are given:

Table 2. Summary of sentiment score

	Movie	Score
Maximum	About Time	0.998112
Minimum	Wrecker	0.003106
Median		0.5006089
Mean		0.57137664

Note: This table is a summary of the sentiment scores in the dataset. The generated scores ranged from 0.003106 to 0.998112, with the mean of 0.571377.

Figure 6 is a scatter plot showing the distribution between viewers' sentiment and revenues. Revenue seems to depend on the sentiment score, which reveals a correlation between the two variables. Besides, there exists several movies not earning much money, regardless of the sentiment grade.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

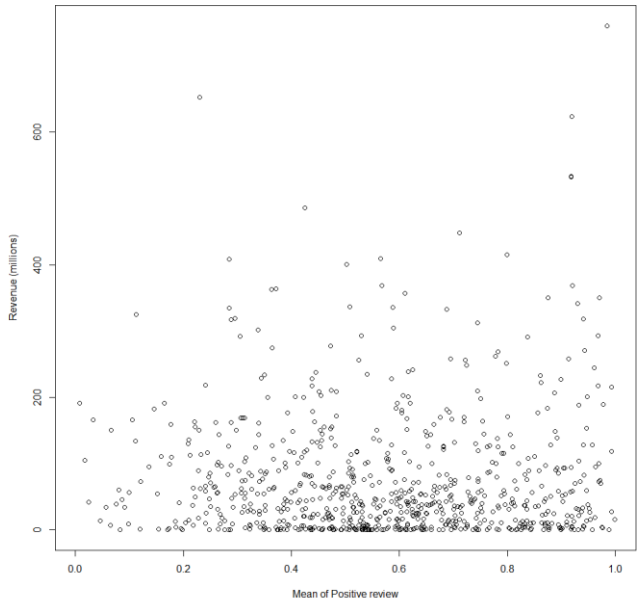

Fig 6. Scatter Plot Between Sentiment Score and Revenues

In Figure 6, some outliers with high revenues but low scores can also be experienced. Except those outliers, the income range nearly has a direct relationship with the emotional scores.

Table 3. The outliers and diversities

Movie	Diversity
Twilight	25217.617
Robin Hood	6141.507
Fifty Shades of Grey	4937.682
Suicide Squad	2886.088
Jurassic World	2832.454
Neighbors	2267.702
Aliens vs Predator - Requiem	1713.030
2012	1582.081
The Hunger Games	1435.315
Lincoln	1253.505

Table 3 record the top 10 abnormal values with the highest diversity. However, these include movie sequels, movies shown during the New Year period, etc. As a result, even though those movies are not highly recommended or favored by reviewers, they continue to enjoy high box office. And that might be the reason why they become outliers. Then, Figure 7 get rid of the big deviation points and the characteristics and trend are more evident.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

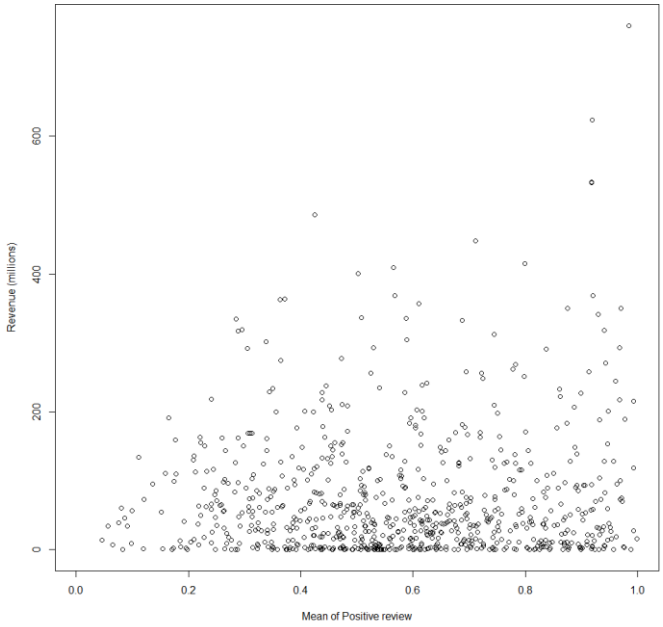

Fig 7. Scatter Plot Between Sentiment Score and Revenues without 10 Outliers

Figure 8 selects the maximum revenues corresponding to 60 different scores and calculate the correlation between scores and revenues for these points. The Pearson correlation coefficient of the regression line is 0.81.

Thus, there is no evidence to contradict that box-office sales are related to the sentiment score.

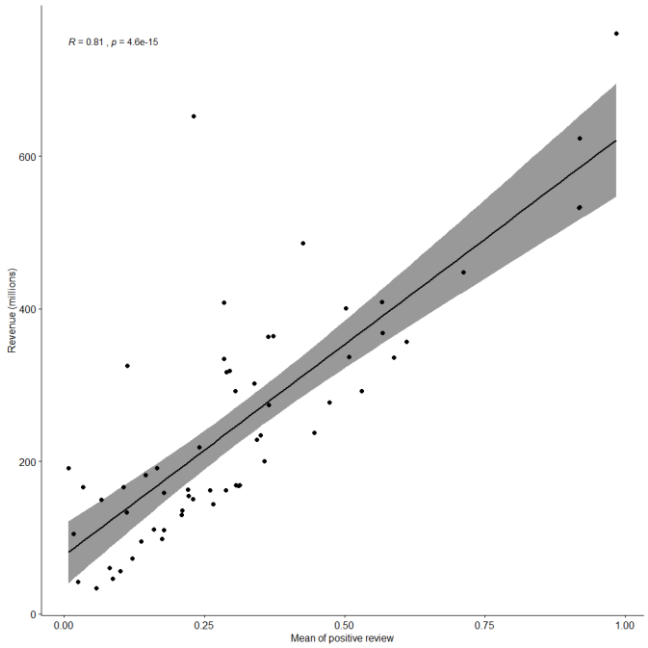

Fig 8. Regression Line for Boundary Points

Moreover, other variables are considered to predict sales with the method of XGBoost. In Figure 9, yellow, green, and blue curves correspondingly represent the true value of the revenues, predictive revenues without considering emotion and that considering sentiment.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

Comparing with those situations without considering sentiment scores, those including sentiment analysis have higher accuracy and are closer to the actual.

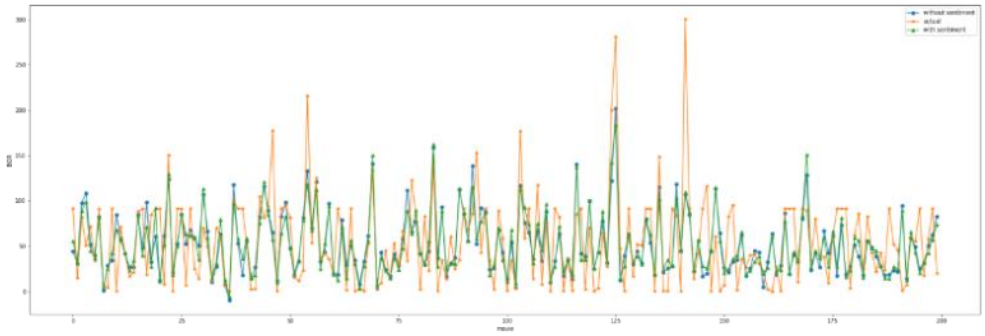

Fig 9. Revenues and Two Kinds of Predicted Revenues

Thus, people's emotion and feedback to the movie can influence the revenues and should be treated as one of the independent variables when estimating the box-office sales.

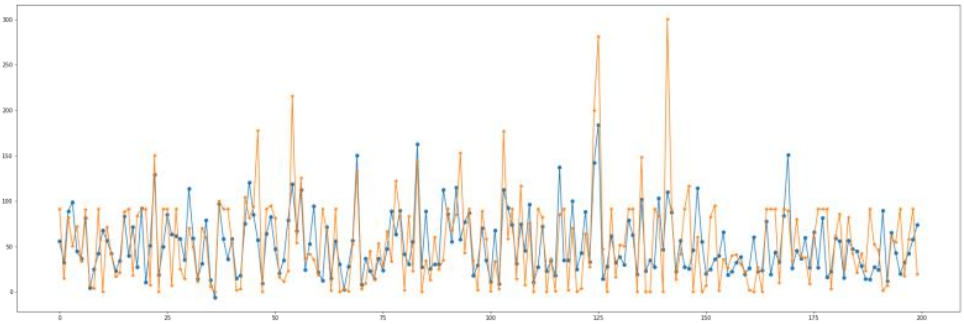

Fig 10. Revenues and Predicted Revenues

In Figure 10, yellow curve shows the prediction, and blue curve represents the actual revenue. From this figure, it can be seen that the prediction result is basically in line with the actual revenue, and error range is mostly within the acceptable range.

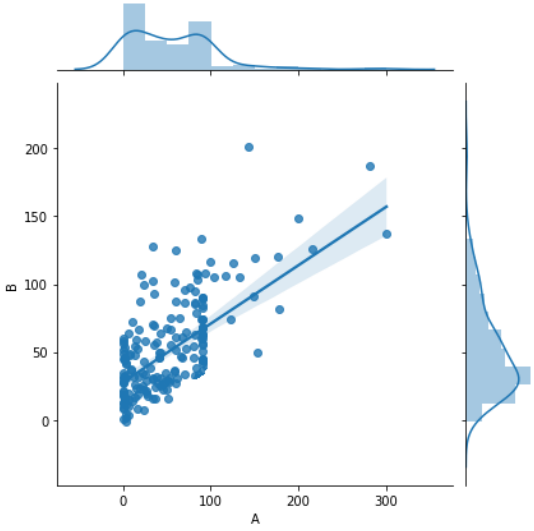

Fig 11. Regression Line between predictions and actual values

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

Figure 11 shows the regression line of predictions and actual values with the Pearson's correlation coefficient of 0.63.

5. CONCLUSION

In this paper, an XGBoost model for Box-Office Revenue (BOR) is proposed. First, an LSTM model is built to calculating the sentiment score of movie comments. Then, an XGBoost model is built, leveraging sentiment score and other data, to make movie box-office revenues predictions. Experiments on the data-set extracted from IMDb demonstrated that this model has better performance than previous research. In comparison with the actual revenue, the predicting result yield high accuracy and practicability [21].

This research reflects a high research value and provides a framework. Future research will focus on expanding the data-set such as adding more variables and finding better methods to quantify some variables such as contributing factors.

REFERENCES

[1] Amos, C., Holmes, G., & Strutton, D. (2008). Exploring the relationship between celebrity endorser effects and advertising effectiveness. International Journal of Advertising, vol.27, no.2, pp.209–234.

[2] Neelamegham, R., & Chintagunta, P. (1999). A Bayesian Model to Forecast New Product Performance in Domestic and International Markets. Marketing Science, vol.18, no.2, pp.115–136.

[3] John, T. (2003). Word of mouth is where it's at. Brandweek, vol.44, no.22, pp.26.

[4] Wang, J., Yu, L.-C., Lai, K. R., & Zhang, X. (2016). Dimensional Sentiment Analysis Using a Regional CNN-LSTM Model. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, vol.2, pp.225-230.

[5] Zhao, J., Dong, L., Wu, J., & Xu, K. (2012). MoodLens. Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD 12, pp.1528-1531.

[6] Liu, T., Ding, X., Chen, Y., Chen, H., & Guo, M. (2014). Predicting movie Box-office revenues by exploiting large-scale social media content. Multimedia Tools and Applications, vol.75, no.3, pp.1509–1528.

[7] Ruus, R., & Sharma, R. (2019). Predicting Movies' Box Office Result - A Large Scale Study Across Hollywood and Bollywood. Complex Networks and Their Applications VIII Studies in Computational Intelligence, pp.982–994.

[8] Fan, W. (2019), Study on film release day box office prediction considering competition and consumption intention. Ph.D-Beijing University of technology.

[9] Wang, X., Yuan, Y., & Shi, L. (2016). Predicting Opening Weekend Box Office Prediction Based on Microblog. Data Analysis and Knowledge Discovery, vol.32, pp.31–38.

[10] Yenter, A., & Verma, A. (2017). Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis. 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), pp.540-546.

[11] Yunian Ru, Bo Li, Jianbo Liu, Jianping Chai (2018). An effective daily box office prediction model based on deep neural networks. Cognitive Systems Research, vol.52, pp.182–191.

[12] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, vol.9, no.8, pp.1735–1780.

[13] Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. Neural Computation, vol.12, no.10, pp.2451–2471.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202011_6(11).0007

[14] Han, J., & Moraga, C. (1995). The influence of the sigmoid function parameters on the speed of backpropagation learning. Lecture Notes in Computer Science from Natural to Artificial Neural Computation, pp.195–201.

[15] Greff, K., Srivastava, R. K., Koutnik, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A Search Space Odyssey. IEEE Transactions on Neural Networks and Learning Systems, vol.28, no.10, pp.2222–2232.

[16] Friedman, J. (2001). Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics, 2, vol.29, no.5, pp.1189–1232.

[17] Wang, B., Yang, K., Wang, D., Chen, S.-Z., & Shen, H.-J. (2019). The applications of XGBoost in Fault Diagnosis of Power Networks. 2019 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia), pp.3496-3500.

[18] Lei, Y., Jiang, W., Jiang, A., Zhu, Y., Niu, H., & Zhang, S. (2019). Fault Diagnosis Method for Hydraulic Directional Valves Integrating PCA and XGBoost. Processes, vol.7, no.9, pp.589.

[19] Chang, W., Liu, Y., Xiao, Y., Xu, X., Zhou, S., Lu, X., & Cheng, Y. (2019). Probability Analysis of Hypertension-Related Symptoms Based on XGBoost and Clustering Algorithm. Applied Sciences, vol.9, no.6, pp.1215.

[20] Yi, H.-C., You, Z.-H., Wang, M.-N., Guo, Z.-H., Wang, Y.-B., & Zhou, J.-R. (2020). RPI-SE: a stacking ensemble learning framework for ncRNA-protein interactions prediction using sequence information. BMC Bioinformatics, vol.21, no.1, pp.1-14.

[21] Lu, W., & Xing, R. (2019). Research on Movie Box Office Prediction Model with Conjoint Analysis. International Journal of Information Systems and Supply Chain Management, vol.12, no.3, pp.72–84.