SAJSET
ISSN 2992-3581

Savannah Journal of Science and Engineering Technology
Volume: 01; Issue: 02; Pages: 16-24; 2023.
www.sajsetjournal.com.ng

DOI: xx.xxxx.x/sajset-01-2023-0014

SAJSET
ISSN 2992-3581

Investigating the role of negative sampling algorithm for employee turnover prediction using improved dynamic bipartite graph embedding

Abdullahi, J. A.,[1] Sani, U.[1] and Garba, A.[2]

[1]*Department of Computer Science, Kano State Polytechnic, Kano, P.M.B.3401, Kano, Nigeria.*
[2]*Department of Computer Science, Faculty of Computer Science and Information Technology, Bayero University, Kano, 3011 Nigeria.*

Paper History

Received: 15th March, 2023
Accepted: 26th April., 2023
Published: 10th May, 2023

Corresponding author
Abdullahi, J. A.
abdullahitariwa@gmail.com

Abstract

Employee turnover has become a big challenge for organizations, specifically in the present competitive environment where experienced employees are the biggest asset of an organizations. Depending on how difficult it is to get the equivalent replacement, the cost of an experienced employee voluntary turnover is ranging from 1.5 to 5 times the employee's annual salary. To some extent such losses affect organizational processes and alter the efficiency of their targeted plan. In an attempt to address the problem, previous researches centered on examining the impact factors such as employees and organization's attributes. This work emphasizes on modelling employee's time- related historical job experience as a dynamic bipartite graph to study a vector representation of employees and organizations. This is achieved by developing a model that generates a sequence for each vertex in the bipartite graph using a horary random walk (HRW) method and input the sequence to a skip-gram with negative sampling (SGNS) to get the vector representation for each vertex. We then combine the vectors with the employee's basic information as input to machine learning classifiers to predict the employee's turnover. The data was collected from one of China's recruitment platform. Precisely, this study proposed a low complexity model called Improved Dynamic Bipartite Graph Embedding (IDBGE). Furthermore, the experimental results shows that our technique had significantly improved the performance of the employee turnover prediction on recall, F1, AUC_ROC with 4.24%, 4.43%, and 2.82% respectively.

Keywords: *Bipartite graph, Graph embedding, Horary random walk, Linear discriminant analysis, Skip-gram sampling*

1. Introduction

Employee turnover is defined as a disclosure or leaving of an intelligent skills person from industry or organization [1]. The development of information technology, particularly with the implementation of human resource information systems (HRIS) in organizations, has triggered human resource managers and professionals to further make use of the data collected over time to enhance decision making [2]. Researchers have proven the significance of machine learning in some areas like commerce, finance, and health to reach a better group of customers or reduce the risk of investment for better performance [3, 4]. But the use of advanced data analytics in the human resources (HR) area is still limited [5, 6]. This drives our motivation to further explore this area by analyzing the data to predict employee's turnover behaviour more effectively [7].

Additionally, the cost of employee turnover is ranging from 1.5 to 5 times the employee's annual pay sum, depending on how difficult it is to make his/her replacement [8]. Sometimes, it demanded a couple of times and money for the organization to get the equivalent replacements and get them trained [9]. As such, ability to forecast the likelihood of an employee to resign will allow the organization to take appropriate action to minimize the cost. In this work, we emphasize on voluntary turnover of the employee. The involuntary turnover is mainly triggered by the organization HR section; hence, it's not considered in this study. There are researches concerning employee turnover behaviour, such as that of Dave *et al.* [10], Yadav [11], Xu *et al.* [2] and Xu *et al.* [12], that used machine learning techniques to classify and predict the employee turnover at different organizational sectors; While some used survey to figure out the factors that lead to an employee's turnover decision [13, 14]. However, most of those research type concentrate on the impacting factors like employee's demographic information while disregarding the employee's working experience network structure; This in turn, may lead to the loss of some vital information that when incorporated may aid the prediction performance. On the other hand, the network-based researches such as Feeley *et al.* [15] and Vardaman *et al.* [16], are mainly restricted to the use of static network features, and because employee's working history record is mainly in sequential order (i.e., in the order of their occurrence time), static network-based features cannot maintain the time-related information contained in the employee's work experience [3], while others are computationally expensive.

In consideration of the above limitation, current study propose a graph embedding technique that lower the computational complexity and enhance performance. This decision

become possible considering that, employee's working experience record can be depicted using a bipartite graph through partitioning the vertices in-to employee's and organization's vertices respectively as shown in Fig. 1.

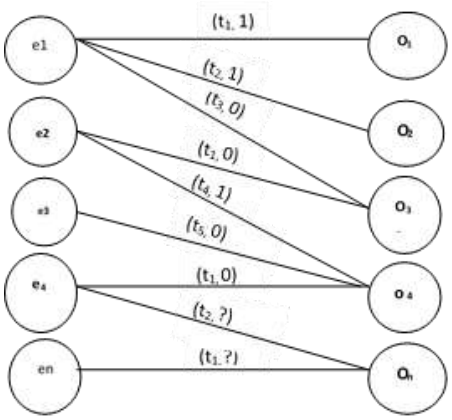

Figure 1: Dynamic Bipartite Graph

To efficiently find a way out for the employee turnover prediction problem with time dependence information we proposed IDBGE approach. Furthermore, dealt with the dynamic bipartite graph embedding problem, we employed HRW, proposed by Cai *et al.* [3] (which considered the time dependence information when carrying out the random walk) to generate a sequence for each vertex in the bipartite graph. This sequence (the output of the HRW) is then inputted in to SGNS model to get a low-dimensional vector representation for each vertex sequence. SGNS is proven to be the model that can produce a high-quality vector representation for a given sequence [17-22]. Ultimately, the network-based features were merged with the employee's basic information and use them as input to various classic machine learning classification algorithms to dealt with the employee turnover prediction problem more effectively. In short, this study contributes to the existing body of knowledge as indicated below.

a. Proposes an IDBGE model that make used of a negative sampling approximation algorithm to reduce the computational complexity and significantly enhance the quality of variables (vectors generated).
b. Indicates the role of Linear discriminant analysis (LDA) in extracting a good component that, when added to the employee's basic information will alleviate the problem of alternating through a series of split reduce component (as applied when Principal Component Analysis (PCA) is used), and significantly improve the prediction performance of employee turnover for the classifiers used.
c. Evaluates and selects an appropriate classification technique with high performance in comparison with other states-of-the-art turnover prediction accuracy., The outcome also indicates that the techniques used can remarkably enhance the turnover prediction accuracy as expected.

The remaining part of the paper is organized in the following sequence. Section 2 presents the adopted methodology for the research. Section 3 is the experimental setup section 4 presents result and discussion and Section 5 is conclusion and future research direction.

2. Materials Methods

In this section, we explain the model of our IDBGE by describing some important concept used in the model and the method used to predict employee turnover.

2.1 Bipartite graph:

A bipartite graph is a structured network type with the properties that its vertices be partition in to two independent groups in such a way that no two vertices in the same partition be connected directly [3]. Base on the above definition we can state the following;

Definition 1: (Turnover Prediction Problem): Let $G = (X, Y, E, T)$ be a dynamic bipartite graph, describing an interaction between Employee (e), Organizations *(o)*, and an edge *(e, o, t)* connecting *"e"* to *"o"* within timeframe "*t*". The central objective of the turnover prediction problem is to predict whether *"e"* will terminate his/her appointment with *"o"* after a certain timeframe of length *"Δt"* based on the basic and interconnected-based features, [7, 23].

Definition 2: (Dynamic Bipartite Graph): A dynamic bipartite graph is a bipartite graph $G = (X, Y, E, T)$, $X \cup Y = V$, where X *and* Y represent the sets of two types of vertices, V is the set of all the vertices in the graph *"G"*, $E \in (X \times Y)$ defines the inter-set edges, and T is the timeframe of E. i.e. for each edge $e = (x,y,t) \in E$ connecting a vertex $x \in X$ and a vertex $y \in Y$, *"e"* is associated with a unique timeframe ' t', [8, 23].

Definition 3: HRW: Let $G = (X, Y, E, T)$ be a dynamic bipartite graph, the HRW choose a vertex (v) randomly from G, and the visitor begins to traverse from "v" to the neighboring vertex on the opposite partition of G, if the visitor is to visit $x \in X$, it will check all the edges in E_x that has not been visited before and chooses to visit the edge with minimum timeframe, else it will randomly choose an adjacent vertex to visit irrespective of whether the edge has been visited or not, the visitor terminate it visitation at a vertex $x \in X$ when there is no unvisited edge or it reaches the maximum length l [3].

2.2 Skip-Gram Negative Sampling (SGNS)

A SGNS is an unsupervised learning technique used in natural language processing (Word2Vec) to learn and represent text/node in form of vectors (known as Text/node Embedding). SGNS algorithm is designed to alleviate the deficiency of computational complexity (i.e., computational cost and fairly qualitative vector generation) as it is in Skip-Gram (SG) and Skip-gram Hierarchical SoftMax (SGHS) [24].

2.3 IDBGE Model

To overcome the existing employee turnover problem more effectively, we propose an IDBGE model that operate in the following phases;

a. First, after all the data pre-processing, the clean data was divided in to Network-based features and basic information features.
b. The network-based features was the low dimensional vectors for both employee and organization through IDBGE module.
c. All the necessary data extraction and data selection for both the network-based and basic information features respectively was performed using appropriate techniques.
d. Concatenate the obtained extracted and selected features (in c above) in a single file to obtain a more combine features.
e. Then, the combine features (in d above) were feed to the historically classic used machine learning classification algorithms to predict the employee turnover individually.
f. Finally, the prediction output of the selected used classifiers was evaluated to obtain the best classification.

The above phases are diagrammatically summarized in the Fig. 2.

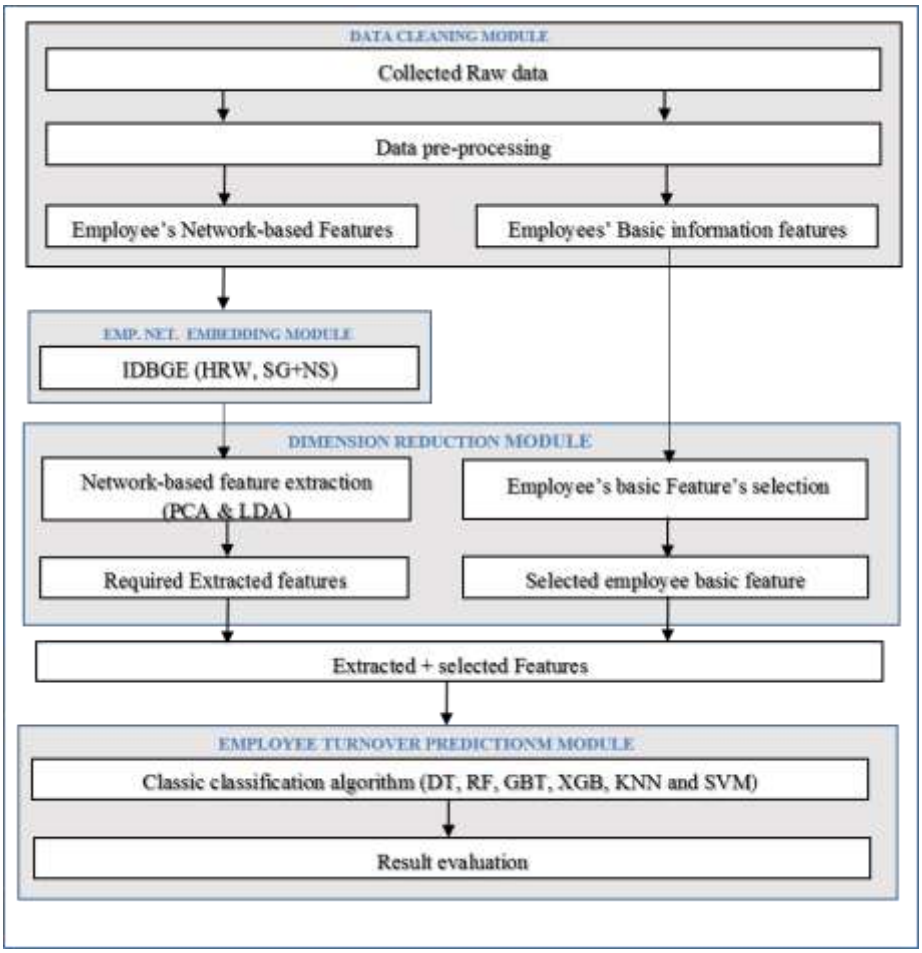

Figure 2: Propose IDBGE Model

2.4 Learning computational complexity of IDBGE

To learn the computational complexity of our proposed IDBGE algorithm, we use the concept of big O-notation to analyze the time complexity by the number of nested loops that the model iterates upon as [3];

$$O\ (l * \boldsymbol{E_{max}} * n * \$V\$ + w\ (d + d(k+1))). \quad (1)$$

Where: l = maximum walk length, E_{max} = the maximum degree of employee vertex, n = number of vertex's walk, V = E + O is the total of all vertices, and k = Number of negative samples drawn, w = window size and d= the dimension of the embedding.

Proof: Considering that the proposed model depends on the HRW algorithm, [3], which involves n-stages of a random walk with condition that, at each stage, if the current vertex (c_v) is an employee vertex, then it will traverse all of it adjacent edges to get the next un-visited edges having the least timeframe, which spends $O(E_{c}) \leq O(E_{max})$ *time*, where E_c is the degree of the current vertex. Otherwise (i.e., if the current vertex is an organization vertex), it will select at random, the nearest vertex to visit, which take complexity of O(1) time, as such, at worst case, the time complexity of HRW is: $O\ (l * E_{max} + 1) \approx O\ (l * E_{max})$ [3]. Moreover, we can easily deduce that the proposed IDBGE algorithm call HRW $(n * v)$ times through both the inner and the outer loop, appended with SGNS having it complexity equals to $O\ (w(d + d(k+1)))$ [25]. Base on the above scenario we can evaluate that the overall time complexity of the proposed IDBGE to be equals to; $O\ (l * \boldsymbol{E_{max}} * n * \$V\$ + w\ (d + d(k+1)))$ □ (end of proof)

2.5 Comparative analysis with state-of-the-art's complexity

Thus, when comparing the complexity of the proposed model; $O(l * \boldsymbol{E_{max}} * n * \$V\$ + w\ (d + d(k+1)))$ with its counterpart (the baseline work); $O(l * K_{max} * r * \$V\$ + w\ (d + d\log_2(\$V\$)))$, we can notice that, when updating the output neurons weight matrix (ONW), the baseline embedding model is computationally very expensive. As it requires scanning through the entire output embedding matrix to compute the probability distribution up to $\log_2(\$V\$)$ vertices for each training sample, where V can be millions or more. Thus, in contrast with the proposed embedding model that makes used of Negative sampling to update only k+1 weights ONW for each training sample. (Where K is a hyper-parameter that can be empirically fine-tuned, with a typical range of $ (5-20) $ vertices for a small dataset and $ (3-5) $ for a very large dataset [25, 26]. For example, with the proposed dataset having a total number of vertices V=70,714 and a number of the hidden (input) layers N = 128, has V*N = 9,051,392 ONW in the neuron's output weight matrix that requires an update for each training sample. Therefore, the baseline model reduces the above update to (Log2 (V)) * N = 2,048 ONW from the output weight matrix, which is still considered much. But with the proposed model, we reduce this update drastically to only (k+1) * N = 768 ONW (where k = 5) from the output weight matrix, i.e., the proposed model has saved about 1,280 Units of time and space compared to the baseline model.

Moreover, the space complexity of the propose embedding model is independent of the training data size and in the worse-case, it requires an O(n) space to store the vertex embedding and its frequency count (where n is the unknown variable).

Algorithm: IDBGE

Input: a dynamic bipartite graph $G = (X, Y, E, T)$; window size (w); embedding size (d); walks per vertex (n); and maximum length (l);

Output: Set of embedding vectors

Step1: Initialize walk to empty,

Step2: for iteration = 1 to r do,

Step3: for all vertices v ∈ (X U Y) do,

Step4: walk = Harary_Walk (G, v, l),

Step5: append walk to walks,

Step6: end for,

Step7: end for,

Step8: corpus = SG + NS (d, w, walks),

Step9: return corpus.

Figure 3: Pseudocode of Improve dynamic bipartite graph embedding algorithm

2.6 Data collection

This study makes use of data collected from one of the China's largest online employees recruitment platform called "51job. (www.51job.com), in its secondary data type form, covering 15 months from 1st January 2018. This data set is made up of a composition of an employee's historical Job record (which is modeled as a dynamic bipartite graph) and his/her basic information containing demographics and cognitive features such as age, gender, Highest qualification, Type of School awarding institution, organization type, organizational employee strength, date of first employment, place of primary assignment, the department posted, number of promotions, salary structure, salary-grade, and salary-level, Maximum turnover, etc. were recorded for the data mining study. Random sampling is used to select employee so that we can get accurate results without been biased.

Moreover, to limit our approach, we only take in to account an employee with minimum of two working experience records, moreover as part of inclusion and exclusion criteria, we set a time limit of 18-month as the time of turnover, i.e., if an employee leaves an organization after 18 months from the time of employment, then we consider it as a voluntary turnover and will be included in the graph else it will be considered as involuntary turnover and will be excluded.

2.7 Data Pre- Processing

This stage involves dataset preparation before applying the data mining techniques. However, in this context of employee turnover, the steps involved in data preprocessing are data cleaning which involves handling missing data and noisy data. Data transformation where the data is transformed into a form suitable for mining. Data scaling to scale the data within a common appropriate scale for all the features. Dimensionality reduction which involves data selection (Using forward feature selection and decision tree) and data extraction process using Linear Discriminant Analysis (LDA) which aims to optimize storage effective utilization as well as to retaining data information and reduce analysis costs.

After cleaning the data, we established a dynamic bipartite graph with 47,257 employee's vertices and 23,457 organization's vertices connected by 203,604 edges with their associated time frame. We also select the nine most relevant basic variables in order of their variation ranking, from the clean data set, these variables can be divided in to three categories as shown in the Table 1. From the dataset, we select 70% of the total summation at random and used it as training set and the remaining 30% are set as testing set.

Table 1: Variable's description

S/N	Variable Type	Variable Name
01	Demographics Variable	Gender, Highest certificate obtained.
02	Present work Variables	Date of employment, Ministry, Department or Agency (MDA) posted, Development levy, Pay Scale code, Position held.
03	Work experience Variables	Start year of waking career, Turnover status.

2.8 Classifiers

In this research, six classifiers are adopted as shown below.

a. Random Forest (RF): This approach is a classic bagging method in ensemble learning, [3, 27].
b. Extreme Gradient Boosting (XGBoost): This approach is a type of boosting method in ensemble learning, [3, 5].
c. Decision Tree (DT): This method is a tree structure in which each internal node represents a test on an attribute, each branch represents a test output, and each leaf node represents a category [28, 29].

d. Gradient boosting tree (GBT): GBT is a classic machine learning algorithm that iteratively constructs an ensemble (CART) of weak decision tree learners through boosting and reduce the sample loss. [1, 30].
e. K-nearest neighbors (KNN): is a supervised machine learning classifier that work operate by observing a distance between query and all example in the dataset, choosing a number of example (k-value) that is closer to the query and vote for the frequent label [1, 31].
f. Support vector machine (SVM): is an algorithm that work by creating the best decision boundary that can separate n-dimensional space into classes [27, 32].

2.9 Evaluation metrics

The following are the three-evaluation metrics in evaluating the proposed model [3]: F1 score: this is the harmonic average of precision and recall, i.e.

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (2)$$

Precision: A precision is defined as the proportion of positive cases in all samples classified as positive [3]. i.e.

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

Recall: is the proportion of samples that are predicted to be positive in all positive samples [3], i.e.

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

Where**:** *TP* = True Positive, *TN* = True Negative, *FN* = False Negative, *FP* = False Positive.

2.10 Parameter setting

As recommended in the work of [21], we use the hyperparameter setting as follows: *Window Size (w) = 5 Vertices, embedding size (d) = 128 dimension, Walk per vertex (n) = 80, Maximum walk length (l) = 15.* To efficiently learn the low-dimensional vectors for both employees and organizations vertices respectively, at the end of the IDBGE execution, we produced a 128-dimensional vector representation for each vertex. (64 vectors for the organization nodes and 64 for the employee nodes). However, to integrate the basic variable of employees with these vectors representation we apply LDA dimensionality reduction techniques on the obtained 128 low-dimensional vectors [13, 21].

As for the Classic machine learning classification algorithms used, we apply a hyperparameter tunning technique to obtain the hyperparameter values used as the most effective parameter setting for each classifier [13, 21].

3. Result and Discussion

3.1 Classifiers using employee basic information

Result of evaluation metric for the classifiers using employee basic information only are shown in Table 1. To confirm whether the proposed IDBGE model can deal with the employee turnover prediction problem more effectively, we simultaneously run and compare the prediction performance of the classifiers used, with and without the IDBGE variables respectively.

Table 1: Classifier's evaluation metrics performance with only employee basic features

S/N	Classifiers Used	Evaluation Metrics					
		Accuracy	Precision	Recall	F1	AUC-ROC	Average Precision
1	DT	98.17	93.26	88.17	90.65	93.73	83.43
2	RF	98.34	96.5	86.63	91.3	93.14	84.94
3	GBT	98.04	**98.73**	84.60	89.35	90.74	84.41
4	XGB	**98.83**	98.02	**89.97**	**93.82**	**94.88**	**89.18**
5	SVM	96.4	92.51	72.98	81.26	85.62	71.09
6	KNN	96.91	92.12	75.14	82.77	87.22	71.68

As can be seen from table 1 above, before inclusion of the IDBGE variable, XGB has the highest prediction performance in Accuracy, Recall, F1, AUC_ROC and Average Precision, having their corresponding prediction values equals to 98.83, 89.97, 93.82, 94.88 and 89.18 respectively. However, this has probably been achieved because it is an ensemble learning algorithm that can handle non-linear data more effectively. While after inclusion of the IDBGE variable (table 2), the outcome favor RF and XGB with high performance in the evaluation metrics.

3.2 Classifiers with IDBGE

Result of evaluation metric for the classifiers with IDBGE variables and employee basic information are shown in Table 2.

Table 2: Classifier's result evaluation table with 'IDBGE' variables reduce by LDA algorithm

S/N	Classifiers Used	Evaluation Metrics					
		Accuracy	Precision	Recall	F1	AUC – ROC	Average Precision
1	DT	98.40	92.81	91.18	91.99	95.20	85.52
2	RF	98.78	98.23	89.50	93.67	94.66	88.98
3	GBT	98.72	**98.98**	88.24	93.30	94.07	88.53
4	XGB	**99.10**	98.30	**92.53**	**95.33**	**96.18**	**91.70**
5	SVM	97.40	96.25	77.22	85.69	88.44	76.63
6	KNN	97.17	93.05	77.22	84.40	88.29	74.12

For both adopted classifiers, in Table 2, we observed that their prediction performances have further improved with the help of the proposed graph embedding variable, learnt through the IDBGE approach as compared to their respective performances presented in Table 1 without the IDBGE variables. This performance differences are tabulated in Table 3 below. Comparing Tables 1 and 2 above that indicated the result with both exclusion and inclusion of the propose IDBGE variables, it is found that XGB algorithm has the best performance having its prediction values as; Accuracy = 99.10, Recall = 92.53, F1 = 95.33, AUC_ROC = 96.18, and Average Precision = 91.70. However, this has probably been achieved because it is an ensemble learning algorithm that can handle non-linear data more effectively. Moreover, its ability to randomize it state at different execution has assist in reducing the effect of overfitting and improve its general performance.

Table 3: Performance improvement (in %) with IDBGE variables reduced by LDA algorithm

S/N	Classifiers Used	Evaluation Metrics					
		Accuracy	Precision	Recall	F1	AUC_ROC	Average Precision
1	DT	0.23	-0.45	3.01	1.34	1.47	2.09
2	RF	0.44	1.73	2.87	2.37	1.52	4.04
3	GBT	0.68	0.25	3.64	3.95	3.33	4.12
4	XGB	0.27	0.28	2.56	1.51	1.30	2.52
6	SVM	**1.00**	**3.74**	**4.24**	**4.43**	**2.82**	**5.54**
7	KNN	0.26	0.93	2.08	1.63	1.07	2.44

To measure the effectiveness of the reduced network-based feature (generated by the proposed model) on each adopted classifier. In Table 3 above, we further compute the difference in the prediction performances between the network-based embedding prediction (Table 2), and the basic employee variable-based prediction (Table 1) for each classifier with respect to the proposed evaluation metrics. Out of which we observe that, the most promising improvement is seen with the SVM having it corresponding increase in evaluation metrics values as: Accuracy= 1.0%, Precision = 3.74%, Recall = 4.24%, F1 = 4.43%, AUC_ROC= 2.84 and AVR-Precision = 5.54%. However, this is achieved by the fact that IDBGE model generate a set of vectors, and SVM is highly effective in dealing with vectors related data. Moreover, SVM is effective with data that has more observation (Instances) than the number of dimensions.

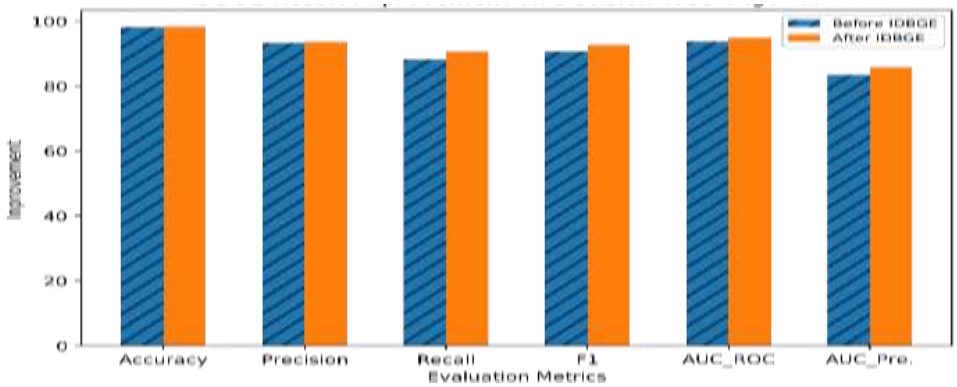

Figure 4: IDBGE result improvement with DT

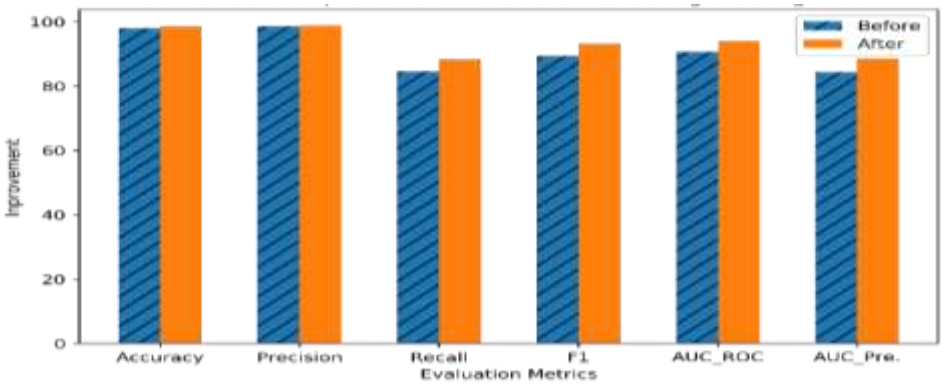

Figure 5: IDBGE result improvement with GBT

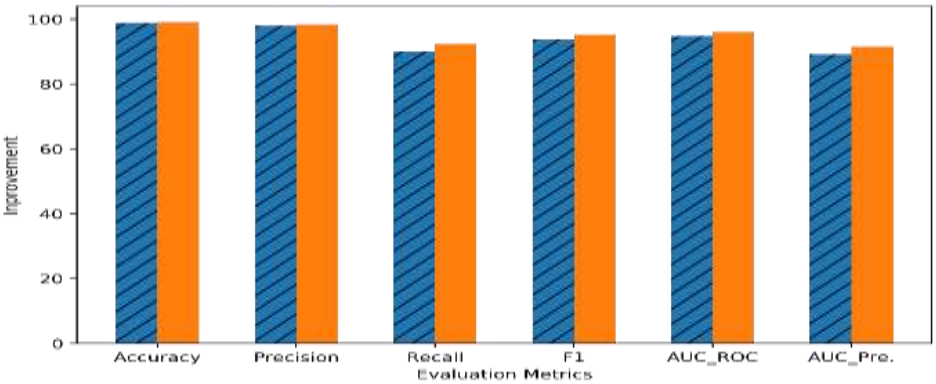

Figure 6: IDBGE result improvement with XGB

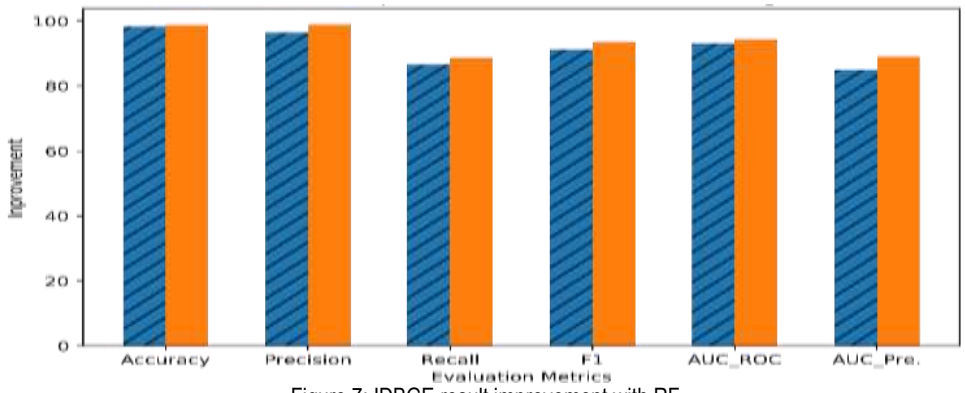

Figure 7: IDBGE result improvement with RF

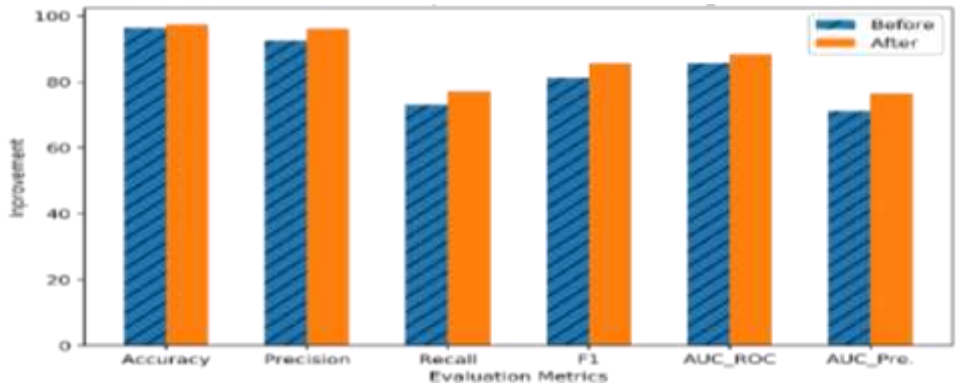

Figure 8: IDBGE result improvement with SVM

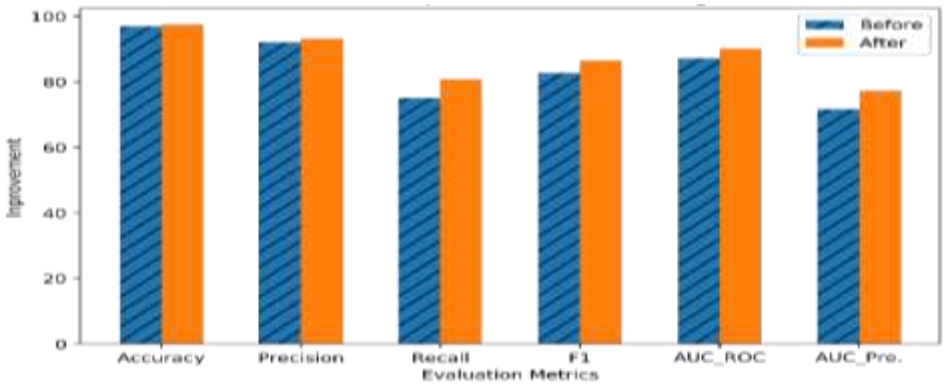

Figure 9: IDBGE result improvement with KNN

To further prove the prediction power of our IDBGE model, we computed, ranked and graphed the importance of all the (both the basic employee variable-based and network-based) variables using decision tree classifier. From Fig. 10 we observed that, the LDA variable has the best rank after comp. scale in the plot, and this has clearly indicated the prediction power of the proposed IDBGE variable in predicting employee turnover more effectively as it positively impacted the employee turnover prediction in all the adopted classifiers.

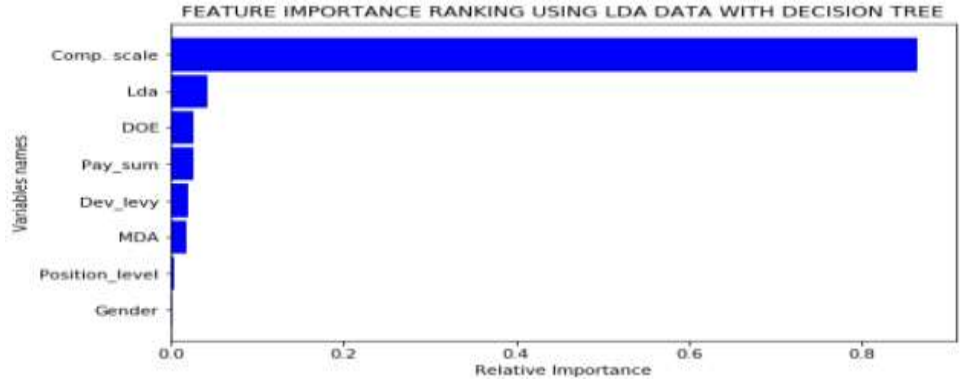

Figure 10: Decision Tree Variable importance ranking.

4. Conclusions

In this work, we proposed an IDBGE model that prove the positive role of negative sampling optimization algorithm in learning the network-based features as low dimensional vectors for both employee and organization. It reduces the computational complexity and assist in addressing employee turnover prediction problem more effectively. After performing all the necessary data extraction and data selection for both the network-based and basic information features respectively, we concatenate the obtained extracted and selected features in a single data-frame to obtain more Combine Features. Then feed the combine features to the most historically used classic machine learning classification algorithms to predict the employee turnover individually. Finally, we evaluate the prediction output of the selected classifiers to obtain the best classification. We conduct an experiment using a real-world data set and the result indicated that our proposed approach was effective in both simplifying the time and space complexity and also the employee turnover prediction problem.

Recommendations.

In the future, researchers might consider:

a. The effect of other natural language processing algorithms such as global vector (GloVe) in embedding the network-based and predict the employee turnover more accurately.
b. The use of different data mining algorithms like neural networks with large scale datasets to measure the effectiveness of the model.
c. Incorporating other features like change in environmental factors, change in the values of services rendered or change in market value.

References.

[1]. Alam, M. M., & Mohiuddin, K. (2019). *A Machine Learning Approach to Analyze and Reduce Features to a Significant Number for Employee's Turn over Prediction*. Springer International Publishing, Dhaka, Bangladesh. https://doi.org/10.1007/978-3-030-01177-2.

[2]. Xu, H., Yu, Z., Xiong, H., Guo, B., & Zhu, H. (2015). Learning Career Mobility and Human Activity Patterns for Job Change Analysis, 2015 IEEE International Conference on Data Mining, 31(3), 1057–1062. https://doi.org/10.1109/ICDM.2015.122.

[3]. Cai, X., Shang, J., Liu, F., Jin, Z., Qiang, B., & Xie, W. U. (2020). DBGE: Employee Turnover Prediction based on Dynamic Bipartite Graph Embedding. *IEEE Access*, https://doi.org/10.1109/ACCESS.2020.2965544.

[4]. Stamolampros, P., Korfiatis, N., Chalvatzis, K., & Buhalis, D. (2019). Job satisfaction and employee turnover determinants in high contact services: Insights from Employees' Online reviews. *Tourism Management*, *75*, 130–147. https://doi.org/10.1016/j.tourman.2019.04.030.

[5]. Jain, R., & Nayyar, A. (2018). Predicting Employee Attrition using XGBoost Machine Learning Approach. *2018 International Conference on System Modeling & Advancement in Research Trends (SMART)*.

[6]. Srivastava, D. K., & Nair, P. (2018). *Employee Attrition Analysis Using Predictive Techniques*. Information and Communication Technology for Intelligent Systems (ICTIS) *Smart innovation system and technology* 12(8), 35-39. https://doi.org/10.1007/978-3-319-63673-3.

[7]. Alao, D. (2013). *Analyzing Employee Attrition Using Decision Tree Algorithms*. Computing, Information Systems & Development Informatics, 4(1), 120-129.

[8]. Cai, H., Zheng, V. W., & Chang, K. C. C. (2018). A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, *30*(9), 1616–1637. https://doi.org/10.1109/TKDE.2018.2807452.

[9]. Peng, H., Li, J., Yan, H., Gong, Q., Wang, S., Liu, L., Wang, L., & Ren, X. (2019). Dynamic network embedding via incremental skip-gram with negative sampling, China information science, 12(9), 35, sarXiv:1906.03586v1. [cs.LG] 9 Jun 2019.

[10]. Dave, V. S., Aljadda, K., & Korayem, M. (2018). *A Combined Representation Learning Approach for Better Job and Skill Recommendation*. Association for Computing Machinery, CIKM'18, October 22-26, 2018, Torino, Italy. ACM ISBN 978-1-4503-6014-2/18/10, https://doi.org/10.1145/3269206.32720231997–2005.

[11]. Yadav, S. (2018). Early Prediction of Employee Attrition using Data Mining Techniques. *2018 IEEE 8th International Advance Computing Conference (IACC)*, 4(3), 349–354.

[12]. Xu, H., Yu, Z., Member, S., Yang, J., Xiong, H., & Member, S. (2018). Dynamic Talent Flow Analysis with Deep Sequence Prediction Modeling. *IEEE Transactions on Knowledge and Data Engineering*, *PP*(c), 1, 1-14, https://doi.org/10.1109/TKDE.2018.2873341

[13]. Chourey, A. (2019). *A survey paper on employee attrition prediction using machine learning techniques*. Journal of Interdisciplinary Cycle Research, *XI*(199), 199–202.

[14]. Punnoose, R. (2016). *Prediction of Employee Turnover in Organizations using Machine Learning Algorithms*. *(IJARAI) International Journal of Advanced Research in Artificial Intelligence, 5*(9), 22–26.

[15]. Feeley, T. H., Moon, S. l, Kozey, R. S., & Slowe, A. S. (2010). An erosion model of employee turnover based on network centrality. *Journal of Applied Communication Research*, *38*(2), 167–188. https://doi.org/10.1080/00909881003639544.

[16]. Vardaman, J. M., Taylor, S. G., Allen, D. G., Gondo, M. B., Amis, Taylor, S. G., Allen, D. G., Gondo, M. B., Amis, J. M., Intentions, T., Taylor, S. G., & Allen, D. G. (2015). *Translating intentions to behavior: the interaction of network structure and behavioral intentions in understanding employee turnover translating intentions to behavior. The Interaction of Network Structure and Behavioral Intentions in Understanding, Institute for Operations Research and the Management Sciences (INFORMS), 17(12), 1-15.*

[17]. Chen, H., Hu, Y., Perozzi, B., & Skiena, S. (2018). HARP: Hierarchical representation learning for networks. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 2127–2134.

[18]. Goldberg, Y., & Levy, O. (2014). Word to vector (*word2vec) Explained: deriving Mikolov et al, negative-sampling word-embedding method*. *2*, 1–5. http://arxiv.org/abs/1402.3722.

[19]. Saradhi, V. V., & Palshikar, G. K. (2011). Expert Systems with Applications Employee churn prediction. *Expert Systems with Applications*, *38*(3), 1999–2006. https://doi.org/10.1016/j.eswa.2010.07.134

[20]. Huang, Y., Yu, L., Wang, X., Cui, B., Vergara, P., Qian, Y., Tang, J. A., Yang, Z., Huang, B., Wei, W., Zhang, Y., Wallace, B., Kumar, S., Carley, K. M. K. M., Hirshman, B. R., Jones, L. A., Carroll, K. T., Tang, J. A., Proudfoot, J. A., (2016). Node to vector (node2vec) Real-time Video Recommendation Exploration Categories and Subject Descriptors. *World Neurosurgery*, *95*(1), 41–50. http://dx.doi.org/10.1016/j.ejso.2016.07.137%5Cnhttp://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/viewPDFInterstitial/6071/6379%5C

[21]. Perozzi, B., & Skiena, S. (2014). *Deep Walk: Online Learning of Social Representations Categories and Subject Descriptors*. ACM 978-1-4503-2956-9/14/08, 701-710. http://dx.doi.org/10.1145/2623330.2623732.

[22]. Tang, J., & Qu, M. (2015). LINE: Large-scale Information Network Embedding. International World Wide Web

Conference Committee (IW3C2). *P1067-Tang*. 1067–1077. http://dx.doi.org/10.1145/2736277.2741093.

[23]. Gao, M., Chen, L., He, X., & Zhou, A. (2018). BiNE: Bipartite network embedding. *41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2018*, *1*, 715–724. https://doi.org/10.1145/3209978.3209987.

[24]. Chen, J., Gong, Z., Wang, W., & Liu, W. (2021). HNS: Hierarchical negative sampling for network representation learning. *Information Sciences*, *542*, 343–356. https://doi.org/10.1016/j.ins.2020.07.015.

[25]. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). *Distributed Representations of Words and Phrases and their Compositionality, arXiv:1310.4546v1 [cs.CL] 16 Oct 2013.*

[26]. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013b). Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, 1–12.

[27]. Shi, C., Hu, B., Zhao, W. X., & Yu, P. S. (2019). Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *31*(2), 357–370. https://doi.org/10.1109/TKDE.2018.2833443.

[28]. Zhao, Y., Hryniewicki, M. K., & Cheng, F. (2019). *Employee Turnover Prediction with Machine Learning: A Reliable Approach*, Springer International Publishing, AISC 869, pp. 737–758, Toronto, Canada. https://doi.org/10.1007/978-3-030-01057.

[29]. Fan, C., Fan, P., Chan, T., & Chang, S. (2012). Expert Systems with Applications Using hybrid data mining and machine learning clustering analysis to predict the turnover rate for technology professionals. *Expert Systems with Applications*, *39*(10), 8844–8851. https://doi.org/10.1016/j.eswa.2012.02.005.

[30]. El-Rayes, N., Fang, M., Smith, M., & Taylor, S. M. (2020). Predicting employee attrition using tree-based models. *International Journal of Organizational Analysis*, *28*(6), 1273–1291. https://doi.org/10.1108/IJOA-10-2019-1903.

[31]. Khera, S. N. (2019). *Predictive Modelling of Employee Turnover in Indian IT Industry Using Machine Learning Techniques, journals.sagepub.com/home/vis, 23(1) 12–21,* https://doi.org/10.1177/0972262918821221.

[32]. Jaffar, Z., Noor, W., & Kanwal, Z. *(2019). Predictive Human Resource Analytics Using Data Mining Classification Techniques*. *International Journal of Computer (IJC), International Journal of Computer* (IJC), 32(*1*), *pp 9-20.*