AUIQ Technical Engineering Science

Volume 1 | Issue 1 Article 9

2024

Analysis of Predicting Co-Authorship Networks Using Support Vector Machine Model

Mohammed Y. Al-khuzaie
College of Nursing, National University of Science and Technology, Dhi Qar, Iraq,
mohammed.y.arabi@nust.edu.iq

Sajad Ali Zearah
Republic of Iraq Ministry of Agriculture, Thi-Qar, Outside North America, Iraq, sajad@alayen.edu.iq

Abbadullah H. Saleh
Graduate School of Natural and Applied Sciences, Gazi University, Ankara, Turkey, a.h.saleh@gazi.edu.tr

Recommended Citation

Al-khuzaie, Mohammed Y.; Zearah, Sajad Ali; and Saleh, Abbadullah H. (2024) "Analysis of Predicting Co-Authorship Networks Using Support Vector Machine Model," *AUIQ Technical Engineering Science*: Vol. 1: Iss. 1, Article 9.
DOI: https://doi.org/10.70645/3078-3437.1008

AUIQ TECHNICAL ENGINEERING SCIENCE 2024;1:106–115

ORIGINAL STUDY

Analysis of Predicting Co-Authorship Networks Using Support Vector Machine Model

Mohammed Y. Al-khuzaie [a], Sajad Ali Zearah [b,*], Abbadullah H. Saleh [c]

[a] College of Nursing, National University of Science and Technology, Dhi Qar, Iraq
[b] Republic of Iraq Ministry of Agriculture, Thi-Qar, Outside North America, Iraq
[c] Graduate School of Natural and Applied Sciences, Gazi University, Ankara, Turkey

ABSTRACT

The area of computer science is flourishing, as shown by the rising number of academic works published and the rising number of scholars adding to the existing body of knowledge. This has made it difficult to find previously unrecognized links between publications and those studying them. Using machine learning techniques to foresee missing links and unveil hidden connections, complex network-based link prediction has emerged as a helpful resource in addressing this difficulty. This study aims to solve the problem of finding links between authors' publications by exploring the use of network-based complex link prediction approaches. A large-scale bibliographic database collected from various reliable sources, including but not limited to Google Scholar, was used. A Support Vector Machine (SVM) classifier was used to predict the possibility of a new link between an author and his publication. Accuracy was one of the several criteria used to evaluate the performance of the SVM classifier, which was relatively high at 96.66%.

Keywords: Link prediction, Support Vector Machine (SVM), Complex networks, Hidden connections, Academic works

1. Introduction

Appraising employees' performance is a fundamental management task that pervades the whole business and helps people grow in their roles. Establishing a method for assessing academics' work is crucial in research settings like universities and organizations. This assessment, more than just a performance review, is predicated on researchers' output, especially their productivity. It's crucial for attracting top professors, getting government-funded research organizations, and gaining prestige in the academic world. Reputable research institutions contribute to societal well-being roundaboutly by drawing in international customers, investors, and talent pool members.

Scientific projects with many organizations working together to further knowledge might be classified as "collaborative research" [1]. It is generally agreed that there is a synergistic effect at work in these partnerships, with the sum of the parts more incredible than the parts themselves [2, 3]. However, there are significant obstacles to setting up and managing such research groups. Finding other researchers to work with is a significant challenge for solo scientists. Domain specialists suffer ambiguity when identifying ideal partnership prospects due to the inherent difficulties in anticipating which partnerships contain the best potential for success.

If researchers have complete knowledge of the research interests and active research efforts of specialists, they could be able to find a solution to this problem. With this data, researchers' domain competence might be evaluated, leading to the easier discovery of potential partners with complementary and synergistic areas of expertise. Unfortunately, there are no centralized sources from which to get this information. Therefore, it is often inaccessible and difficult to acquire.

Received 23 July 2024; accepted 16 August 2024.
Available online 27 August 2024

* Corresponding author.
E-mail addresses: mohammed.y.arabi@nust.edu.iq (M. Y. Al-khuzaie), sajad@alayen.edu.iq (S. A. Zearah), a.h.saleh@gazi.edu.tr (A. H. Saleh).

https://doi.org/10.70645/3078-3437.1008

Co-authorship networks may be easily constructed from published work if authors are represented as nodes and their collaborations as connections. Adamic and Common Neighbor topological properties inside these co-authorship networks are useful for predicting future co-author connections among current writers [3, 5, 6]. In essence, plausible proposals for possible research partnerships may be made based on reliable forecasts of new ties between two existing authors in co-authorship networks.

The H-Index is the predominant statistic used to evaluate individual researchers' influence [7]. This measure considers the number of publications an author has produced and the number of citations those papers have received. However, there is a possibility that the H-Index is not designed to provide the most precise assessment possible of a researcher's actual influence. This restriction derives from the fact that it has the propensity to misrepresent an author's overall visibility, which may be caused by variables such as the author's habit of self-citation or their participation in research activities across a variety of academic fields [8, 9]. Consequently, there is a compelling need for an alternative impact measure that may offset the adverse effects of these disadvantages and appraise the academic contributions that are more thorough.

This research uses supervised models to establish the appropriate weights associated with different topological characteristics for predicting co-author connections via extracting structural and topological features from co-authorship networks. The suggested approaches were tested on co-authorship networks in computer science, with results confirming the topological dependency of co-author relationship development. In addition, supervised learning approaches were shown to be effective in using this correlation to make reliable co-authorship predictions.

2. Related works

Researchers have looked at several options to improve the accuracy of collaboration suggestions. Using suggestions based on shared interests in research is a common tactic in this field. Extracting authors' research interests from titles, abstracts, and keywords is a common goal of text-mining approaches, which are at the heart of these inquiries. Text-matching methods are also used to recommend researchers with similar areas of expertise.

In recent years, exciting developments in geometric deep learning have led to graph neural networks combining the best features of fully connected and convolutional neural network designs. Many of these methods draw on the idea of neighborhood information aggregation from graph convolution networks (GCNs) [10]. They are improved by incorporating well-known deep learning architectures like Recurrent neural networks (RNN) [11], Approximate Message Passing (AMP) [12], A generative adversarial network (GAN) [13], and graph transformers [14].

Abbasi et al. [15] developed a theoretical framework based on social network theories and analytical tools to study scholarly cooperation networks, particularly co-authorship networks. They aimed to evaluate the impact of social networks on scholarly performance in the field of information systems as measured by citations using social network analysis (SNA) metrics such as normalized degree centrality, normalized closeness centrality, normalized betweenness centrality, normalized eigenvector centrality, average tie strength, and efficiency. A Poisson regression model was used to find positive correlations between the g-index and four of the seven SNA metrics except for normalized betweenness and closeness centrality.

Under Italian law, Behrouz et al. [16] comprehensively analyzed three "academic disciplines"—computer engineering, mathematics, and economics. Because they span so many themes and interests, researchers chose these areas. They collected academic scholar data from Elsevier's Scopus public database. Then, authorship networks were created. Each network's topology and community were examined, and comparative analysis was used to explore the differences and similarities across disciplines of study.

Kumar et al. [17] developed a flexible link prediction technique using several node centralities and machine learning classifiers. They employed classic and novel node centrality metrics to better capture the network's local, quasi-local, and global structural features. These node centrality values become feature labels for network nodes. Negative samples showed non-existent edges, whereas positive ones indicated existing ones. These labeled features and edge endpoints were combined for a link prediction dataset. The dataset was tested using many machine learning classifiers.

Nasiri, Elahe, et al. [18] improved local random walks by proposing a technique that guides the random walk to more impactful nodes. This method selects the next node based on its influence. Researchers assessed asymmetric mutual impact using mutual information. The approach was compared to several local, quasi-local, and global similarity-based algorithms.

Nasiri et al. [19] developed Robust Graph Regularization Nonnegative Matrix Factorization for Attributed Networks (RGNMF-AN) to solve the direct link prediction problem in attributed networks. This model effectively includes network topology

and node attribute data. The SARWS scoring matrix measures high-order proximities between nodes. This scoring matrix may indicate structural and attributed properties in high-order proximity to better capture attribute information. It combines the SARWS score matrix with topological and attribute information via graph regularisation to better aggregate meaningful attribute information within high-order proximities.

Hasin et al. [20] investigated link prediction (LP) in Academic Social Networks (ASNs) to forecast future scholar collaborations. This research compares the main taxonomies of topological, content, and hybrid approaches. These approaches provide similarity ratings for every pair of unconnected nodes in ASNs to solve the LP issue.

3. Graph theory

The branch of mathematics studies graphs, which are mathematical structures used to represent relationships between entities. A graph comprises a set of nodes, or vertices, and a collection of lines, or edges, that connect the nodes. Social networks, computer networks, transportation systems, and molecular structures are just a few examples of phenomena that may be modeled using vertices and edges [21, 23].

It provides a framework for analyzing the properties of graphs, developing methods for dealing with graph-related problems, and more. Moreover, graph theory aids in resolving issues associated with graph structures. Connectivity, paths, cycles, distances, and graph colors are just topics it explores. Graph theory also includes many other kinds of graphs, including directed and undirected graphs, weighted graphs, and bipartite graphs [21, 24, 25].

Many fields, from computer science and operations research to physics, biology, and sociology, use concepts from graph theory [24, 25]. Graph theory's versatility has led to its use in various contexts, including studying data dependencies, modeling complex systems, appreciating the interconnection of things, and solving optimization issues (Fig. 1).

3.1. Common Neighbor Classifier (CNC)

A machine learning method that predicts linkages between social networks and graph-based data structures. This approach evaluates network connectivity by measuring how frequently two nodes have familiar neighbors. This method assumes that linked nodes have more overlapping neighbors than disconnected ones [27, 28]. The equation for this behavior is:

$$CNC(x\ y) = |\Gamma(x)| \cap \left|\Gamma(y)\right| \quad (1)$$

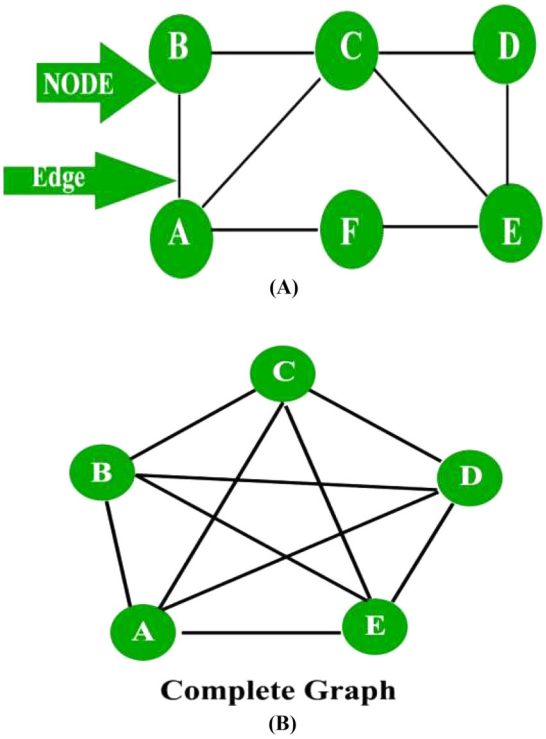

Fig. 1. Displays plots (A) and (B); it exemplifies a mathematical form of graph theory [26].

Where;

CNC (x,y): The Common Neighbor Classifier score equals the number of common neighbors between x. and y.

Γ(x) and Γ(y): These denote the neighbors of nodes x and y, respectively. The neighborhood Γ(x) includes all nodes directly connected to node x.

3.2. Jaccard Coefficient (JC)

This scale evaluates the degree of similarity between two data sets. Its use spans various machine learning subfields, including clustering, classification, and recommendation systems [29]. It is the ratio of the intersection of two groups to the union of these two groups used to obtain the Jaccard coefficient.

$$JC(x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|} \quad (2)$$

Where:

Γ(x): The set of neighbors of node x.

Γ(y): The set of neighbors of node y.

Γ(x) ∩Γ(y): The intersection of the sets Γ(x) and Γ(y), i.e., the set of common neighbors between x and y.

Γ(x) ∪Γ(y): The union of the set Γ(x) and Γ(y), i.e., the set of all distinct neighbors of x and y combined.

3.3. Preferential Attachment Coefficient (PAC)

It causes higher-degree nodes to link more. The PAC quantifies preferential attachment [30, 31]. The PAC is the ratio of the square of the total of all network degrees to the product of any two node degrees, normalized by a constant factor.

$$\mathrm{P\,A}\left(\mathrm{x},\mathrm{y}\right) = \left|\Gamma\left(\mathrm{x}\right)\right| * \left|\Gamma\left(\mathrm{y}\right)\right| \quad (3)$$

3.4. Adamic Adar Coefficient (AAC)

Link Prediction (LP) activities in social networks and other graph-based data structures often employ the Jaccard similarity coefficient [32]. This technique assumes that neighbors of different nodes will connect more often. The following equation calculates Jaccard similarity:

$$\mathrm{AA}\left(\mathrm{x},\ \mathrm{y}\right) = \sum\nolimits_{\mathrm{z}\in\Gamma(\mathrm{x})\cap\Gamma(\mathrm{y})} \left(a_n \cos \frac{1}{\mathrm{Log}\left(\left|\Gamma\left(\mathrm{z}\right)\right|\right)} \right) \quad (4)$$

Where;

| Γ(z)|: The degree of node z, i.e., the number of neighbors of z.

log(| Γ(z)|)1: The contribution of the common neighbor z to the similarity between x and y. The logarithm dampens the contribution of highly connected neighbors, making the coefficient more sensitive to less common neighbors.

4. Methodology

4.1. Dataset

The dataset employed in this investigation was systematically acquired from scholarly journals that predominantly disseminate articles within the realm of physics. Collecting data entailed a meticulous approach to acquiring and consolidating articles from the journals above. A methodical methodology was subsequently utilized to classify and organize the gathered articles, culminating in the establishing of a unique dataset. In this context, Each node in the collection represents an individual group of authors who have worked on scientific works.

The edges, on the other hand, serve as a representation of the collaborative connections that exist between these authors. The intricate web of collaborative efforts among authors is comprehensively documented by implementing this dataset structure. It can be systematically examined, resulting in significant revelations regarding the nature and trends of scientific collaboration across various fields. 1050 distinct nodes made up the dataset that was used for this investigation. As a result of running the code, it was discovered that 237 of the nodes included in the dataset were incorporated into the database. The database consists of three columns: one for the titles of the papers, another for the people participating in each publication, and a third column for the specified code numbers linked with the execution of the code. Each of these columns is labeled with the appropriate heading.

Table 1. Splitting the data into test and training sets.

Data count	Training set	Testing set	Validation set
1050	788	210	52

It is recommended to divide the data set into three sets: a training set, a test set, and a validation set. This will ensure the effective implementation of training and model evaluation. It is recommended that the training set contain 75% of the total data, the test set 20%, and 5% for validation (to maintain the 1/3 ratio). The model's performance is evaluated with the help of the test set. In contrast, the training set is primarily used to improve it (Table 1).

Various approaches may be taken to successfully prevent overfitting, which happens when a model overly adapts to the data used for training and cannot generalize well. One method includes separating the dataset into training and test sets. In order to conduct an accurate analysis of the model's performance, these data sets must have patterns comparable to those seen in the actual world. In addition, using a validation set is vital for comparing many models and identifying which performs better, independent of how well the models have performed individually.

4.2. Implementation

Support vector machine (SVM) is a computer technique that uses an example-based learning approach. Forecasting labels for input feature vectors or datasets may be used as a reliable approach to building classifiers by defining a decision border between two classes [33, 34].

In order to do the calculations, it is essential to use techniques that have been optimized to deal with the massive size of the graphs involved. Making sure these procedures work as intended and use resources efficiently is crucial. The developer would take a lot of time and effort to create such techniques from scratch. To reduce these expenses, it will use free third-party libraries. After an exhaustive investigation, it was determined that NetworkX and the Pandas Python Data Analysis Library were the top two options. Python's NetworkX module

provides comprehensive capabilities for network research. At the same time, the Pandas data analysis toolbox is a popular open-source resource. Google's NetworkX provides extensive tools for doing network analysis.

The file containing the code may be split into three main sections. The first section is dominated by the time required to read the input datasets and produce the corresponding graphics. This approach uses the functions provided by the NetworkX library. It also shows how input data is gathered and how graphs are split into test and training sets. This partition enables the train and test graphs to be handled separately.

To create the massive component, it is necessary first to generate the graphs and then eliminate any pairs of nodes that are geographically far from one another. It is crucial to double-check and ensure that the sets of nodes in the train and test graphs are the same. Since the main objective of the studies is to evaluate the effectiveness of prediction algorithms in detecting new connections between existing nodes in the network, maintaining this consistency is extremely important. Hence, the requirement can be satisfied by eliminating nodes in only one of the graphs.

To evaluate the predictions generated by the typical neighbourhood technique, it is necessary to calculate the number of familiar neighbors for each potential pair of nodes inside the network. One can acquire the capacity to make predictions by utilizing this formula. In order to achieve this objective, a framework of nested loops has been devised to generate all possible pairs.

The total number of potential node pairings in a graph with n nodes can be calculated by applying a formula that selects all subsets of the node-set, where each subset consists of exactly two components. This will provide the total count of possible combinations of nodes. The calculation of the number of possible node pairs is simplified by employing this approach, resulting in a significant increase in the number of potential combinations, rising exponentially. Consequently, it is impossible to produce and analyze all pairs simultaneously. The formation and processing of node pairs are fragmented to solve this difficulty.

The number of combinations for undirected networks can be calculated using the formula n(n-1)/2, where n is the total number of nodes in the network. Given this information, the sequence of the nodes is no longer relevant (Fig. 2).

The subsequent stage quantifies the number of familiar neighbors each pair shares inside the generated chunk. This is completed once the chunk has been formed. Subsequently, the calculated values are compared to the most optimal values obtained thus far.

```
node_A_degree = [G.degree(edge[0]) for edge in edges]
#print(node_A_degree)
node_B_degree = [G.degree(edge[1]) for edge in edges]
#print(node_B_degree)


common_neighbors = [sum(nx.common_neighbors(G,edge[0],edge[1])) for edge in edges]
#print(common_neighbors)


# Prediction using common_neighbors
pred_cn = list(nx.common_neighbor_centrality(G))
score_cn, label_cn = zip(*[(s, (u, v) in edges) for (u, v, s) in pred_cn])
```

Fig. 2. The screenshot shows the calculation of the quantity of renowned adjacent individuals.

From the combined list of values, the applicants who have shown the most promise are chosen to go on, while the values that haven't shown much promise are thrown out. As part of the evaluated strategy for forecasting, this approach ensures that only applicants with the highest probability of success are considered. Also known as pairs, "It," are stored in memory as part of the technique.

Consequently, memory use may be improved, which will help alleviate problems that arise from having inadequate memory. Furthermore, with the completion of each step, a more significant fraction of pairings can be assessed. The findings indicate that the candidates with the highest likelihood of success are determined by the total amount of graph data that has been analyzed.

4.3. Model testing

The suggested models are put through extensive testing, which consists of entering test data, which is then preprocessed and fed into the models while the testing phase is in progress. After that, the test results are examined to determine the pertinent characteristics that may be used for the LP (Learning Process). The system will provide an output that validates the existence of LP based on the knowledge gathered via the learning process if the preset conditions are fulfilled. On the other hand, the system will generate an LP assertion if the conditions are not satisfied. The validity of the output model is tested using two different methods: (1) checking its alignment with the labels included in our dataset, and (2) making sure that a data portability technique reduces the amount of data that is lost when it is applied to the data gathering process.

Table 2. Performance metrics.

Effectiveness measure	Relation
Accuracy	$= \frac{TP+TN}{TP+TN+FP+FN}$
Recall	$= \frac{TP}{TP+FN}$
Precision	$= \frac{TP}{TP+FP}$
$F1-score$	$= \frac{2TP}{2TP+FP+FN}$

5. Assessment

During the stage that is dedicated to the first data preparation, the data that is being entered is converted into a graph that is a representation of the network. Following that, in the "method execution" step, The created graph is inputted into the implementation codes of the procedure, and these programs undergo thorough testing. During the final phases, evaluation metrics obtained before to, during, and after the execution of the method are retained as a crucial component of the comprehensive outcome report. This step takes place after all other stages have been completed.

The approach incorporates several different components, This includes techniques for predicting future outcomes, organising the appearance of data, creating visual representations, and saving output files.

Processing graphs that are produced from real-world datasets offers considerable hurdles as a result of the enormous quantity of information that is stored inside these datasets. These graphs may have a wide variety of nodes and edges, which can sometimes number in the thousands and form complicated relationships between the nodes and edges.

5.1. Performance metrics

In order to guarantee the accuracy that our model is capable of delivering, we have used four different metrics that reflect Recall, F1-Score, and Precision. The following are the formulas of the connections between each of these measures (Table 2):

6. Results

This article describes and analyzes the results of the training process that was used in the suggested model. The training process made use of a dataset that was constituted of scientific knowledge about physics. In the wake of a single cycle of training that lasted for 25 epochs, it will now continue to explain the results obtained from the relevant experimental attempts (Tables 3 and 4).

Table 3. Using SVM classifier to classify the physics science dataset.

Dataset	Algorithm	Type of link	Accuracy
Physics science	SVM	LP	96.66 %

Table 4. Performance metrics for SVM classifier to the physics science dataset.

Type of link	Precision	Recall	F1-score
LP	100 %	92.47%	96.08%

Fig. 3 demonstrates the use of the confusion matrix method, which consists of four separate metrics: true positive (TP), false positive (FP), false negative (FN), and true negative (TN) values. These metrics are allocated to corresponding indices. We can arrive at the definitive categorization results by solving

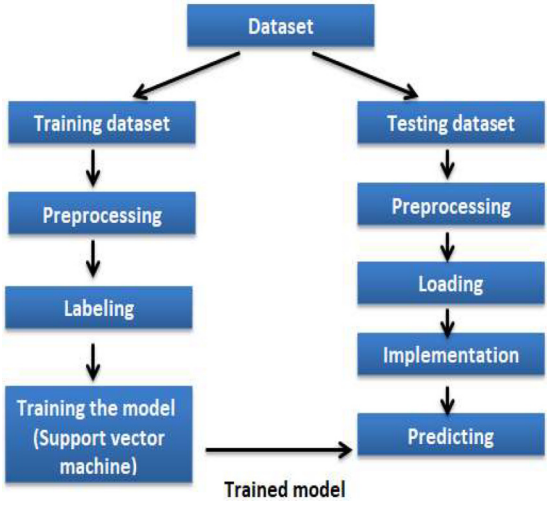

Fig. 3. Flowchart illustrating the process of testing and training in a model.

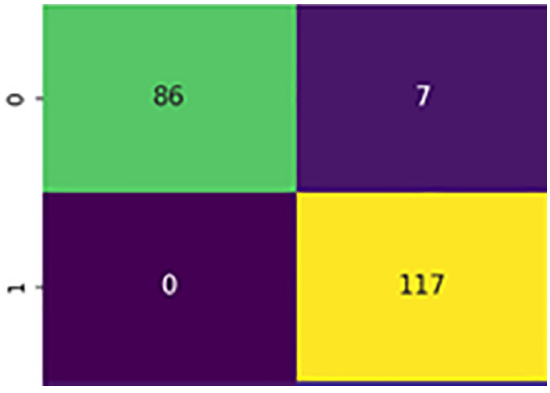

Fig. 4. Execution of the confusion matrix.

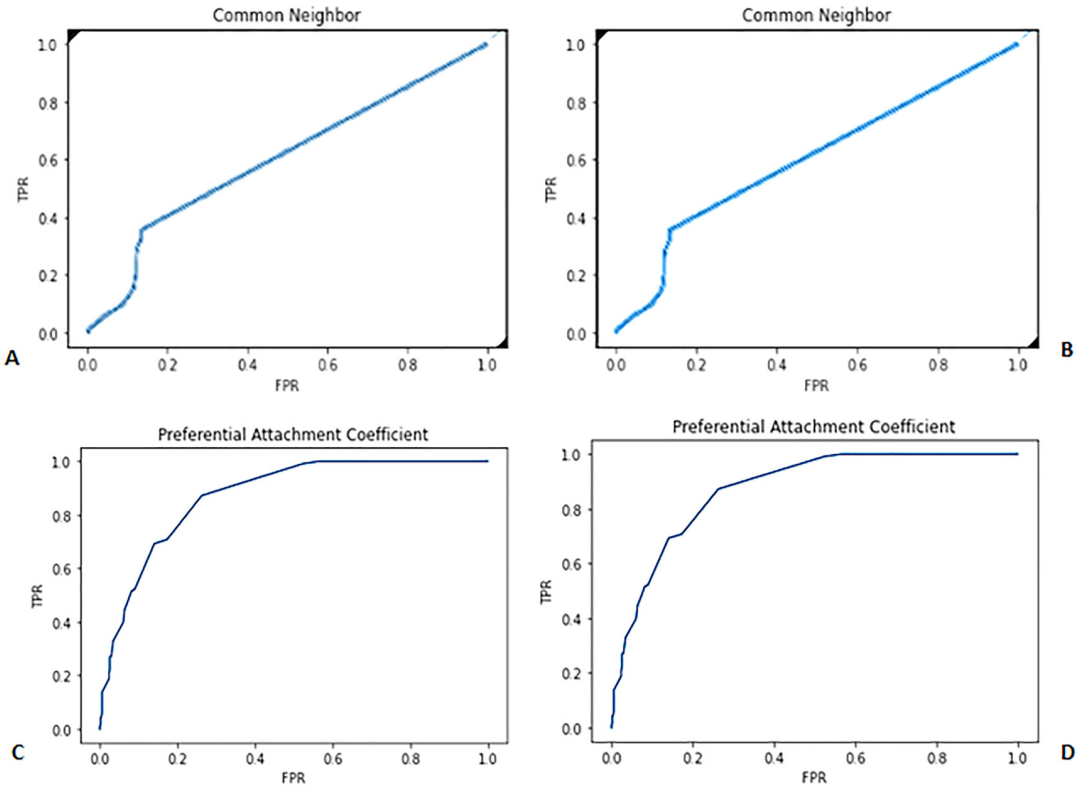

Fig. 5. Illustration of the representation of node affinity metrics in a social dataset. (A); CNC (B); AAC (C); JC (d) PAC.

the equations above. The dataset's results are shown graphically using four plots: the confusion matrix, the training test accuracy curve, the loss accuracy curve, F1- score, etc.

7. Discussion

Metrics that measure performance provide insightful data on the efficiency of various approaches to data in physics research. The Preferential Attachment Coefficient has shown much better performance than the other three ways when compared to the strategies that have been investigated. On the other hand, compared with the other two methods in terms of performance, the Jaccard Coefficient and Jaccard Coefficient methodology come out on top. Notable about these approaches is that they use the Jaccard Coefficient somehow (Fig. 5). Because these research networks are non-targeting platforms, the offered information may be accessed by a broad audience rather than customized for specific individuals. This feature suggests that the content that is provided is open to discussion.

Using the Preferential Attachment Coefficient methodology, node pairings with greater normalized values demonstrate a better similarity index. This indicates that these node pairs are more likely to be picked when making predictions. The fact that this was seen gives rise to the conclusion that the paired nodes in question have a strong link to one another and are more likely to have qualities or interests in common with one another. However, it is essential to remember that a more significant percentage of presently shared neighbors among the total population of neighbors in a particular region may also indicate a feeling of satisfaction or saturation in that area. This is something that has to be taken into consideration. This suggests that a person may have explored their interests at length, limiting the possibility of frequent contributions on the same themes covered in detail earlier.

The empirical findings of the research reveal that, in comparison to other methodologies, there is a greater alignment between user behavior and the ideas that underlie the Preferential Attachment Coefficient (PAC) approach. Based on this finding, users seem interested in various subject areas. On the other hand, examining the performance of these two methods using the core dataset of European email finds no substantial gap between them. After carefully reviewing the data, this finding presents unmistakable evidence. This conclusion could result from a combination of many different causes, each contributing to the total effect.

The results obtained from applying the Support Vector Machine (SVM) model to predict co-authorship networks within the field of physics provide significant insights into the efficacy of machine learning techniques in complex network analysis.

The SVM model achieved a high accuracy rate of 96.66%, highlighting its robust predictive capability in identifying potential links between authors and publications. This level of accuracy suggests that the SVM model is well-suited to link prediction in scholarly networks, where identifying hidden connections can be critical for advancing knowledge and fostering collaboration.

One of the key strengths of the SVM model lies in its ability to handle the high-dimensional data typical of co-authorship networks effectively. These networks are characterized by many nodes (representing authors) and edges (representing collaborations), making traditional statistical methods less effective. With its capacity for handling non-linear relationships and finding an optimal hyperplane that separates different classes, the SVM model proves to be an effective tool in this context. This is further evidenced by the model's performance across various metrics, including precision, recall, and F1-score, demonstrating the model's balance between identifying positive links and minimizing false positives and negatives.

The precision of 100% indicates that every link predicted by the model as a potential co-authorship was indeed correct, reflecting the model's high specificity. However, the model's 92.47% recall rate suggests that it successfully detected almost all actual co-authorship links but failed to spot a small proportion of them, implying that there is still room for capturing all possible co-authorships better.

This trade-off between precision and recall is common in machine learning, especially in complex network analysis where nodes are not always linear or easily discernible.

The F1-score of 96.08% combines precision and recall to assess the model's performance objectively. This measure is particularly relevant in terms of link prediction because it considers how accurately the predictions of the model were made and its ability to capture all relevant links. The high F1 score thus underscores the effectiveness of the SVM model in forecasting co-authorship ties, thereby serving as a valuable tool for exploring hidden collaboration patterns within academic networks.

Further supporting this conclusion is using a confusion matrix to evaluate performance. When looking at true positives, false positives, false negatives, and true negatives, one can see where things went right and wrong for the model.

The SVM model's accuracy and reliability are underlined by a large number of true positives and a low number of both false negatives and positives.

This study's success in SVM models also emphasizes the importance of careful data preparation and proper evaluation metrics. The dataset used for this study was obtained by meticulous curation from scientific publications in physics to ensure that it closely represented the real co-author network being studied. By carefully training the model on each part of the dataset, including training, testing, and validation sets, the researchers could test whether their findings could be generalized beyond their sample without risking overfitting.

This model's performance was also determined using various evaluation measures such as accuracy, precision-recall, and F1 score. One metric may not be applicable for capturing all nuances of data or showing how well the developed model works in complex network analysis, thus requiring a diverse approach to evaluation. Through multiple measures considered by the researcher regarding its strengths and limitations, a fuller understanding of how useful it is in solving this problem can be reached based on these results.

Table 5 shows how different classifiers performed when employed for link prediction (LP) within the physics science dataset. The SVM classifier has a

Table 5. Comparison of the results with the other related studies.

		Results			
Study	Model	Accuracy	Precision	Recall	F1-score
[35]	Logistic Regression	NA	69.6%	67.7%	67.1%
	SVM	NA	69.7%	67.8%	67.1%
[36]	Gradient boosting machines	86.68%	NA	86.95%	NA
	Random Forest	84.06%	NA	84.18%	NA
[37]	Random Forest	NA	20.2%	83.4%	91.7%
	k-nearest neighbors	NA	14.8%	82.2%	90%
[38]	ANN	NA	91.3%	97.5%	94.3%
[39]	CNN	59.19%	72.27%	53%	61.05%
This study	SVM	96.66%	100%	92.47%	96.08%

striking accuracy of 96.66%, which is remarkable compared to other models designed for similar tasks. However, a highly accurate model still predicts wrong links, leading to false positives.

The recall rate is slightly lower at 92.47% because the truth is that no model can perfectly predict all links, and this one missed out on some. The F1-score of the SVM classifier amounts to 96.08%, which combines precision with recall as a measure of the model's performance over data fitting and prediction in terms of both true positives and negatives, making it very reliable for co-authorship networks in link prediction.

On the other hand, compared to other classifiers, most predictive models often have less precision and recall, thereby preventing an equilibrium between these measures. Consequently, this may increase the number of false positives or negatives, thus affecting predictions in complex networks and influencing their accuracy and reliability most seriously. The superior performance of the SVM classifier in all evaluated metrics, as depicted in Table 5, confirms its effectiveness and robustness in handling the complexities of the co-authorship dataset in physics science.

8. Conclusion

All the methods evaluated here use node-centric data to make predictions, making them suitable for classification as topology-based techniques. Machine learning (ML) and other cutting-edge methods that use path-based data or random traversal approaches may be used to improve speed. A more profound comprehension of the features that contribute to the success of link prediction (LP) techniques would result from expanding the scope of this research to include the classes above methodologies. This enlargement may help practitioners get a better grasp of the material.

For future work: Extending the study by including more extensive datasets and training more classifiers and algorithms inside an ML framework is proposed for future research. This elaboration would allow the study's results to be used in R&D settings, raising the study's practical relevance.

9. Limitation and constraints

The limits of the used model are evident in densely populated and diverse areas of the graph. One way to alleviate these restrictions is by utilizing a taxonomy of the publications' subjects or explicitly modeling links between topics.

Alternatively, we can consider the diversity within a neighborhood by creating a model that measures the impact of one node on the prediction of another node. This can be based on factors such as the overlap between the neighborhoods of the two nodes, the number of shared references between them, or the proportion of links a predicting node has within the neighborhood of the predicted node.

Alternatively, a node's impact on a prediction can be determined by its position or structural characteristics in the co-authorship graph. Social network analysis provides a range of node centrality measures that can be used for this purpose.

10. Ethical considerations

For ethical purposes, We thus confirm that we have completed the current study topic.

Acknowledgments

We thank our colleagues at Al Ain University, National University of Science and Technology, Dhi Qar, Iraq, and Gazi University/Turkey in the Department of Computer Engineering for their great support in completing this study.

References

1. H. Dong et al., "On the equivalence of decoupled graph convolution network and label propagation," *Proceedings of the Web Conference*, 2021.
2. L. Espín-Noboa, T. Peixoto, and F. Karimi, "Social network modeling and applications, a tutorial," *arXiv preprint arXiv:2306.11004*, 2023.
3. M. S. Lyra, F. L. Pinheiro, and F. Bacao, "Public procurement fraud detection: a review using network analysis." *Complex Networks & Their Applications X: Volume 1, Proceedings of the Tenth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2021 10*. Springer International Publishing, 2022.
4. de Bruin, G. Jan et al., "Experimental evaluation of train and test split strategies in link prediction," Complex Networks & Their Applications IX: Volume 2, Proceedings of the Ninth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2020. Springer International Publishing, 2021.
5. A. Ghasemian et al., "Stacking models for nearly optimal link prediction in complex networks," *Proceedings of the National Academy of Sciences*, vol. 117, no. 38, pp. 23393–23400, 2020.
6. R. Interdonato et al., "Feature-rich networks: going beyond complex network topologies," *Applied Network Science*, vol. 4, pp. 1–13, 2019.
7. I. Makarov et al., "Dual network embedding for representing research interests in the link prediction problem on co-authorship networks," *PeerJ Computer Science*, vol. 5, e172, 2019.
8. R. Molontay, and M. Nagy, "Two decades of network science: as seen through the co-authorship network of network scientists," *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, 2019.

9. H. A. Hasin and D. Hassan, “Link prediction in co-authorship networks,” *Science Journal of University of Zakho*, vol. 10, no. 4, pp. 235–257, 2022.
10. T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907* (2016). Homophily-Evidence from IS senior Scholar’s Basket of Eight Journals for Business Analytics Research.” *AMCIS*, 2020.
11. J. You et al., “Graphrnn: generating realistic graphs with deep auto-regressive models,” *International conference on machine learning*. PMLR, 2018.
12. V. Petar et al., “Graph attention networks,” *stat*, vol. 1050, no. 20, pp. 10–48550, 2017.
13. M. Ding, J. Tang, and J. Zhang, “Semi-supervised learning on graphs with generative adversarial nets,” *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018.
14. J. Park et al., “Meta-node: a concise approach to effectively learn complex relationships in heterogeneous graphs,” *arXiv preprint arXiv:2210.14480*, 2022.
15. A. J. A. Abbasi and L. Hossain, “Identifying the effects of co-authorship networks on the performance of scholars: a correlation and regression analysis of performance measures and social network analysis measures,” *Journal of Informetrics*, vol. 5, no. 4, pp. 594–607, 2011.
16. A. Behrouz and M. Seltzer, “Anomaly detection in multiplex dynamic networks: from blockchain security to brain disease prediction,” *arXiv preprint arXiv:2211.08378*, 2022.
17. S. Kumar, A. Mallik, and B. S. Panda, “Link prediction in complex networks using node centrality and light gradient boosting machine,” *World Wide Web*, vol. 25, no. 6, pp. 2487–2513, 2022.
18. E. Nasiri et al., “Impact of centrality measures on the common neighbors in link prediction for multiplex networks,” *Big Data*, vol. 10, no. 2, pp. 138–150, 2022.
19. E. Nasiri, K. Berahmand, and Y. Li, “Robust graph regularization nonnegative matrix factorization for link prediction in attributed networks,” *Multimedia Tools and Applications*, vol. 82, no. 3, pp. 3745–3768, 2023.
20. H. A. Hasin and D. Hassan, “Link prediction in co-authorship networks,” *Science Journal of University of Zakho*, vol. 10, no. 4, pp. 235–257, 2022.
21. S. M. Kostić, M. I. Simić, and M. V. Kostić, “Social network analysis and churn prediction in telecommunications using graph theory,” *Entropy*, vol. 22, no. 7, p. 753, 2020.
22. A. Majeed and I. Rauf, “Graph theory: a comprehensive survey about graph theory applications in computer science and social networks,” *Inventions*, vol. 5, no. 1, p. 10, 2020.
23. D. Goldenberg, “Social network analysis: from graph theory to applications with python,” *arXiv preprint arXiv:2102.10014*, 2021.
24. B. Sanchez-Lengeling et al., “A gentle introduction to graph neural networks,” *Distill*, vol. 6, no. 9, e33, 2021.
25. M. Li et al., “Percolation on complex networks: theory and application,” *Physics Reports*, vol. 907, pp. 1–68, 2021.
26. R. Diestel, *Graph theory*, Springer (print edition); Reinhard Diestel (eBooks), 2024.
27. H. Wang and Z. Le, “Seven-layer model in complex networks link prediction: a survey,” *Sensors*, vol. 20, no. 22, p. 6560, 2020.
28. M. Haddad et al., “Temporalnode2vec: temporal node embedding in temporal networks,” Complex Networks and Their Applications VIII: Volume 1 Proceedings of the Eighth International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2019 8. Springer International Publishing, 2020.
29. L. da F. Costa, “Further generalizations of the Jaccard index,” *arXiv preprint arXiv:2110.09619*, 2021.
30. L. Iskhakov et al., “Clustering properties of spatial preferential attachment model,” *Algorithms and Models for the Web Graph: 15th International Workshop, WAW 2018, Moscow, Russia, May 17–18, 2018, Proceedings 15*, Springer International Publishing, 2018.
31. S. Sidorov et al., “Temporal behavior of local characteristics in complex networks with preferential attachment-based growth,” *Symmetry*, vol. 13, no. 9, p. 1567, 2021.
32. M. Jia, B. Gabrys, and K. Musial. “Measuring quadrangle formation in complex networks,” *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 2, pp. 538–551, 2021.
33. O. A. Alimi, K. Ouahada, and A. M. Abu-Mahfouz, “A review of machine learning approaches to power system security and stability,” *IEEE Access*, vol. 8, pp. 113512–113531, 2020.
34. M. Tanveer et al., “Comprehensive review on twin support vector machines,” *Annals of Operations Research*, pp. 1–46, 2022.
35. Q. Yu et al., “Predicting co-author relationship in medical co-authorship networks,” *PloS one*, vol. 9, no. 7, e101214, 2014.
36. G. Resce, A. Zinilli, and G. Cerulli, “Machine learning prediction of academic collaboration networks,” *Scientific Reports*, vol. 12, no. 1, p. 21993, 2022.
37. D. Hassan, “Supervised link prediction in co-authorship networks based on research performance and similarity of research interests and affiliations,” *2019 International Conference On Machine Learning And Cybernetics (ICMLC)*, IEEE, 2019.
38. Z. Meng, “Link prediction using machine learning algorithms.” *International Research Journal of Engineering and Technology*, 2020.
39. V. D. Quang et al., “An improved adaboost algorithm for highly imbalanced datasets in the co-authorship recommendation problem,” *IEEE Access*, 2023.