RESEARCH ARTICLE

Transforming Citation Networks into Insights: Mapping Scholarly Influence with Advanced Graph Models

Tahir Sher[1], **Abdul Rehman**[2,*] and **Imran Ihsan**[3]

[1] Department of Artificial Intelligence, Korea University, Seoul 02842, Republic of Korea
[2] Convergence Institute of Human Data Technology, Jeonju University, Jeonju 55069, Republic of Korea
[3] Department of Creative Technologies, Air University, Islamabad 44000, Pakistan

Abstract

The growing role of citation relations in identifying research impact has spurred much investigation on assessing the most cited papers and their roles within datasets. Due to the richness of the CORA dataset, this study selects highly cited papers and measures the results of node classification, as well as the H-index of research articles. Besides, it explores the correlations and robustness with regard to the nodes by computing their chances and studying their connections. To these ends, linear transformation was utilized for mapping low-level node features to high-level, and the Graph Attention Networks (GAT) for node classification. The study was able to find highly cited papers and compute their H-index, which gives insight into the citation patterns in the CORA dataset. For instance, Paper ID 12182 reported an H-index of 20, while a high citation paper index of 35 received 166 citations. On the test dataset, the study achieved a node classification accuracy ranging from 78.6% to 82% and an F1 score of 78.14%. Furthermore, 7 nodes of the machine learning domain have also been distinguished and categorized according to their features and their relations within the graph. In the citation network identified, the present research detailed the citation interconnection that characterized the works within the dataset. Our research mainly focuses on new tasks such as the extraction of highly cited papers and the calculation of the H index that improve the comprehension of the scholarly influence and the citation relation for future development strategies in citation network analysis.

Keywords: deep learning, text mining, graph attention networks, citation, h-index.

1 Introduction

Graphs are widely used to represent complex relationships in various fields like proteomics [1], social networks [2], fraud detection [3, 4], traffic systems [5], and computer vision [6]. They capture both data and structure, from simple node-edge structures to more intricate types like trees, cyclic graphs, and directed acyclic graphs (DAGs). Applications generally fall into two categories: graph-focused, where models analyze the

Submitted: 14 May 2025
Accepted: 09 July 2025
Published: 04 October 2025

Vol. 1, **No.** 4, 2025.
10.62762/TACS.2025.939169

***Corresponding author:**
Abdul Rehman
a.rehman.jj@jj.ac.kr

Citation
Sher, T., Rehman, A., & Ihsan, I. (2025). Transforming Citation Networks into Insights: Mapping Scholarly Influence with Advanced Graph Models. *ICCK Transactions on Advanced Computing and Systems*, 1(4), 238–257.

graph as a whole [7], and node-focused, which target individual nodes. Advances like random walk models and RNNs have paved the way for Graph Neural Networks (GNNs), which handle both types without heavy pre-processing.

A graph consists of nodes and edges [8, 9]. With the rise of machine and deep learning, graphs like knowledge graphs have gained importance in fields such as the social and natural sciences, and protein interaction networks [10–13]. Ramsay's theorem explains that any large enough graph must contain either a fully connected subset or a group of independent nodes, with precise conditions on the number of vertices [14]. Graphs come in various forms—hypergraphs, fuzzy graphs, and their directed or undirected variants. These are further categorized into cyclic and acyclic types, as shown in Figure 1(c) and Figure 1(d) [15–18]. Figure 1(b) illustrates how directed and undirected graphs can also be modeled as fuzzy cognitive maps (FCMs) [19].

Researchers represented undirected graphs (UDGs) as symmetric fuzzy cognitive maps (FCMs) using unweighted edges (value one), as shown in Figure 1(c), with labeled nodes in Figure 1(d). The symmetric adjacency matrix confirms that UDGs fall within both UDG and FCM sets [20]. Graphs are useful for meta-analysis, helping extract knowledge efficiently [21]. While knowledge graphs (KGs) are widely used for information retrieval, they often lack completeness. Graph Neural Networks (GNNs) help capture the structural patterns in these KGs [22].

Artificial Neural Networks (ANNs), a deep learning method inspired by the human brain, consist of input, hidden, and output layers. Each node passes values with weights and thresholds, as illustrated in Figure 2. Graph Convolutional Networks (GCNs), an extension of CNNs, use graph structures to aggregate information from neighboring nodes. Known for their expressive power, GCNs have shown strong performance across various tasks. Recent developments include graph auto-encoders [23, 24], generative models [25, 26], attention models [27, 28], and graph recurrent networks [21, 24, 34]. GNNs operate through information diffusion, where each node (unit) updates its state based on graph connectivity until reaching stability. These models apply to both positional and non-positional graphs [35, 36], and have been widely used for node classification [92, 93, 99–101].

In our work, we adopt Graph Attention Networks (GAN), which prioritize neighboring nodes based on learned attention scores [37, 38]. Unlike fixed-weight approaches like GCN [33] or GraphSAGE [39], GAN dynamically adjusts weights through attention mechanisms, as formalized in Eq. 1.

$$\beta_{nu}^{(k)} = Softmax\left(g\left(\vec{a}^{T}\left[W^{(k)}H_n^{(k-1)} \,||\, W^k H_u^{(k-1)}\right]\right)\right) \tag{1}$$

where g(·) is a LeakyReLU activation function, and *'a'* represents the vector for learnable parameters. The softmax function ensures that the attention weights sum up to one for all neighbors of the node 'n.' Additionally, GAN employs a multi-head attention mechanism to improve performance on node classification tasks compared to GraphSage. While GAN assumes that all attention heads contribute equally, Gated Attention Networks (GAAN) [40] introduce a new self-attention mechanism that assigns different attention scores to each attention head. Furthermore, GeniePath [41] introduces a new Gating mechanism, Long Short Term Memory (LSTM), to observe and control the flow of information across graph convolutional layers. Figure 3(a) GAN [38] considers the attention coefficient β_{ij} and gives more weight to important nodes. Figure 3(b) GCN [33] gives a non-parametric weight $\beta_{\mathrm{ij}} = \frac{1}{\sqrt{deg(n_{\mathrm{i}})deg(n_{\mathrm{j}})}}$ to the neighbor n_j of n_i during the aggregation process. Where $H_n^0 = y_n$, the attention weight considers the strength between the node "n" and its neighbor "u." Multi-task learning using GANs has been widely explored across text, image, and video data. Since most data—about 98%—is raw and unstructured, organizing it meaningfully is essential for effective research. Only a small portion (around 2%) carries valuable knowledge. To extract this, various data mining techniques are applied, including pattern discovery, clustering, classification, text mining, and social network analysis [51, 52, 83–87, 94–98].

Given its strong performance, we focus on applying GANs for text mining, particularly due to the challenges of unstructured text data [43, 45, 53, 79]. Key applications include information extraction, knowledge discovery, and text data mining. However, high-dimensional and sparse text data—where documents represent only a small subset of a vast vocabulary—requires careful preprocessing (as shown in Table 1 and Figure 3). Each document is represented as an nr matrix, with entries showing word frequencies. This sparsity demands specialized

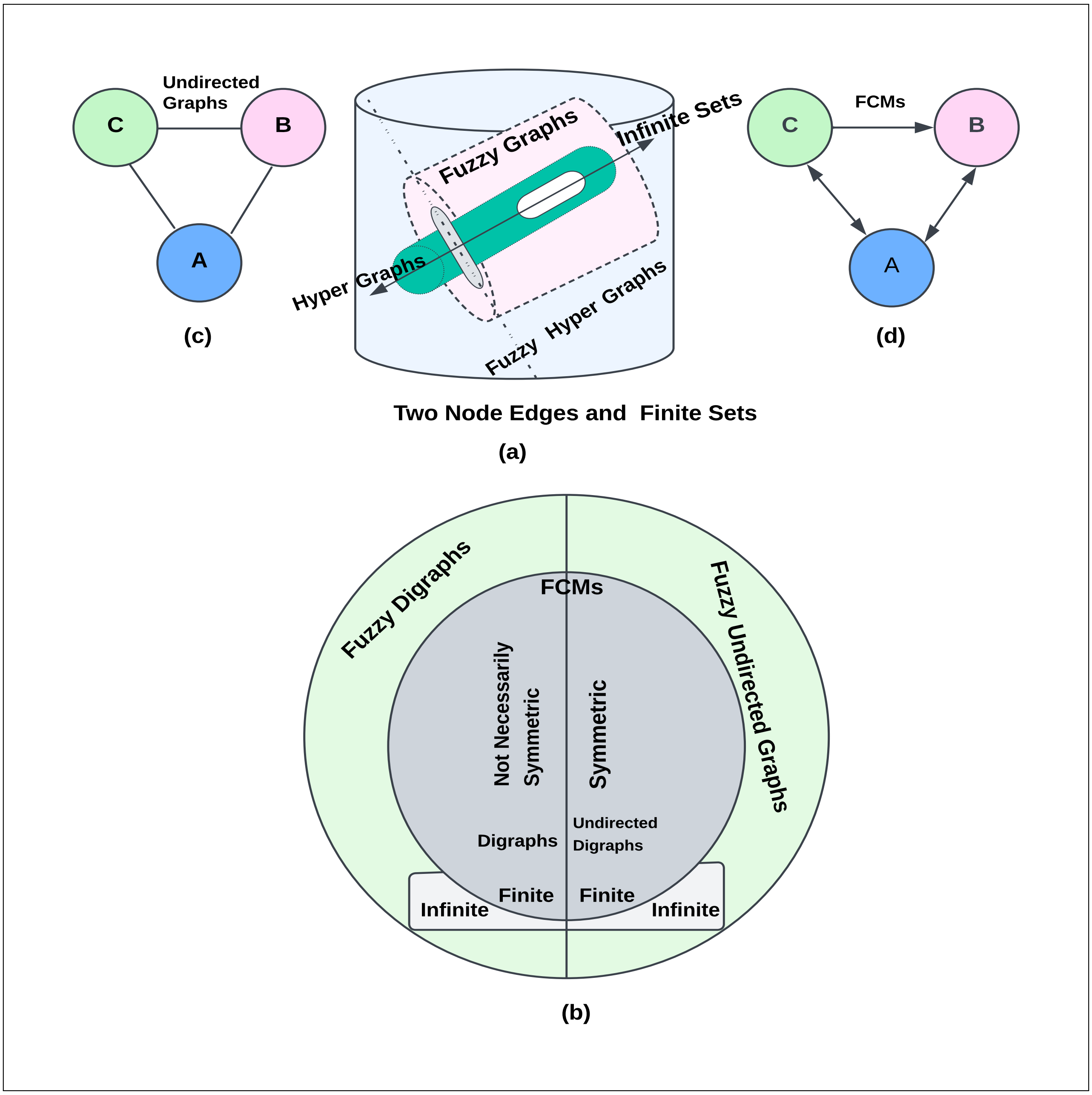

Figure 1. Visualization of the relationships among different graphs.

dimensionality reduction techniques to ensure efficient analysis.

Similarly, [49, 102] proposed the process of knowledge discovery in databases by analyzing the vast amount of text data and other dominant techniques employed for (i) Natural language processing and text pre-processing, (ii) Classify words by using dictionary-based techniques, and (iii) Classify texts or textual units with the help of algorithmic techniques (supervised algorithms) or unsupervised algorithms [103]. The following are a few techniques that may be employed for the task of text mining:

- Information Extraction
- Information Retrieval
- Summarization
- Clustering
- Categorization
- Concept Linkage

Moreover, the following are text Mining applications [54] which are most useful in the following sectors:

- Spam filtering, Patent analysis, Search strategy, article selection, Print Media, Broadcasting and Social media, Unsupervised Learning Methods for Text Data (clustering and topic modeling),

Table 1. A complete description of the text mining process.

Process	Description
Text Document	A useful information will be extracted from the document
Text Pre-Processing	Different strategies will be implemented for the purification of the data
Text transformation	The text will be transformed in a good manner for further process
Feature Selection	Consideration of Important and relevant features with respect to the task
Pattern Discovery	Extraction of standard patterns from the data with the help of suitable tools
Evaluation	Detailed analysis of results collected

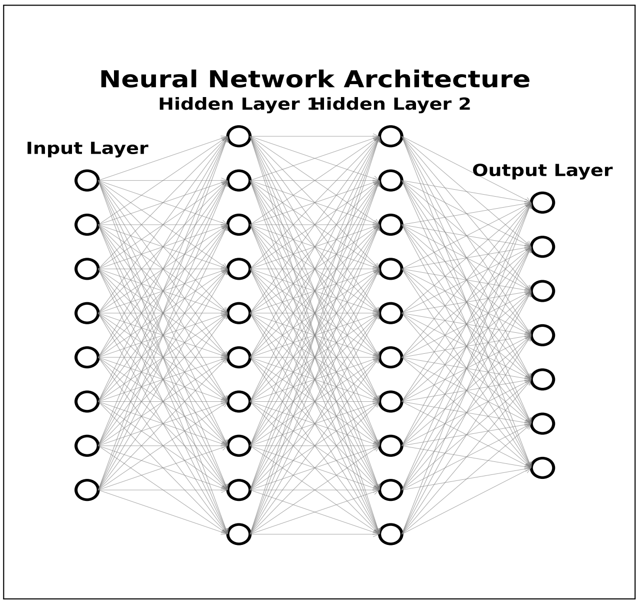

Figure 2. Neural Networks working and structure.

Hierarchical (Dendogram) Clustering, Latent Semantic Indexing, and Dimensionality Reduction methods for Text Mining

- Digital Libraries, Internet and Information technology sector, Energy, Telecommunications and other industries, Healthcare, Pharmaceutical and research companies,

Text mining plays a vital role in various industries by uncovering unique patterns that support production and knowledge management. Its application varies by sector—for example, the publishing industry uses it for cataloging, while insurance and banking focus on enhancing Customer Relationship Management (CRM) systems through automated message routing and natural language query handling [55, 88–91]. It also improves information retrieval and powers search engine responses. Additionally, text mining enables analysis of citation networks, which is central to our study. The main contributions of this paper are as follows:

- We proposed the Graph Attention Network model for node classification on the complete CORA dataset.
- Researchers extensively worked on probing the probability of citations from one scientific group of research papers to another.
- Highly cited papers and in the dataset were also evaluated.
- H-index of the papers using the CORA dataset is also considered in this study.
- We also compare our results with state-of-the-art machine learning models like Random Forest and Support Vector Machine.

This paper consists of the following sections: Section 2 includes a review of the literature, Section 3 consists of the Proposed Methodology, Section 4 deals with Experimental Design, Analysis & Results, and Section 5 is the conclusion.

2 Related Work

Neural Networks boosted recent research in every field of study. Graph Neural Networks (GNNs) open a new chapter of success in pattern recognition & classification for getting more reliable, faster computation processes. GNN is extensively used for single / multi-label classification. Graph Attention Networks (GAN) opened a new era of research that improves the overall structure by providing variable weights for every neighboring node. Numerous types of research have been carried out using GAN for multiple tasks using text, image, and video datasets. Researchers used GAN to work on more tasks on textual datasets than image and video datasets with GAN. Graph Attention Networks play a vital role in multiple tasks using textual datasets. It outperforms when compared with the other best models by a considerable margin. A few papers use graph attention networks due to their interest in solving different scenarios using text datasets.

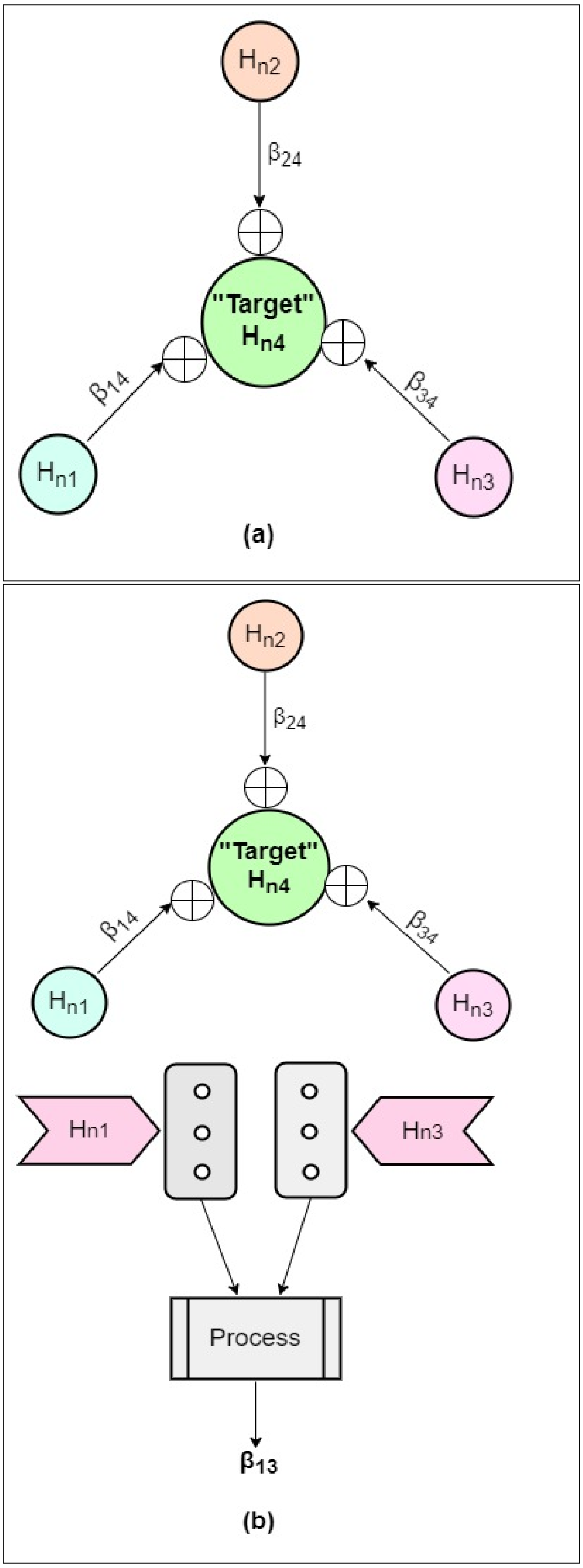

Figure 3. The difference is shown between GCN [33] and GAN [38].

Previous work [56] presented a novel idea for Account Takeover Detection: Selective Graph Attention Networks. This model is deployed for fraud detection by Alibaba Group companies. The graph attention mechanism within the sequence (GAS) pulls out inherent successive patterns and illustrates the account/ context with graph node embedding and Recurrent Neural Network (RNN). [57, 58] introduced the novel study for Relevance Matching between documents and texts using Multiresolution Graph Attention Networks. Researchers in [57] considered long documents or short texts for relevance matching between web searching and query-document matching information retrieval was a crucial task. An undirected graph is handy for extracting the structural information of documents. In undirected graphs, a vertex represents a keyword, and the degree of interaction between keywords means the weight of an edge. Keyword graphs supported [57] for drawing Multiresolution Graph Attention Network (MRGAN). MRGAN learned the multi-layered representation of vertices with the help of a Graph Convolutional Network (GCN), and it outperformed other models when compared.

Previous work [59] employed a Graph Attention Network to explore interactive information from text and Knowledge Graph for text entailment. Previous studies have found that textual entailment is used to extract context information from the sentences instead of complete knowledge without considering the background information. Recognizing textual entailment (RTE) architecture devised for a background knowledge interaction by [59], as it provides an opportunity to capture background knowledge regarding assumptions and hypotheses by external KG. [59] explored that KG subgraphs pay attention to encoding by graph attention networks with the help of graph-based entailment inference, and it outperformed the other models when compared.

Previous work [60] proposed a novel study for Deep Latent Representation of Multifaceted Social Effects in Recommender Systems by Dual Graph Attention Networks (DGAN). In this study, [60] described that DGAN consists of two parts: user-specific/ end-user attention weight, developed as its first part, and dynamic, as well as context-aware attention weight, developed as its second part. Moreover,data sparsity and cold start problems are faced by conventional collaborative filtering (CF) methods when employed for a recommendation system, but social recommendation systems may overcome them.

When considering the user's arena, friends may influence each other, like social influence and social homophily. The proposed model is equivalent to two dual graph attention networks: one dual GAT for users, which includes GAT to consider the social influence and study social homophily, and the second one is a dual GAT for items. Moreover, it outperformed the other models when compared. [61] projected the novel idea for Session-based social recommendation in online communities using Session-Based Social Recommendation via Dynamic Graph Attention Networks. Many pieces of research have already been carried out using Poisson or Gaussian matrix factorization. The proposed model by [61] collects user behavior using RNNs within a session and also deploys three kinds of recommender systems as well as compared by DGRec: (i) Classical methods, (ii) Social recommender system, (iii) session-based recommendation methods. The researchers found that Dynamic GAT outperformed other employed models. [62] introduced the novel model Heterogeneous Graph Attention Networks (HGAN) for semantic-level and node-level attention, and it will acquire knowledge of the critical nodes and their meta-paths. GAT leverages the attention mechanism for unique types of nodes or links, known as a homogeneous graph, and uses the concept of neural graph networks. The value of each meta path may be learned by Semantic-level attention and assigned suitable weights. Different attention weights are considered for each node at node-level attention depending on the importance of meta-path-based neighbors. Graph Attention Networks are employed to learn the importance of nodes and their neighbors. They use neighbors to accomplish the node classification and embedding of heterogeneous graphs, focusing on maintaining meta-path-based structural knowledge. HGAN outperformed the other models when compared. [63] presented the novel idea for recommendation using a Knowledge Graph Attention Network (KGAT). Path-based and regularization-based are two types of Collaborative knowledge graphs (CKG) structures for the recommendation. KGAT architecture has two models to address the concerns in high-order relation modeling: (i) Attention-based aggregation and (ii) Recursive embedding propagation. Attention-based aggregation considers the neural attention mechanism during propagation and learns the weight of nearby nodes. Cascaded propagations have attention weights and can unveil the value of high-order connectivity. Recursive embedding propagation modifies a node's embedding depending on its nearby nodes' embedding and subsequently executes such embedding propagation to acquire high-order channels in a linear time complexity. KGAT outperformed the other models when compared.

Previous work [64] proposed a model for Reasoning. The model is Probabilistic Logic Graph Attention Networks, and its purpose is to apply this model in Node classification and link prediction on graphs. Nowadays, an open area of research is Knowledge base completion graphs, which include the prediction of lost relations among entities in a knowledge graph. First-order logic and Markov logic networks drive the link prediction and question-answering related tasks of knowledge graphs. However, it has limitations relative to scalability and broader applicability in diverse areas of research due to intractable inference. Node classification and link prediction are complex tasks of knowledge graphs that get better results when they deal with graph attention neural networks, as they also consider features of neighboring entities. [64] proposed model is a refined model with a variational EM algorithm characterized by a Markov logic network, and it helped a lot to combine graph attention networks and first-order logic expeditiously. The proposed model outperformed the other models when compared. [65] gave the new document classification model and expressively purified the imported word embeddings using Graph Attention Topic Modeling Network (GATON). Probabilistic Latent Semantic Indexing (PLSI) produces generative patterns for the text collection, describing the document as a bag of words. However, it faces overfitting and high inference complexity issues and needs to consider the contemporary correlations in Latent Dirichlet Allocation (LDA). [65] provided a novel technique in which input is provided with amortized inference using word embedding to control the overfitting problem of PLSI. Researchers displayed the relationship between graph neural networks (GNNs) and the generative stochastic block model (SBM), primarily graph attention networks (GAT). [65] noticed that the pLSI could be seen as SBM on a particular bi-partite graph, and it is observable that nodes have two types, i.e., documents and words. To overcome the limitations of i.i.d. data supposition of vanilla amortized inference, [65] proposed a novel GATON architecture for correlated topic modeling. GATON develops the graph topology with the bilateral graph of documents and words, and it investigates the topic framework by convolving the node attributes over the graph with an attention mechanism. The proposed model outperformed the other models when compared.

Previous research [66] introduced a novel idea for the Early Detection of Rumors using Heterogeneous Graph Attention Networks on Twitter. Exciting features were extracted using data mining techniques from user profiles, text content, and different patterns of information. [66] employed a heterogeneous framework that structures the sources used to propagate rumors and text content. It consists of a word, tweets, and user nodes. It captures the global semantic relations of text contents and fuses them with the information entangled in the source used in the propagation of tweets for rumor detection. The proposed model outperformed the other models when compared. [67] presented the novel architecture for Citywide Traffic Flow Forecasting, enabling Spatial-Temporal Convolutional Graph Attention Networks. Spatial-temporal forecasting exploited geo-anomaly prediction, ride-hailing demand management, and user location prediction. The government will manage the transportation system with the help of proper foresight of citywide traffic flow.

Previous research [69] proposed the idea of Learning Signed Network Embedding via Graph Attention (SNEA) for Node Embeddings instead of Node Features. Many pieces of research have already been carried out for unsigned networks (consisting of only positive links) using GCN-based network embedding. [69] studied traffic forecasting using global region-wise dependencies and multi-granularity temporal dynamics employing a structured attention network, self-attention module, and graph attention subnet. It simultaneously models global inter-region traffic dependencies and multi-granularity temporal dynamics. The proposed model outperformed the other models when compared. [70] deployed a model SR-HGAN for node clustering, classification, and relation prediction. The SR-HGAN model consists of two layers, in which the core task of the primary layer is to assign attention weights to nodes and consider their symmetric relation. Moreover, considering the symmetric relation for various assignments, the secondary layer assigns weights to latent semantics. The two-layer mechanism is used to learn attention values and hierarchically obtains the ideal composite of neighbors and several proportionate relations. The proposed model outperformed the other models when compared. [71] explored the Recommendation system with Multimodal Graph Attention Network (MGAN). The MGAN model considers the customer's preference and performs extensive analysis for an item's suitability for a user. MGAN employs a novel framework of GCN that will acquire a high-order representation for forecasting and provoke information duplication. Furthermore, it adopts GNN to accumulate knowledge from neighboring nodes and amalgamate the accumulated outcomes with the knowledge of the head entity node. Afterward, MGAN concatenates different order representations of nodes to discriminate various contributions of orders and uses a gated attention mechanism to process them smoothly. The proposed model outperformed the other models when compared.

Reference [72] explored the model for Fraud Transactions Detection by using Intention-aware Heterogeneous Graph Attention Networks (IHGAN). The proposed model is designed to influence transactions and intentions using cross-interaction information. It contains two kinds of nodes, intention and transaction nodes, and transaction and intention are two types of edges. A sequence-based model is utilized for encoding every intention node due to its sequential actions for fraud detection, and the segmentation process is carried out keeping in view the identical behavior sequences using a sequence-based model. An attention mechanism is assumed to accumulate the intended neighbors for a target transaction. Finally, it continuously collects the transaction neighbors by a multi-head graph attention layer. The proposed model outperformed the other models when compared. [75] projected an idea of Aspect-based Sentiment Analysis (ABSA) with Weighted Relational Graph Attention Network (WRGAN). The task of ABSA is to ascertain the polarities of different aspects in one sentence. A relational graph attention network (R-GAN) is a type of GAN that considers diverse dependency relations. At the same time, it disregards the information concealed in the word pairs, and the WRGAT model is an over-reshaped dependency tree. WRGAN can manipulate precise syntactic information by joining features from dependency relations and their relative pair of words. The BERT-WRGAN framework is deployed for the ABSA task, where the BERT model is utilized to develop sentence-aspect pair embedding as inputs to the WRGAN. In addition, an index selection method is used to retain the word-level dependencies consistent with the word-piece unit of BERT. The proposed model outperformed the other models when compared.

As reported in [74], the Heterogeneous Graph Attention Networks (HGAN) model was employed for semi-supervised short text classification. The proposed model deals with both labeled and unlabeled data, enabling information to spread along the graph. The output of the HGAN model corresponds to the probabilities of a short text belonging to each text group. The proposed model outperformed the other models when compared. [76] introduced the model Hierarchical Graph Attention Networks (HGAN), which is used for the collection of reliable information without taking into account whether these two data entities belong to the same entity or not. Collective ER models and Pairwise ER models are two kinds of ER. A heterogeneous hierarchical Graph (HHG) is used to find entities, their attributes, and words. Another task of HHG is to draw the relationship among the entities; these relationships are similar to entity-entity relationships, attribute-entity relationships, and token-attribute relationships. The explored model is the Hierarchical Graph Attention Transformer (HierGAN), which amalgamates the Transformer attention mechanism and the HGAT model for node representations in HHG. The proposed model outperformed the other models when compared.

Reference [73] deployed a model for Short Text Generation Based on Adversarial Graph Attention Networks (SGANGAN). GAN (Generative Attention Nets) is used in the proposed model as a discriminator, creating a solid relationship among texts of the same type. The graph attention neural network is deployed in the proposed model as the discriminator in response to remarks, and it will lead the generator to an actual location for generating a particular type of short text. [68] deployed the idea for top-N recommendation using Disentangled Heterogeneous Graph Attention Network. To encode collaborative signals among customers and objects, [68] leverages embedding propagation to expressly contain

Note: Table 2 contains the different evaluation metrics used by authors for their result analysis and discussions, but all authors did not use all metrics, so those metrics represented with "x" as well as a few researchers did not use any evaluation metrics which we considered for comparison for our study. Therefore, those studies are not included in Table 2, but the description is provided. context statistics with rich semantic structure via the use of meta relations for decomposing high-order connectivity in HIN and advise a disentangled embedding propagation layer to accumulate remarkable components of semantic data for customers and items, respectively. The proposed model can spontaneously generate meta paths with semantic information while drawing the central aspects of fact flows in excessive-order connectivity.

However, the links have different polarities, e.g., positive and negative, in the real world, so these types of networks are signed. Since negative and positive links have different properties from one another, considering only negative links harms the excellence of network embedding. SNEA exploits masked self-attentional layers to accumulate more significant knowledge from neighboring nodes to develop the node embeddings built on balance theory.

Researchers found that different models embedded with the Graph Attention Network (GAN) for classification show the best performance in multiple domains like sentiment analysis, fraud detection, recommendation systems, vehicle detection, person identification, etc., using textual, image, or video datasets. Drawing a conclusion or inferring the data context is challenging, as textual data is sparse. Therefore, we are interested in text mining using Graph attention networks on the Cora dataset. We will find an essential node concerning its neighbors, and that node will draw a context of the class to which it belongs. We will perform a thorough analysis and discover the seven nodes, each comprising many features, which are scientific publications that cite one another. The study will also emphasize extracting highly cited papers and calculating the H-index using the CORA dataset. These are the novel tasks that the researchers will perform on the CORA dataset.

3 Methodology

The proposed model employs Graph Attention Networks (GANs) for text mining, with a focus on node classification within a dataset. The pre-processing stage involves calculating node vectors for the dataset. GANs are then utilized to classify the data and generate scores based on the significance of different classes. The model uses a single or multi-head attention mechanism to extract information with the same context. After the computation process, GANs produce unique class labels that correspond to various label occurrences within the dataset. The dataset is split into training and testing subsets to evaluate the model's performance. The flowchart illustrating the process of text mining using GANs is depicted in Figure 4.

Table 2. Description of text data sets using graph attention networks.

Authors	Datasets	Accuracy	FI Score	Precision	Recall	AUC	NDCG
[56]	Alibaba Group	0.57	0.22	x	x	x	x
[57]	The NF Corpus	0.9407	0.9533	x	x	x	x
	Ohsumed	0.8075	0.8118	x	x	x	x
[59]	SciTail	0.86	x	x	x	x	x
[60]	Epinions	x	x	0.7781	x	x	x
	WeChat Top-Story	x	x	0.0823	x	0.8165	x
[61]	Douban	x	x	x	0.1861	x	0.1959
	Delicious	x	x	x	0.4066	x	0.2944
	Yelp	x	x	x	0.0842	x	0.1427
[62]	ACM	x	x	x	x	x	x
	IMDB	x	x	x	x	x	x
	DBLP	x	x	x	x	x	x
[63]	Yelp	x	x	x	0.0712	x	0.0867
	Amazon book	x	x	x	0.1489	x	0.1006
	Last-FM	x	x	x	0.087	x	0.1325
[65]	20 News	x	0.85	0.859	0.842	x	x
	Reuters	x	0.977	0.975	0.979	x	x
[66]	Twitter	x	0.6834	x	x	0.9829	x
[69]	Bitcoin Alpha1	x	0.927	x	x	0.861	x
	Bitcoin OTC2	x	0.924	x	x	0.818	x
	Slash dot4	x	0.868	x	x	0.799	x
	Epinions3	x	0.933	x	x	0.861	x
[70]	DBLP	x	x	x	x	x	x
	Amazon book	x	x	x	x	x	0.2198
	Movielen	x	x	x	x	x	0.3913
[71]	Tiktok	x	x	0.1251	0.5965	x	0.3838
[72]	MovieLens	x	x	0.1271	0.5412	x	0.3251
	Alibaba Group	x	x		0.2744	0.9687	x
[75]	Laptop14	0.8213	x	x	x	x	x
	Restaurant14	0.8661	x	x	x	x	x
	Twitter	0.7673	x	x	x	x	x
	MAMS	0.8436	x	x	x	x	x
[74]	AGNews	0.721	0.7161	x	x	x	x
	Snippets	0.8236	0.7444	x	x	x	x
	Ohsumed	0.4268	0.2482	x	x	x	x
	TagMyNews	0.6172	0.65381	x	x	x	x
	MR	0.6275	0.6236	x	x	x	x
	Twitter	0.6321	0.6248	x	x	x	x
	AGNews	0.7023	0.6843	x	x	x	x
	Snippets	0.794	0.7769	x	x	x	x
	Ohsumed	0.4208	0.2571	x	x	x	x
	TagMyNews	0.582	0.4977	x	x	x	x
	MR	0.6118	0.5977	x	x	x	x
	Twitter	0.626	0.6047	x	x	x	x

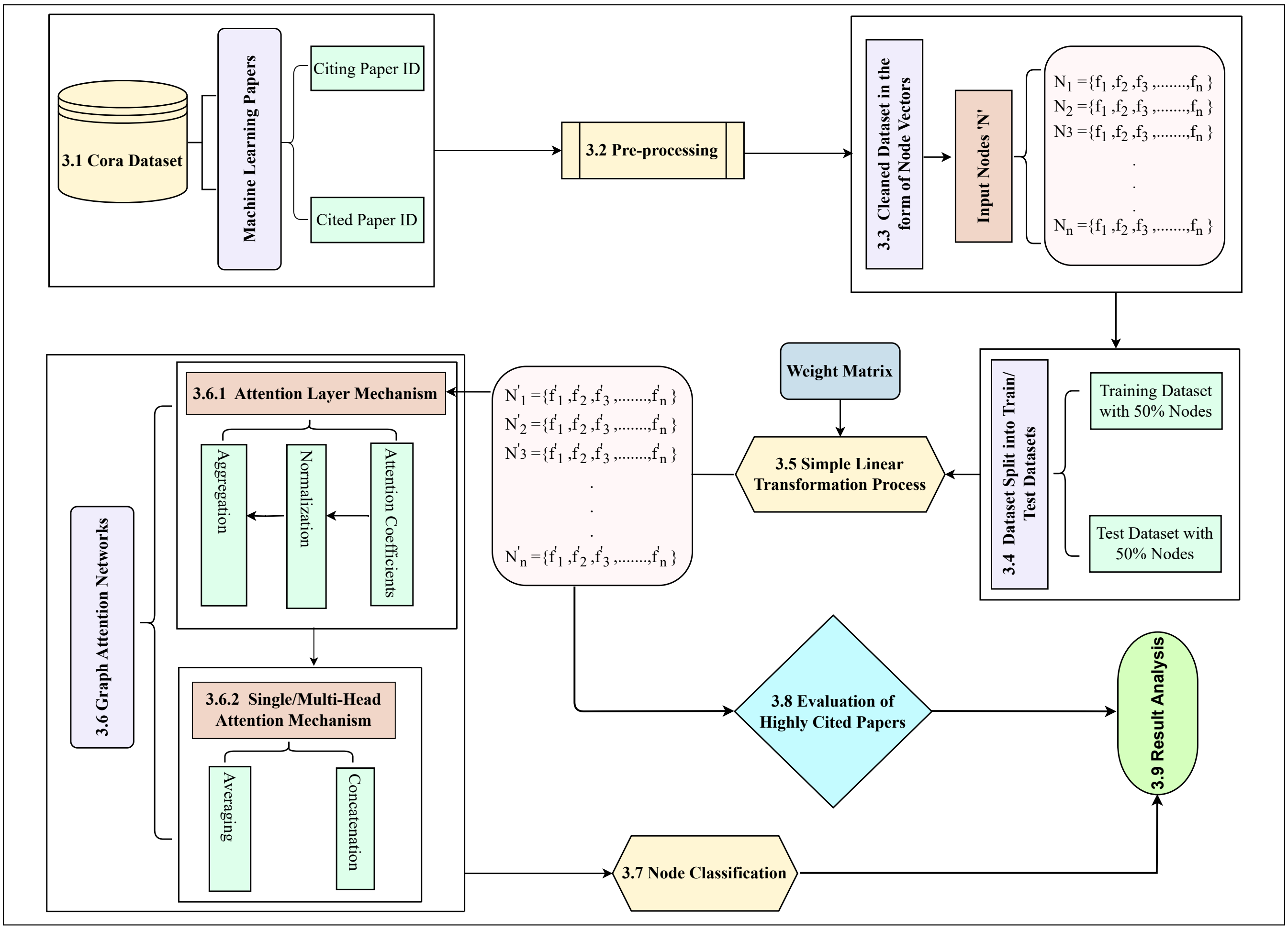

Figure 4. Flow chart of text mining using GAN.

3.1 Dataset

The proposed Graph Attention Networks method for text mining was experimented on the benchmark "Cora" dataset, which is publicly available at the Cora Dataset. It contains 2708 i.e *Case_Based, Genetic_Algorithm, Neural_Networks, Probabilistic_Methods, Reinforcement_Learning, Rule_Learning and Theory* are machine learning papers. The dataset consists of two files with the extension of .content file, which contains the description of the paper in the $< paper\_id >$, $< word\_attributes > + < class\_label > format$. Moreover, a unique string ID is shown in the first line of the paper, and it is shown as a binary number (indicated by 1) if each word is available in the vocabulary or (indicated by 0) if the number is absent in the vocabulary and paper class is represented by the last line. Furthermore, citation graphs of the corpus are contained in the other file, i.e., the "cite" file. Each line describes a link in the <ID of cited paper><ID of citing paper>format is represented in each line, and it works as a link among them. A strategy was developed in such a way that another paper should cite every paper in the corpus. The corpus contains 2708 papers and 1433 unique words, which were attained by pre-processing steps. It is pertinent to mention that those words with a document frequency of less than ten were removed.

3.2 Cleaned Dataset in the form of Node Vectors

The cleaned dataset received after the pre-processing step is now pure from anomalies, and the dataset contains a total of 2708 nodes $N_i \epsilon \Re^F$, and each node contains 'n' features. Eq. 2 shows that the nodes contain different machine-learning papers with unique characteristics/ features.

$$
\begin{aligned}
N_1 &= \{f_{11}, f_{12}, f_{13}, \dots, f_{1n}\} \\
N_2 &= \{f_{21}, f_{22}, f_{23}, \dots, f_{2n}\} \\
N_3 &= \{f_{31}, f_{32}, f_{33}, \dots, f_{3n}\} \\
&\vdots \\
N_n &= \{f_{n1}, f_{n2}, f_{n3}, \dots, f_{nn}\}
\end{aligned} \tag{2}
$$

3.3 Dataset Split into Train/ Test sets

The purpose of splitting the data into two parts is to avoid overfitting. Therefore, the researcher split the dataset into two sets: 50 percent for training and 50 percent for the testing phase.

3.4 Simple Linear Transformation

Linear transformation remodels the low-level input features into higher-level features by providing weights to the input features. Therefore, shared linear transformation, parametrized by a weight matrix, $W \in \Re^{\acute{F} \times F}$, is applied to every node shown in Eq. 3.

$$N'_i = WN_i \tag{3}$$

where N'_i is a new set of node vectors potentially of different cardinality F' and $N'_i \in \Re^{F'}$ which is shown in Eq. 4.

$$\begin{aligned} N'_1 &= f'_{11}, f'_{12}, f'_{13}, \ldots, f'_{1n}\} \\ N'_2 &= \{f'_{21}, f'_{22}, f'_{23}, \ldots, f'_{2n}\} \\ N'_3 &= \{f'_{31}, f'_{32}, f'_{33}, \ldots, f'_{3n}\} \\ &\vdots \\ N'_n &= \{f'_{n1}, f'_{n2}, f'_{n3}, \ldots, f'_{nn}\} \end{aligned} \tag{4}$$

3.5 Graph Attention Layer Mechanism

3.5.1 Attention Layer Mechanism

The novel idea of an attention mechanism was floated by [105], and the idea of an attention mechanism came into existence to address the bottleneck problem that arose with the usage of a fixed-length encoding vector, where the decoder provides minimal information from the input vector. Thus, this idea works well to address the limitations of graph convolutional networks by providing a score of attention weights to its neighbors for their importance for full nodes. Therefore, the input to the layer is $N = N_1, N_2, N_3, \ldots, N_n$, where $N \in \Re^F$. where "n" is the number of nodes, each containing several features "F." The layer produces a new set of node features $\acute{N}_i = \acute{N}_1, \acute{N}_2, \acute{N}_3, \cdot, \acute{N}_n$, where N_i are set of output nodes. The whole process in this layer is shown in Figure 5.

Moreover, the attentional layer is broken into the following three parts:-

- Attention Coefficients
- Normalization
- Aggregation

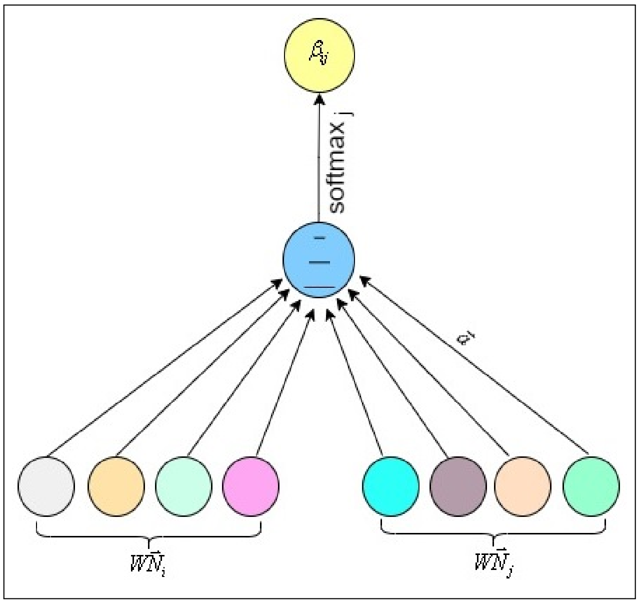

Figure 5. Attention layer mechanism.

- **Attention Coefficients:** After the linear transformation process, an un-normalized attention score between any two neighbors will be calculated. In this scenario, a concatenation process will be implemented on N_i embeddings of two nodes, and we use the $\|$ symbol for concatenation. Then, we will take a dot product of it with the learnable weight vector a. After doing this, we will use the LeakyReLU activation function. Moreover, a shared attentional mechanism like the self-attention mechanism on the nodes is used, i.e., $a : \mathbb{R}^{F'} \times \mathbb{R}^F \rightarrow \mathbb{R}$ to compute attention coefficients μ_{ij} to compute attention coefficients μ_{ij}. The attention coefficient is an important and essential step, which calculates the importance of node N_j's features to node N_i. It is an essential step as it considers that every node in the corpus will attend to another node without considering the structural information of the neighboring nodes. Furthermore, Eq. 5 defines the attentional coefficients as:

$$\mu_{ij} = LeakyRelu(\overrightarrow{a}^T (N_i \,||\, N_j)) \tag{5}$$

- **Normalization:** In the next process level, a softmax function is implemented on Eq. 5 as we can make the coefficients comparable among all "j" nodes. Eq. 6 is used for normalization purposes.

$$\beta_{ij} = \frac{exp(\mu_{ij})}{\sum_{k \in N_i} exp(\mu_{ik})} \tag{6}$$

- **Aggregation:** Aggregation step is similar to that used in GCN. Attention scores are used for scaling the embedded neighbors, and these neighbors are aggregated as in Eq. 7.

$$N_1'' = \sigma(\sum_{j \epsilon N_i} \beta_{ij} N_j') \tag{7}$$

- **Single/ Multi-Head Attention Mechanism:** Similar to Convolutional Networks, GAN also uses a multi-head attention mechanism for two purposes: first, to enrich the capacity of the model and, secondly, for the stability of the learning process. Therefore, "K" independent attention mechanisms execute the transformation of Eq. 6, and their output may be aggregated based on the requirement. Thus, concatenation shown in Eq. 8 is used for intermediary layers, and the averaging concept is used for the final layer as shown in Eq. 9.

$$N_i'' = ||_{k=1}^{K} \sigma(\sum_{j \epsilon N_i} \beta_{ij}^{k} N_j') \tag{8}$$

$$N_1'' = \sigma(\frac{1}{K} \sum_{k=1}^{K} \sum_{j \epsilon N_i} \beta_{i,j}^{k} N_j') \tag{9}$$

Suppose $d(N_i) = 64$, $K = 4$, then $d(N_{i,k}) = 16$ using Eq. 8. Hence, each "head" is responsible for 16 dimensions.

4 Experimental Design, Analysis, and Results

The researchers collected the Cora citation dataset from https://linqs-data.soe. and another link is https://relational.fit.cvut.cz/da taset/CORA [80–82]. This study aims to extract highly cited papers from the Cora Dataset, consisting of directed graphs that make a citation link between the documents belonging to different classes. This study also seeks to classify the dataset based on graph attention networks. The path to achieving the goals mentioned above is as follows:

- Node classification using graph attention networks on the CORA dataset
- H-index on Cora dataset and extraction of highly cited papers from the pool of citations, i.e., CORA dataset

Now, revising the study's preface, the previously implemented models do not include the citation as a parameter for the model to be trained on, while our approach considers the citation as a part of the dataset for training.
The dataset is followed by 2708 scientific publications collected in graphs, as shown in Figure 6, and classified into seven nodes or classes.

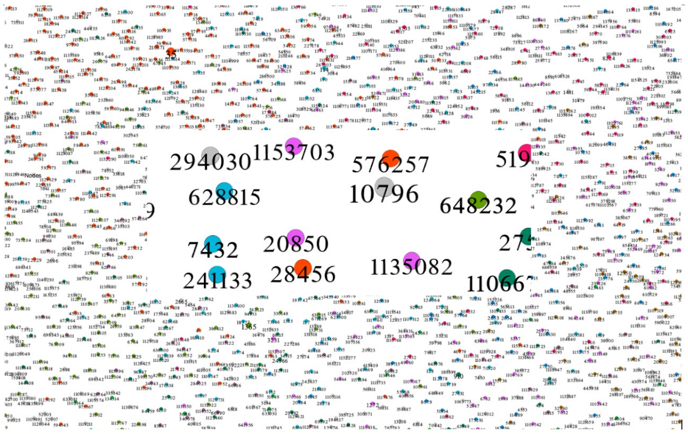

Figure 6. Graph representation of the Cora dataset.

The description of the dataset is given as follows:

- $(5429, 2)$ edge shape represents the links between the papers with two key features
- Node features of shape $(2708, 1433)$
- The articles are classified as binary classes in which "0" represents a paper not existing and "1" illustrates an article exists

The dataset consists of two files: Cora citations and Cora content. Cora sites consist of two features, namely "target" and "source," while core content consists of paper_id along with its features and belonging classes. The data set was processed in pre-processing steps, and null or empty values were removed. To achieve high-level features from low-level features, the following techniques were applied:

- Pre-processing the given dataset
- Applied one-hot encoding on the dataset
- Applied linear transformation of one-hot encoded data
- Separating features and labels into different variables
- Train / Test split with a ratio of 50:50

The implementation of Graph attention networks for node classification achieves the goal. The steps to

Table 3. Probability of intra-classes for citation (in decimal form).

	P(CB)	P(GA)	P(NN)	P(PM)	P(RLg)	P(RL)	P(Theory)
P(CB)	0.96041	0.00031	0.00016	0.01562	0.00052	0.165761	0.05689
P(GA)	0.00551	0.99861	0.00001	0.00542	0	0.01698	0.04226
P(NN)	0.00139	0.00002	0.98854	0.50645	0.00059	0.08024	0.19832
P(PM)	0.00656	0	0.00519	0.41824	0	0.17988	0.54079
P(RLg)	0.00238	0.00031	0.00148	0.01641	0.99764	0.01584	0.03134
P(RL)	0.01942	0.00071	0.00077	0.00502	0.00026	0.49422	0.04427
P(Theory)	0.00432	0.00004	0.00386	0.03284	0.001	0.04713	0.08612

implement the GAT module are as follows:

- Dataset
- Pre-processing
- Dataset Split
- Simple Linear Transformation
- Application of Graph Attention Networks for Node Classification and Graph Representation
- Finding the Citation Probability of Each Paper on a Node to Another Node
- Extraction of H-index and Highly Cited Papers from the Pool of Citations and Graph Representation of Highly Cited Papers

The graph representation of the overall dataset depicts the paper, citing each other using directed graphs. The graph can be seen in Figure 7.

The graph attention mechanism was implemented for computing pairwise attention scores, and then the scores were normalized. The node states of the neighbors and applied attention scores to it, and the aggregation results achieved by neighbors were aggregated. The implementation mentioned above achieved the following results:

- Researcher used the LeakyRelu activation function and kernel attention function with eight heads
- The results were compiled by concatenating the individual results
- After testing the implemented model, the researchers achieved an accuracy that fluctuates from 82% to 78.6%, F1 Score 78.14% on the test data set for node classification [78], [104]
- The probability of a paper belonging to each class was also calculated

As we observed, the achieved accuracy is comparatively lower than the previous model by a margin of 1% and 4.6%. To justify the loss in accuracy, we will counter by stating that the previous author has exempted the CORA citation file from the CORA dataset. In our case, the citation has increased the intra-class confusion, which causes accuracy to drop, as it can be visible in the table and the probability of each paper to each class as shown in Table 3. The graph representation of the node-classified papers citing each other using directed graphs. The graphs can be seen in Figure 8. We found the highly cited papers as shown in Table 4, and they follow the order of citation in the graph:

Table 4. Highly cited papers extracted from the CORA dataset.

Paper_ID	Rank	No of Citations
35	1	166
6213	2	76
1365	3	74
3229	4	61
114	5	42
910	6	41
4330	7	38
1272	8	32
3231	9	32
4584	10	32
19621	11	31
2440	12	30
24966	13	29
2665	14	28
6214	15	28

The results validated the purpose of this research, and after the deduction of complete graphs, we found the H-index of the paper from the pool of citations. Paper_ID: 12182 in the Cora data set has a high H-index (20) as the H-index uses two parameters: (i) Quantity – Number of papers, (ii) Quality – Number of citations.

Figure 7. Complete graph drawn with node classification.

Moreover, the researcher found that paper_id 35 is a highly cited (166) paper in the pool of citations. Furthermore, 15 highly cited papers are shown in Figure 8 and Table 4.

4.1 Comparative Study of Proposed Model With Machine Learning Models

The researcher implemented a support vector machine and random forest classifiers for the proposed task and compared their results with the Deep Learning (ML) based model, i.e., Graph Attention Networks, and found that the proposed study outperformed both ML models with clear margin as shown in Figure 9. Therefore, we recommend that the deep learning-based model will provide better results for CORA and other citation datasets.

5 Conclusion and Future Scope

The goal of this study and the novelty of the research is to observe and closely study the relationships among different papers on separate nodes for node classification and to extract the highly cited papers. Moreover, this task was extended, and the H-index on the CORA dataset was evaluated. To accomplish these objectives, we thoroughly analyzed the citation dataset, which is a citation network. The primary task in this data set was node classification, extracting highly cited papers, and calculating the H-index.

The structural characteristics of the citation network itself have a very complex architecture and need particular emphasis for performing the required task. This research includes the study of bonding among the different nodes by evaluating the probability of each node to other nodes and comparing its relationship with other nodes. We achieved a fluctuating accuracy range from 82% to 78.6% and an F1 Score of 78.14% on

Figure 8. Extraction and visualization of highly cited papers.

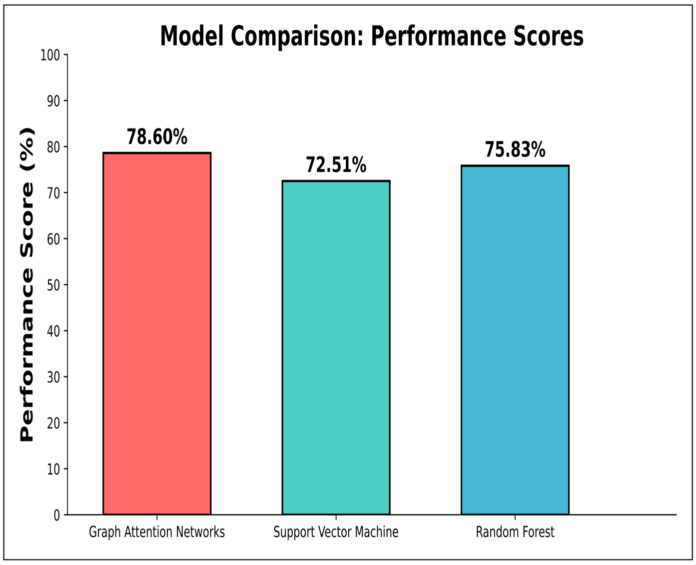

Figure 9. Comparative analysis (Accuracy (%)) of graph attention network model with machine learning models.

the test dataset, which is comparatively less than the already achieved accuracy by a margin of 1% to 4.6% by applying graph attention networks. Still, they only used one part of the dataset and ignored the other for node classification. On these grounds, the accuracy we achieved is much better than the already established accuracy on the benchmark CORA dataset. Moreover, we successfully extracted highly cited papers and H-index from the CORA dataset. This study found seven nodes consisting of machine learning papers, but with different classifications. Moreover, the researcher found a high H-index of paper_ID 12182 in this study, and highly cited papers have paper_id 35. Therefore, this study provides the inner sense of citation network structure and the influence of one paper on another in the node. Moreover, this research will open avenues, such as investigating the potential impact of incorporating domain-specific knowledge into the node classification process. This could involve developing novel feature extraction techniques that leverage domain-specific information or exploring the use of domain-specific ontologies to guide the node classification process. Furthermore, in the future, we will explore and develop novel methods for assessing the reliability of citation data in scientific publications. In addition, in the future, we will expand our study to other standard datasets like Citeceer, PubMed, etc. This could involve using machine learning techniques to identify and flag potentially unreliable citation information or exploring the use of crowd-sourcing or expert reviewers to validate citation data.

Data Availability Statement

The code supporting the findings of this study is openly available on GitHub at https://github.com/TahirSher/Graph-Attention-Networks-for-Node-Classification-and-Highly-Cited-Papers. All experiments were conducted using Google Colab with Python 3.10 and standard library versions compatible with Python 3.10.

Funding

This work was supported without any funding.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Alachram, H., Chereda, H., Beißbarth, T., Wingender, E., & Stegmaier, P. (2021). Text mining-based word representations for biomedical data analysis and protein-protein interaction networks in machine learning tasks. *PloS one, 16*(10), e0258623. [CrossRef]

[2] Backstrom, L., & Leskovec, J. (2011). Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining* (pp. 635–644). [CrossRef]

[3] Akoglu, L., Tong, H., & Koutra, D. (2015). Graph based anomaly detection and description: a survey. *Data mining and knowledge discovery, 29*(3), 626–688. [CrossRef]

[4] Zhang, S., Zhou, D., Yildirim, M. Y., Alcorn, S., He, J., Davulcu, H., & Tong, H. (2017, June). Hidden: hierarchical dense subgraph detection with application to financial fraud detection. In *Proceedings of the 2017 SIAM international conference on data mining* (pp. 570-578). Society for Industrial and Applied Mathematics. [CrossRef]

[5] Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

[6] Monti, F., Boscaini, D., Masci, J., Rodolà, E., Svoboda, J., & Bronstein, M. M. (2017, July). Geometric Deep Learning on Graphs and Manifolds Using Mixture Model CNNs. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 5425-5434). IEEE. [CrossRef]

[7] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2008). The graph neural network model. *IEEE transactions on neural networks, 20*(1), 61–80. [CrossRef]

[8] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., & Hwang, D. U. (2006). Complex networks: Structure and dynamics. *Physics reports, 424*(4-5), 175-308. [CrossRef]

[9] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., … Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open, 1*, 57–81. [CrossRef]

[10] Wu, Y., Lian, D., Xu, Y., Wu, L., & Chen, E. (2020, April). Graph convolutional networks with markov random field reasoning for social spammer detection. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 01, pp. 1054-1061). [CrossRef]

[11] Dessì, D., Osborne, F., Recupero, D. R., Buscaldi, D., & Motta, E. (2021). Generating knowledge graphs by employing natural language processing and machine learning techniques within the scholarly domain. *Future Generation Computer Systems, 116*, 253-264. [CrossRef]

[12] Nicholson, D. N., & Greene, C. S. (2020). Constructing knowledge graphs and their biomedical applications. *Computational and structural biotechnology journal, 18*, 1414-1428. [CrossRef]

[13] MacLean, F. (2021). Knowledge graphs and their applications in drug discovery. *Expert opinion on drug discovery, 16*(9), 1057-1069. [CrossRef]

[14] Chakrabarti, D., & Faloutsos, C. (2006). Graph mining: Laws, generators, and algorithms. *ACM computing surveys (CSUR), 38*(1), 2-es. [CrossRef]

[15] West, D. B. (2001). *Introduction to graph theory* (Vol. 2). Prentice hall Upper Saddle River.

[16] Portmann, E., Kaltenrieder, P., & Pedrycz, W. (2015). Knowledge representation through graphs. *Procedia Computer Science, 62*, 245–248. [CrossRef]

[17] Mordeson, J. N., & Nair, P. S. (2012). *Fuzzy graphs and fuzzy hypergraphs* (Vol. 46). Physica. [CrossRef]

[18] Kosko, B. (1986). Fuzzy cognitive maps. *International journal of man-machine studies, 24*(1), 65–75. [CrossRef]

[19] Bax, L., Ikeda, N., Fukui, N., Yaju, Y., Tsuruta, H., & Moons, K. G. (2009). More than numbers: the power of graphs in meta-analysis. *American journal of epidemiology, 169*(2), 249–255. [CrossRef]

[20] Arora, S. (2020). A survey on graph neural networks for knowledge graph completion. *arXiv preprint arXiv:2007.12374*.

[21] Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *arXiv preprint arXiv:1606.09375*.

[22] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature, 521*(7553), 436–444. [CrossRef]

[23] Chung, F. R. (1997). *Spectral graph theory* (Vol. 92). American Mathematical Soc. [CrossRef]

[24] Henaff, M., Bruna, J., & LeCun, Y. (2015). Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*.

[25] Wang, H., Wang, J., Wang, J., & others. (2017). Graph representation learning with generative adversarial nets. *AAAI, Graphgan*.

[26] You, J., Ying, R., Ren, X., Hamilton, W., & Leskovec, J. (2018, July). Graphrnn: Generating realistic graphs with deep auto-regressive models. In *International conference on machine learning* (pp. 5708-5717). PMLR.

[27] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). Graph attention networks. *arXiv preprint arXiv:1710.10903*.

[28] Lee, J. B., Rossi, R., & Kong, X. (2018). Graph classification using structural attention. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 1666–1674). [CrossRef]

[29] Li, Y., Tarlow, D., Brockschmidt, M., & Zemel, R. (2015). Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*.

[30] Tai, K. S., Socher, R., & Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

[31] Sperduti, A., & Starita, A. (1997). Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks, 8*(3), 714–735. [CrossRef]

[32] Dai, H., Kozareva, Z., Dai, B., Smola, A., & Song, L. (2018, July). Learning steady-states of iterative algorithms over graphs. In International conference on machine learning (pp. 1106-1114). PMLR.

[33] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

[34] Levie, R., Monti, F., Bresson, X., & Bronstein, M. M. (2018). Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Transactions on Signal Processing, 67*(1), 97–109. [CrossRef]

[35] Zhang, D., Yin, J., Zhu, X., & Zhang, C. (2018). Network representation learning: A survey. *IEEE transactions on Big Data, 6*(1), 3–28. [CrossRef]

[36] Pan, S., Wu, J., Zhu, X., Zhang, C., & Wang, Y. (2016). Tri-party deep network representation. In *25th International Joint Conference on Artificial Intelligence* (*IJCAI-16*). AAAI Press/International Joint Conferences on Artificial Intelligence.

[37] Kriege, N. M., Johansson, F. D., & Morris, C. (2020). A survey on graph kernels. *Applied Network Science, 5*(1), 6. [CrossRef]

[38] Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). Graph attention networks. *stat, 1050*(20), 10-48550.

[39] Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive representation learning on large graphs. *Advances in neural information processing systems, 30*.

[40] Zhang, J., Shi, X., Xie, J., Ma, H., King, I., & Yeung, D.-Y. (2018). Gaan: Gated attention networks for learning on large and spatiotemporal graphs. *arXiv preprint arXiv:1803.07294*.

[41] Liu, Z., Chen, C., Li, L., Zhou, J., Li, X., Song, L., & Qi, Y. (2019). Geniepath: Graph neural networks with adaptive receptive paths. In *Proceedings of the AAAI Conference on Artificial Intelligence* (pp. 4424–4431). [CrossRef]

[42] Madni, H. A., Anwar, Z., & Shah, M. A. (2017). Data mining techniques and applications—A decade review. In *2017 23rd international conference on automation and computing* (*ICAC*) (pp. 1–7). [CrossRef]

[43] Imran, A. S., Yang, R., Kastrati, Z., Daudpota, S. M., & Shaikh, S. (2022). The impact of synthetic text generation for sentiment analysis using GAN based models. *Egyptian Informatics Journal, 23*(3), 547-557. [CrossRef]

[44] Aggarwal, C. C., & Zhai, C. (2012). An introduction to text mining. In *Mining text data* (pp. 1–10). Springer. [CrossRef]

[45] Usai, A., Pironti, M., Mital, M., & Aouina Mejri, C. (2018). Knowledge discovery out of text data: a systematic review via text mining. *Journal of knowledge management, 22*(7), 1471-1488. [CrossRef]

[46] Croft, W. B., Metzler, D., & Strohman, T. (2010). *Search engines: Information retrieval in practice* (Vol. 520). Addison-Wesley.

[47] Jones, K. S., & Willett, P. (1997). *Readings in information retrieval*. Morgan Kaufmann.

[48] Baeza-Yates, R. A., & Ribeiro-Neto, B. A. (2011). *Modern Information Retrieval - the concepts and technology behind search, Second edition*. ACM, Inc.

[49] Krzywicki, A., Wobcke, W., Bain, M., Martinez, J. C., & Compton, P. (2016). Data mining for building knowledge bases: techniques, architectures and applications. *The Knowledge Engineering Review, 31*(2), 97-123. [CrossRef]

[50] Salton, G., & McGill, M. J. (1983). *Introduction to modern information retrieval*. mcgraw-hill.

[51] Manning, C. D. (2008). Prabhakar Raghavan, and

Hinrich Schutze. *Introduction to information retrieval*. Cambridge University Press.

[52] Grossman, D. A., & Frieder, O. (2004). *Information retrieval: Algorithms and heuristics* (Vol. 15). Springer Science & Business Media.

[53] Hotho, A., Nürnberger, A., & Paaß, G. (2005). A brief survey of text mining. In *LDV Forum* (pp. 19–62).

[54] Bolasco, S., Canzonetti, A., Capo, F. M., Della Ratta-Rinaldi, F., & Singh, B. K. (2005, October). Understanding text mining: A pragmatic approach. In *Knowledge Mining: Proceedings of the NEMIS 2004 Final Conference* (pp. 31-50). Berlin, Heidelberg: Springer Berlin Heidelberg. [CrossRef]

[55] Gupta, V., & Lehal, G. S. (2009). A survey of text mining techniques and applications. *Journal of emerging technologies in web intelligence, 1*(1), 60–76. [CrossRef]

[56] Tao, J., Wang, H., & Xiong, T. (2018). Selective graph attention networks for account takeover detection. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)* (pp. 49–54). [CrossRef]

[57] Zhang, T., Liu, B., Niu, D., Lai, K., & Xu, Y. (2018). Multiresolution graph attention networks for relevance matching. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 933–942). [CrossRef]

[58] Wen, K., Gu, X., & Cheng, Q. (2020). Learning dual semantic relations with graph attention for image-text matching. *IEEE transactions on circuits and systems for video technology, 31*(7), 2866-2879. [CrossRef]

[59] Chen, D., Li, Y., Yang, M., Zheng, H.-T., & Shen, Y. (2019). Knowledge-aware textual entailment with graph attention network. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 2145–2148). [CrossRef]

[60] Wu, Q., Zhang, H., Gao, X., He, P., Weng, P., Gao, H., & Chen, G. (2019). Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. In *The World Wide Web Conference* (pp. 2091–2102). [CrossRef]

[61] Song, W., Xiao, Z., Wang, Y., Charlin, L., Zhang, M., & Tang, J. (2019). Session-based social recommendation via dynamic graph attention networks. In *Proceedings of the Twelfth ACM international conference on web search and data mining* (pp. 555–563). [CrossRef]

[62] Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., & Yu, P. S. (2019). Heterogeneous graph attention network. In *The world wide web conference* (pp. 2022–2032). [CrossRef]

[63] Wang, X., He, X., Cao, Y., Liu, M., & Chua, T.-S. (2019). Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 950–958). [CrossRef]

[64] Harsha Vardhan, L. V., Jia, G., & Kok, S. (2020). Probabilistic logic graph attention networks for reasoning. In *Companion Proceedings of the Web Conference 2020* (pp. 669–673). [CrossRef]

[65] Yang, L., Wu, F., Gu, J., Wang, C., Cao, X., Jin, D., & Guo, Y. (2020). Graph attention topic modeling network. In *Proceedings of The Web Conference 2020* (pp. 144–154). [CrossRef]

[66] Huang, Q., Yu, J., Wu, J., & Wang, B. (2020). Heterogeneous graph attention networks for early detection of rumors on twitter. In *2020 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–8). [CrossRef]

[67] Zhang, X., Huang, C., Xu, Y., & Xia, L. (2020). Spatial-temporal convolutional graph attention networks for citywide traffic flow forecasting. In *Proceedings of the 29th ACM international conference on information & knowledge management* (pp. 1853–1862). [CrossRef]

[68] Wang, Y., Tang, S., Lei, Y., Song, W., Wang, S., & Zhang, M. (2020). Disenhan: Disentangled heterogeneous graph attention network for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 1605–1614). [CrossRef]

[69] Li, Y., Tian, Y., Zhang, J., & Chang, Y. (2020). Learning signed network embedding via graph attention. In *Proceedings of the AAAI Conference on Artificial Intelligence* (pp. 4772–4779). [CrossRef]

[70] Zhang, Z., Huang, J., & Tan, Q. (2020). SR-HGAT: symmetric relations based heterogeneous graph attention network. *IEEE Access, 8*, 165631–165645. [CrossRef]

[71] Tao, Z., Wei, Y., Wang, X., He, X., Huang, X., & Chua, T.-S. (2020). MGAT: multimodal graph attention network for recommendation. *Information Processing & Management, 57*(5), 102277. [CrossRef]

[72] Liu, C., Sun, L., Ao, X., Feng, J., He, Q., & Yang, H. (2021). Intention-aware heterogeneous graph attention networks for fraud transactions detection. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (pp. 3280–3288). [CrossRef]

[73] Chen, M. (2021). Short Text Generation Based on Adversarial Graph Attention Networks. In *2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture* (pp. 2889–2893). [CrossRef]

[74] Yang, T., Hu, L., Shi, C., Ji, H., Li, X., & Nie, L. (2021). HGAT: Heterogeneous graph attention networks for semi-supervised short text classification. *ACM Transactions on Information Systems (TOIS), 39*(3), 1–29. [CrossRef]

[75] Huo, Y., Jiang, D., & Sahli, H. (2021). Aspect-based Sentiment Analysis with Weighted Relational Graph Attention Network. In *Companion Publication of the 2021 International Conference on Multimodal Interaction* (pp. 63–70). [CrossRef]

[76] Yao, D., Gu, Y., Cong, G., Jin, H., & Lv, X. (2022). Entity Resolution with Hierarchical Graph Attention Networks. In *Proceedings of the 2022 International Conference on Management of Data* (pp. 429–442). [CrossRef]

[77] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[78] Shukla, S. S. P., & Singh, M. P. (2024). Stacked classification approach using optimized hybrid deep learning model for early prediction of behaviour changes on social media. *ACM Transactions on Asian and Low-Resource Language Information Processing, 23*(11), 1-22. [CrossRef]

[79] Kanishka Silva, K. S., Can, B., Sarwar, R., Blain, F., & Mitkov, R. (2023). Text data augmentation using generative adversarial networks–a systematic review. *Journal of Computational and Applied Linguistics, 1*, 6-38. [CrossRef]

[80] Izadi, M. R., Fang, Y., Stevenson, R., & Lin, L. (2020). Optimization of graph neural networks with natural gradient descent. In *2020 IEEE international conference on big data (big data)* (pp. 171–179). [CrossRef]

[81] Yang, C., Liu, J., & Shi, C. (2021). Extract the knowledge of graph neural networks and go beyond it: An effective knowledge distillation framework. In *Proceedings of the web conference 2021* (pp. 1227–1237). [CrossRef]

[82] Chen, M., Wei, Z., Huang, Z., Ding, B., & Li, Y. (2020). Simple and deep graph convolutional networks. In *International conference on machine learning* (pp. 1725–1735).

[83] Irfan, R., King, C. K., Grages, D., Ewen, S., Khan, S. U., Madani, S. A., ... & Li, H. (2015). A survey on text mining in social networks. *The Knowledge Engineering Review, 30*(2), 157-170. [CrossRef]

[84] Stahl, F., Gaber, M. M., & Adedoyin-Olowe, M. (2014). A survey of data mining techniques for social network analysis. *Journal of Data Mining and Digital Humanities, 18*.

[85] Rehman, A., Paul, A., & Ahmad, A. (2020). A query based information search in an individual's small world of social internet of things. *Computer Communications, 163*, 176–185. [CrossRef]

[86] Rehman, A., Paul, A., Yaqub, M. A., & Rathore, M. M. U. (2020). Trustworthy intelligent industrial monitoring architecture for early event detection by exploiting social IoT. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing* (pp. 2163–2169). [CrossRef]

[87] Abdul, R., Paul, A., Gul M, J., Hong, W.-H., & Seo, H. (2018). Exploiting small world problems in a SIoT environment. *Energies, 11*(8), 2089. [CrossRef]

[88] Kalusivalingam, A. K., Sharma, A., Patel, N., & Singh, V. (2020). Enhancing Customer Relationship Management with Natural Language Processing: A Comparative Study of BERT and LSTM Algorithms. *International Journal of AI and ML, 1*(2).

[89] Dash, B., Swayamsiddha, S., & Ali, A. I. (2023). Evolving of smart banking with NLP and deep learning. In *Enabling Technologies for Effective Planning and Management in Sustainable Smart Cities* (pp. 151-172). Cham: Springer International Publishing. [CrossRef]

[90] Quân, T. M. (2024). A Strategic Analysis of AI-Driven Customer Relationship Management Systems in Enhancing Personalization and Retention in Financial Institutions. *Orient Journal of Emerging Paradigms in Artificial Intelligence and Autonomous Systems, 14*(9), 1-8.

[91] Ihsan, I., Rahman, H., Shaikh, A., Sulaiman, A., Rajab, K., & Rajab, A. (2023). Improving in-text citation reason extraction and classification using supervised machine learning techniques. *Computer Speech & Language, 82*, 101526. [CrossRef]

[92] Bhagat, S., Cormode, G., & Muthukrishnan, S. (2011). Node classification in social networks. In *Social network data analytics* (pp. 115-148). Boston, MA: Springer US. [CrossRef]

[93] Rong, Y., Huang, W., Xu, T., & Huang, J. (2019). Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903*.

[94] Nanayakkara, A. C., Kumara, B. T. G. S., & Rathnayaka, R. M. K. T. (2021). A survey of finding trends in data mining techniques for social media analysis. *Sri Lanka Journal of Social Sciences and Humanities, 1*(2). [CrossRef]

[95] Ergün, M. (2016). Using the techniques of data mining and text mining in educational research. *Participatory Educational Research, 4*(1), 140-151.

[96] Antons, D., Grünwald, E., Cichy, P., & Salge, T. O. (2020). The application of text mining methods in innovation research: current state, evolution patterns, and development priorities. *R&D Management, 50*(3), 329-351. [CrossRef]

[97] Manzoor, H. U., Khan, A. R., Sher, T., Ahmad, W., & Zoha, A. (2023). Defending Federated Learning from Backdoor Attacks: Anomaly-Aware FedAVG with Layer-Based Aggregation. In *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)* (pp. 1–6). [CrossRef]

[98] Ihsan, I., Imran, A., Sher, T., Al-Rawi, M. B. A., Elmeligy, M. A., & Pathan, M. S. (2024). Graph-Based COVID-19 Detection Using Conditional Generative Adversarial Network. *IEEE Access, 12*, 191323–191344. [CrossRef]

[99] Xu, Y., & Zhang, Y. (2023). Enhancement Economic System Based-Graph Neural Network in Stock Classification. *IEEE Access, 11*, 17956–17967. [CrossRef]

[100] Xiao, B. Y., Tseng, C. C., & Lee, S. L. (2022, July). A

Hypergraph Convolutional Neural Network for Node Classification of Citation Network Data. In *2022 IEEE International Conference on Consumer Electronics-Taiwan* (pp. 243-244). IEEE. [CrossRef]

[101] Li, B., & Pi, D. (2019). Learning deep neural networks for node classification. *Expert Systems with Applications*, *137*, 324–334. [CrossRef]

[102] Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, *22*(10), 1345-1359. [CrossRef]

[103] Antons, D., Grünwald, E., Cichy, P., & Salge, T. O. (2020). The application of text mining methods in innovation research: current state, evolution patterns, and development priorities. *R&D Management*, *50*(3), 329–351. [CrossRef]

[104] Sher, T., Rehman, A., Kim, D., & Ihsan, I. (2023). Exploiting Data Science for Measuring the Performance of Technology Stocks. *Computers, Materials and Continua*, *76*(3), 2979-2995. [CrossRef]

[105] Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855–864). [CrossRef]

Tahir Sher pursuing his PhD in Artificial Intelligence from Korea University, Seoul Campus, Korea. He done his MS in Data Science from Air University and bachelor's degree in Mathematics from the International Islamic University, Islamabad and achieved gold-medal. With strong teaching experience, he has taught subjects like Calculus, Differential Equations, Linear Algebra, and Data Science methods to students from Pakistan and allied countries. As a research scholar in the Explainable AI Group at Air University, his interests span IoT, Social IoT, Machine Learning, NLP, Time Series, Mathematical Modeling, Federated Learning, and Computer Vision. He is dedicated to bridging computer science with human-centered research. (Email: 2025010294@korea.ac.kr)

Abdul Rehman is currently a Research Professor at the Human Data Convergence Research Institute, Jeonju University, South Korea. He previously worked as a Postdoctoral Research Associate at the Hyper-Connectivity Convergence Technology Research Center, Kyungpook National University (KNU), Daegu, South Korea, and served as an Assistant Professor at the University of Central Punjab, Lahore, Pakistan. He holds a bachelor's degree in Mathematics from the International Islamic University, Islamabad, Pakistan, and an Integrated Ph.D. in Computer Science and Engineering from KNU. His research interests include Deep Learning, Machine Learning, Data Science, and their applications in Computer Vision, Smart IoT (S-IoT), and Data Analysis. He also explores Complex Network Navigation, Mathematical Modeling, and Big Data Analytics. Dr. Rehman received the KINGS Scholarship for his Ph.D. studies and was honored with the "Outstanding Researcher" Award from the School of Computer Science and Engineering at KNU. He serves as a guest editor and editorial board member for several international journals and has contributed to numerous international conferences as a session chair and publication chair. More information about Dr. A. Rehman is available at https://sites.google.com/view/drrehman/home.

Imran Ihsan received a PhD degree in Knowledge Graphs and Explainable AI from the Capital University of Science and Technology, Islamabad, in 2020. He has over 25 years of academic (teaching, project supervision, and academic administration) and industry (software, web, graphics, animation, and game development and operational management) experience. Since 2013, he has been working as Professor with the Department of Creative Technologies, Faculty of Computing and AI, Air University, Islamabad. He has authored or coauthored research work in various conferences and journals. His research interests include knowledge engineering, semantic computing, natural language processing, and computational linguistics.