GRAIL: Graph Retrieval-Augmented In-Context Learning for Node Classification in Real-World Textual-Attributed Graphs

Chanuk Lim
Korea Institute of Science and Technology Information
& Chungnam National University
Daejeon, South Korea
chanuklim@kisti.re.kr

Kyong-Ha Lee
Korea Institute of Science and Technology Information
Daejeon, South Korea
kyongha@kisti.re.kr

Hyun Ji Jeong
Kongju National University
Cheonan, South Korea
hjjeong@kongju.ac.kr

Sungsu Lim*
Chungnam National University
Daejeon, South Korea
sungsu@cnu.ac.kr

ABSTRACT

Textual-Attributed Graphs (TAGs) are gaining attention for their ability to structurally represent scientific and technological information, enabling diverse applications such as classification and recommendation systems. However, existing TAGs and Graph Neural Networks (GNNs) face significant challenges in integrating heterogeneous data sources and handling complex, real-world scenarios. To address these limitations, we propose a **K-GIST** (*KISTI Graph for the Integrated Scientific and Technological Domain*), a comprehensive graph that unifies academic papers, patents, and research projects to effectively represent diverse scientific and technological domains. We present a **GRAIL** (*Graph Retrieval-Augmented In-context Learning*), a framework that enhances node classification in K-GIST by integrating graph representation learning with Large Language Models (LLMs). GRAIL employs a two-phase process: embedding nodes with a graph neural network and retrieving the top-k relevant nodes for in-context learning. Experimental results demonstrate that GRAIL significantly improves multi-label classification accuracy by an average F1-score of 0.311, particularly in fine-grained and complex scenarios, outperforming baseline models. This study highlights the significant advancements achieved by integrating structural graph data with semantic inference, paving the way for innovative applications in scientific and technological information analysis.

1 INTRODUCTION

Graphs play a pivotal role in representing vast amounts of interconnected data, enabling a wide range of applications. In many real-world scenarios, these graphs include textual attributes—often referred to as Textual-Attributed Graphs (TAGs)—which combine structural graph information with semantic insights from language models. Consequently, TAGs are useful in diverse domains, including patent knowledge graphs, academic paper citations, social networks, e-commerce, and recommendation systems [22]. In the scientific and technological domain, researchers have constructed TAGs from scientific papers and patents [4, 8, 16, 23, 30, 31, 31, 32, 40, 43], with patent-focused graphs especially useful for search and recommendation tasks [27, 39, 47]. By incorporating textual semantics, these TAGs capture more intricate relationships, thereby linking patents and other entities in more meaningful ways. In parallel, recent Graph Neural Networks (GNNs) are designed to learn graph representations by combining text features extracted from each node of TAGs with structural information. Advancements in language models have further enhanced the performance of these GNN-based approaches [41, 42, 44, 46], demonstrating notable improvements in key tasks such as node classification [7, 37, 45].

Despite this widespread impact on the industry, existing TAGs and GNNs have two limitations. First, while real-world scientific and technological information include various contents such as academic papers, patents, reports, and R&D projects, existing TAGs do not integrate this diverse information into a cohesive graph. In other words, existing TAGs focus on constructing graphs using mainly homogeneous content without integrating such heterogeneous information. Table 1 provides a comparison of existing graphs within the scientific and technological domain. Despite considerable efforts to model literature, projects, authors, and other components as entities and to establish connections among them, the range of relationships used to link these entities remains limited. The simplicity of these graphs often limits their ability to capture the complexity and nuance of real-world relationships and data.

Moreover, in complex graphs reflecting the real world, GNNs face limitations in node classification performance. In the real world, document categorization is hierarchical and exhibits characteristics of multi-class, multi-label classification. For instance, the International Patent Classification (IPC) system categorizes patents into detailed levels such as sections, classes, and subclasses, with a single patent potentially assigned multiple IPC codes. However, the existing graphs in Table 1 are not designed to take these characteristics into account. Classifying numerous labels becomes a highly challenging problem for models, especially as the semantic similarity between labels increases [21]. For example, when patent examiners review a new patent, they must examine similar patents and related academic literature, and technical reports to assess its novelty and inventive step. Existing graph-based approaches, however, fail to provide meaningful support for this process. Their inability to integrate heterogeneous data sources and represent the intricate relationships between patents, academic literature, and technical documents renders them ineffective in aiding such tasks. Without significant improvements in graph construction and classification, these approaches remain unsuitable for addressing the complex requirements of real-world patent examination.

*Corresponding author.

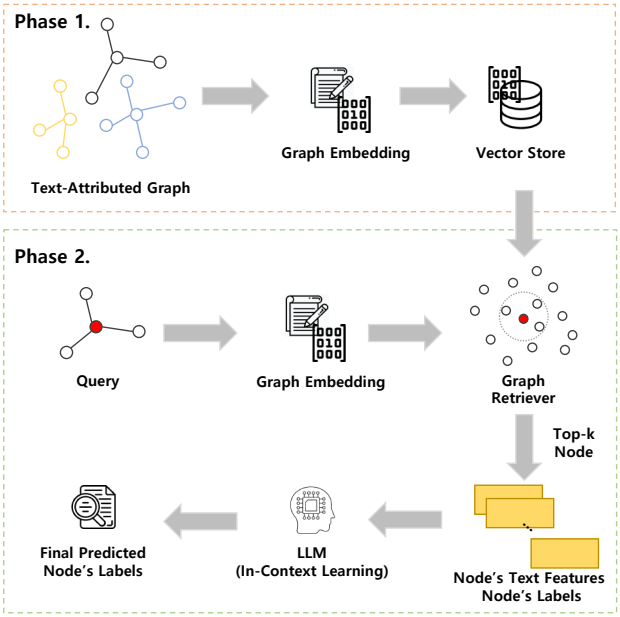


Figure 1: The proposed GRAIL model. Node embeddings are stored in a vector database, and the top-k similar embeddings are retrieved as context to enable the LLM to infer node labels via in-context learning, enhancing classification in K-GIST.

To address these issues, we propose new TAGs and a node classification method specifically for the scientific and technological domain. Our contributions are as follows:

(i) We propose a ScienceON Graph System for constructing a **K-GIST** (*KISTI Graph for the Integrated Scientific and Technological Domain*), a type of TAGs, designed to build a comprehensive and systematic graph to tackle the challenges of data-driven analysis within real-world scientific and technological domain. Notably, the data fueling this system is supported by the government and curated by the Korea Institute of Science and Technology Information (KISTI), offering a wealth of knowledge spanning all scientific and technological fields. Additionally, the system includes an automated pipeline for the seamless update and expansion of scientific and technological information.

(ii) To improve node classification in the proposed real-world TAGs, we introduce a **GRAIL** (*Graph Retrieval-Augmented In-context Learning*) model. We utilize a graph embedding model as a retriever to extract k nodes similar to the target node and feed the textual information of these nodes to Large Language Models (LLMs). This method has the advantage of simultaneously leveraging the graph representation capabilities of the graph embedding model and the semantic inference capabilities of LLMs. GRAIL addresses the challenges of computational inefficiency and high label cardinality in multi-class, multi-label classification by first retrieving candidate nodes through graph embeddings and then leveraging LLMs' linguistic inference capabilities to enhance node classification performance.

(iii) We present various applications of K-GIST, providing evidence for integrated analysis of scientific and technological information that existing TAGs could not achieve. Furthermore, we conduct various experiments to analyze the proposed GRAIL model. Experimental results demonstrate that GRAIL outperforms baseline models, achieving an average F1-score improvement of 0.311.

2 MOTIVATION AND BACKGROUND

The scattered distribution of scientific and technological information across disparate systems and repositories poses significant challenges. KISTI, under Article 40, CHAPTER V, of the Enforcement Decree of the Framework Act on Science and Technology, plays a crucial role in the comprehensive collection of domestic and international scientific and technological information, supported by government funding. This enables the systematic construction of a wide array of scientific and technological information and continuous efforts in data discovery. The scientific and technological information used in this study is collected under national policies and forms the foundation of our research, with KISTI collaborating with domestic and international institutions—acquiring academic data through various academic organizations and patent data via the Korea Institute of Patent Information (KIPI)—and retaining the rights to replicate, distribute, and transmit the collected data.

At KISTI, the metadata collected from academic and R&D projects undergo a manual verification process. In addition, when service users submit feedback regarding any data issues, system operators proceed to review the data. Most importantly, as public data, they must adhere to government database standardization guidelines to ensure standardization across databases and are validated by external independent audits. This verification process enables the data constructed by KISTI to serve as a gold standard in the field of science and technology.

ScienceON[6] is KISTI's flagship service, offering open data through Web and REST APIs for students, researchers, and policymakers, and promoting effective information delivery and technological collaboration. It serves as a data ecosystem that integrates data collected by KISTI, employing an ETL process to continuously aggregate large volumes of data from both internal and external sources. As shown in Table 2, ScienceON has collected 170 million records related to scientific and technological information (as of November 2024). As shown in Table 3, the scientific and technological information provided by KISTI through ScienceON has steadily increased, starting from 100 million calls in 2020 and surpassing 200 million calls by 2023.

In summary, given the data accumulated over several decades and recent user trends, its importance becomes evident. Moreover, constructing graphs based on this data can significantly enhance existing systems and unlock substantial potential for further applications. Recognizing these opportunities, K-GIST and GRAIL were developed in alignment with national R&D policies to address real-world needs, such as bibliographic data classification, document search, expert identification, and collaborative network analysis.

3 RELATED WORK

3.1 Graphs in scientific and technological domain

Research on constructing and refining graphs in the scientific and technological domain has been active, leading to numerous applications. Microsoft Academic Graph (MAG) [31] and Aminer [32]

[1] https://relational.fit.cvut.cz/dataset/CORA
[2] https://relational.fit.cvut.cz/dataset/citeseer
[3] https://snap.stanford.edu/data/cit-Patents.html
[4] https://www.helsinki.fi/en/researchgroups/unified-database-management-systems-udbms/datasets/patent-dataset
[5] https://www.microsoft.com/en-us/research/project/open-academic-graph/
[6] https://scienceon.kisti.re.kr/

Table 1: Comparison of Graphs in the Scientific and Technological Domain.

Graph	Entity Type	Relationship Type
Cora[1][28]	Paper(P)	P-P
CiteSeer[2][28]	Paper(P)	P-P
cit-Patents Graph (SNAP)[3]	Patent(P)	P-P
Unified Database Management Systems(UDBMS)[4]	Patent(P), Inventor(I), Assignee(A), Class(C), Category(T)	P-P, P-A, P-C, P-I, C-T
PubMed KG[40]	Article(A), Author(Au), Affiliation(Af), Funding(F), Project(P)	A-A, A-P, A-Au, Au-Af, Au-F, F-P
Open Academic Graph(OAG)[5]	Paper(P), Author(A), Affiliation(Af), Venue(V)	P-P, P-A, P-Af, A-Af, P-V
AIDA[2]	Paper(P), Author(A), Affiliation(Af), industrialSector(S), DBpediaCategory(C), Topic(T), Patent(Pt)	P-A, A-Af, P-Af, P-S, P-C, Af-S, Af-C, P-T, Pt-T, Pt-S, Pt-C
K-GIST	Top-Project(T), Sub-Project(S), Author(A), Organization(O), Paper(P), Journal(J), PaperCategory(PC), Patent(Pt), IPC(I), Report(R), ReportCategory(RC), Keyword(K)	T-S, S-P, S-R, S-Pt, P-J, P-P, P-A, P-PC R-P, R-R, R-RC, R-A, Pt-Pt, Pt-I, Pt-A A-O, P-K, R-K

Table 2: Record Counts in Various Categories of Scientific and Technological Information.

Category	Description	Number of Records
Domestic Papers	Searchable papers published in domestic journals and conferences.	Journal Articles: 3,661,247 Conference Papers: 401,853 Thesis: 1,352,910
International Papers	Searchable papers published in international journals and proceedings.	Journal Articles: 113,501,156 Conference Papers: 12,711,259
Patents	Patent information for patents registered and disclosed in various countries, including Korea, the U.S., Europe, and Japan.	Korea: 6,749,003, U.S.: 16,905,092, Europe: 5,317,913 Japan: 11,573,913, PCT : 5,863,505
Reports	Analytical reports produced through national R&D projects and various institutes.	National R&D Report: 359,785 Various Analysis Reports: 68,990
Trends	Provides the latest trends and issues in major scientific and technological fields globally.	Various Trends Reports: 67,908
Researchers	Lists of researchers identified in domestic papers, reports, and patents.	Researchers: 917,425

Table 3: User Calls Statistics of Scientific and Technological Information on ScienceON.

Category	2020	2021	2022	2023
Papers	47,993,208	100,773,457	99,981,733	159,659,715
Patents	45,965,731	15,114,090	23,250,042	41,735,819
Reports	3,620,827	12,685,573	7,517,101	24,789,015
Trends	3,647,181	3,999,092	3,026,813	13,255,091
Researchers	2,711,490	1,224,326	1,681,298	236,392
Total	103,938,437	133,796,538	135,456,987	239,676,032

are prominent examples, integrating heterogeneous entities like papers, patents, authors, and affiliations to create comprehensive scholarly graphs. Open Academic Graph (OAG) [43] further combines MAG and Aminer to build a large-scale linked entity graph. Non-profit organizations like OpenCitations [23] and OpenAIRE [19] provide open scholarly information and infrastructure, utilizing semantic web technologies and hosting various scientific outputs. Analyzing documents from both academic and industrial sectors together offers deeper insights than evaluating them individually [1, 3, 11, 14, 17, 20, 26]. Studies have examined academy-industry relationships and their influence on higher education [1], complementary knowledge transfers [11], collaboration trends [14], shifts in basic research [17], and the outcomes of academic-industrial collaborations [3]. Methods combining semantic technologies with machine learning have been proposed to quantify research trends in both spheres [26], and the influence of non-academic and industrial publications has been assessed [20].

3.2 Graph Representation

Graph representation aims to structure information so computers can comprehend and utilize real-world knowledge for complex problem-solving within computational limitations. Graph Neural Networks (GNNs) have emerged as powerful tools for learning graph representations, surpassing traditional models [38]. GNNs use propagation modules involving aggregation and update functions to learn representations of nodes, edges, and graphs by integrating feature vectors from neighbors. They aim to convert graphs into low-dimensional vectors for downstream tasks at node, edge, and graph levels. GNNs with convolution operators are categorized into spectral-based and spatial-based methods [5]. Spectral-based GNNs, like Spectral GCNs [6], define convolution in the Fourier domain but are limited to transductive settings. Spatial-based GNNs, such as GraphSAGE [12], address this by aggregating features from local neighborhoods in the graph domain. Attention mechanisms have also been incorporated, with models like GAT [35] applying self-attention to assign different weights to neighbors [34]. However, these methods focus on homogeneous graphs and cannot capture the complexity of real-world heterogeneous graphs. To model different types of nodes and edges, frameworks like HAN [36] and HGT [13] have been developed; HAN leverages meta-path-based neighbors, while HGT introduces heterogeneous mutual attention and message passing to learn type-dependent representations.

4 DEVELOPMENT OF K-GIST: KISTI GRAPH FOR INTEGRATED SCIENTIFIC AND TECHNOLOGICAL DOMAIN

4.1 Data Collection and Integration in the ScienceON Graph System for K-GIST

We present the ScienceON Graph System for K-GIST, which is depicted in Figure 2, a comprehensive data ecosystem that integrates scientific and technological information from various legacy systems, with capabilities to collect a wide range of data types, including papers, journals, authors, organizations, patents, research reports, and R&D projects. It aggregates extensive scientific and technological information from multiple legacy systems and web sources. This comprehensive approach allows ScienceON to serve as a pivotal resource, enabling researchers and practitioners to access a rich repository of scientific and technological information, thereby facilitating advanced research and development activities across various domains.

KISTI provides data files covering papers, journals, authors, affiliations, and more. **Journal/Conference article** has a unique ID and attributes like title, authors, affiliations, journal, publication date, DOI, and abstract. Citation relationships link referencing and referenced papers through unique IDs. **Author & Organization** includes author and affiliation details for papers, patents, and reports. Identifying authors and affiliations is challenging due to homonyms. Seol et al. [29] proposed a method for this issue. Each author and affiliation has a unique ID, with names as attributes, and authors are linked to ISNI and ORCID for better extensibility.

NTIS [7] provides information on **National R&D projects and reports**, including attributes such as project title, abstract, and duration. Crucially, the metadata concerning R&D projects from NTIS forms the backbone of the graph, enabling the interconnection of disparate scientific and technological information. Researchers produce various R&D deliverables, including papers, patents, and reports. These deliverables, submitted by the researchers, are manually correlated by operators with the unique IDs of scientific literature previously assigned by KISTI, ensuring accurate mapping with the corresponding research projects. This hands-on approach guarantees that each output such as papers, patents, and reports is meticulously linked to its relevant project, enhancing the integrity and utility of the graph.

Patent data from KIPRIS [8], provided by the KIPI includes extensive patent records. Each patent record is uniquely identified and detailed with information such as the patent title, inventor(s), affiliations, and abstract. Additionally, patents are categorized according to the International Patent Classification (IPC) system, allowing for the organization of patents into specific fields of invention. With an approximate count of 47 million patents, KIPRIS serves as a vital resource for accessing a wide range of patents filed or registered worldwide, thereby supporting innovation and research by offering insights into existing patents and their classifications.

Document classifications, including those for papers and patents, are acquired through web crawling. The academic paper data compiled in ScienceON are invariably linked to a journal. However, these journals lack a taxonomy of scientific concepts. To address this, we perform web crawling on Google Scholar's

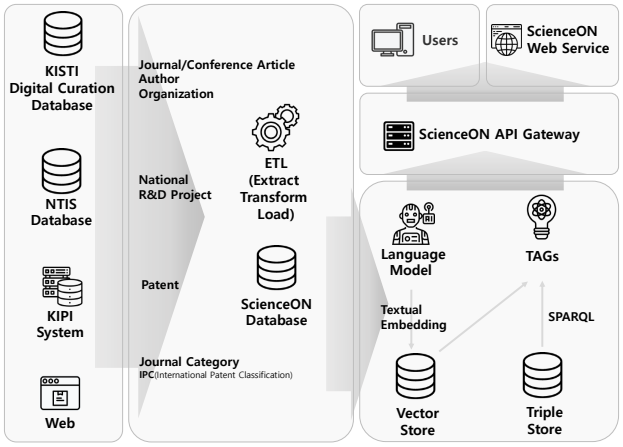


Figure 2: The ScienceON Graph System's architecture depicts the integration and processing of scientific and technological information. From multiple data sources to the utilization of an ETL pipeline for data aggregation and the building of K-GIST via RDF, SPARQL, and language models, this system facilitates the structured dissemination and in-depth analysis of scientific and technological information.

journal categories [9]. The patent data from KIPRIS is mapped to IPC but lacks descriptive information, which we supplement by crawling explanatory details from the World Intellectual Property Organization (WIPO) [10]. Similarly, National R&D data are categorized using the Korea National Science and Technology Standards Classification Codes by KISTEP[11]. The categories are designed in a way that allows an article to correspond to multiple categories, enabling a one-to-many (1:n) mapping. Furthermore, the categories follow a two-level hierarchical structure, effectively mirroring the complexity of the real world.

4.2 Construction of K-GIST Ontology

K-GIST is meticulously engineered to elucidate the complex interconnections among various scientific and technological information. It achieves this by defining a comprehensive network of entities and their interrelations, as depicted in Figure 3. In this model, entities are not isolated data points but are intricately connected to represent comprehensive knowledge in science and technology. The ontology covers a variety of entities, including R&D projects, papers, journals, patents, reports, authors, institutions, keywords, and categories. Each node is assigned a unique ID that has been verified by human experts, and nodes are linked according to their relationships within the network.

To construct this ontology, we utilize Protégé [12], a premier tool for ontology modeling that facilitates complex designs. Based on the developed ontology, the graph's structure is explicitly modeled using the Resource Description Framework (RDF), a World Wide Web Consortium (W3C) standard. To ensure data integrity, we validate the RDF data using the W3C Shapes Constraint Language (SHACL) [13], thereby enhancing the credibility

[7] https://www.ntis.go.kr/
[8] http://www.kipris.or.kr/
[9] https://scholar.google.com/citations?view_op=top_venues&hl=en
[10] https://www.wipo.int/classifications/ipc/en/
[11] https://www.kistep.re.kr/
[12] https://protege.stanford.edu/
[13] https://www.w3.org/TR/shacl/

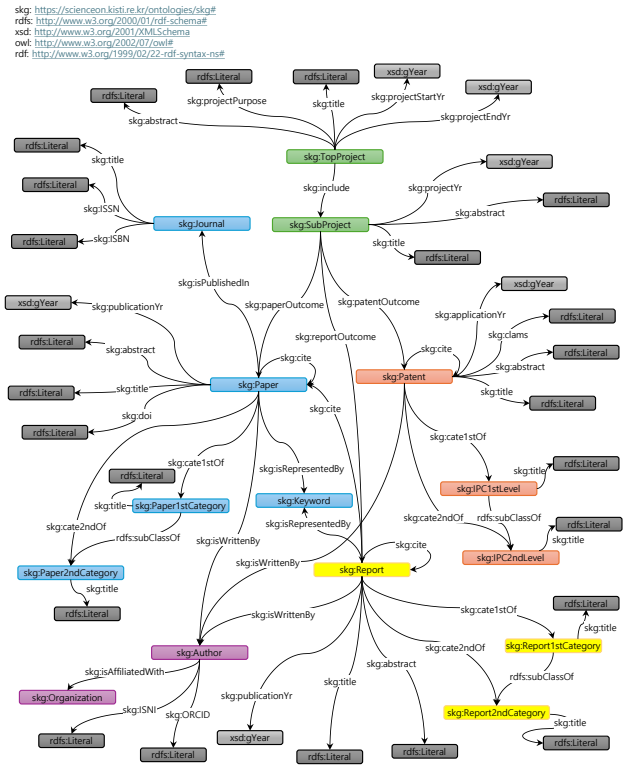


Figure 3: K-GIST ontology illustrating the intricate network of entities and their interrelationships representing scientific and technological information.

of the information. Additionally, we transform relational database content into RDF format using R2RML mappings, enabling precise alignment with the ontology. This step is crucial for converting structured relational data into RDF triples that populate the graph. Once validated through SHACL, K-GIST is stored in a triple store, enabling sophisticated query capabilities via SPARQL [14]. For this purpose, we selected GraphDB for its exemplary performance as both a SPARQL engine and triple store. This approach semantically links stored data, enabling advanced analyses to uncover hidden patterns and drive innovation across scientific and technological domains.

5 DEVELOPMENT OF GRAIL: GRAPH RETRIEVAL-AUGMENTED IN-CONTEXT LEARNING

We propose a GRAIL model to enhance node classification, addressing the limitations of existing graph embedding models in performing node classification on TAGs that reflect real-world scenarios. This model combines the graph structural information provided by the graph representation learning model with the semantic reasoning capabilities of the LLM, as illustrated in Figure 1. The proposed method consists of a two-phase approach: **(i)** A graph embedding model converts all nodes into embedding vectors, which are then stored in a vector store. **(ii)** When a query for a node classification task is made, the graph retriever calculates the similarity of embedding vectors stored in the vector store, extracts the top-k results, and uses them as context, enabling the LLM to infer the node's label using an in-context learning approach.

Phase 1. We employ the GNNs framework, which has recently demonstrated superior performance among various graph representation learning models, to learn K-GIST. A graph, denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, consists of a set of nodes $\mathcal{V}$ and a set of edges $\mathcal{E}$ with node feature vector $\mathcal{X}_v$ for $v \in \mathcal{V}$. A general GNN framework composes two operations of the aggregation function and the update function to learn a node representation vector, denoted as h_v.

$$Aggregation : a_v^{(\ell)} = \text{Aggregate}^{(\ell)}(h_u^{(\ell-1)}, \forall u \in \mathcal{N}_v), \quad (1)$$

$$Update : h_v^{(\ell)} = \text{Update}^{(\ell)}(h_v^{(\ell-1)}, a_v^{\ell}), \quad (2)$$

where $\mathcal{N}_v$ is the neighborhood of the node v, $h_u^{(\ell-1)}$ is the feature vector of node v at the ℓ-th layer of a GNN.

K-GIST is designed to realistically reflect the scientific and technological domain, resulting in nodes having multiple labels. Since most existing GNN frameworks use loss functions for single-label problems, we need to define a loss function to solve multi-class and multi-label problems. For graph representation learning with multi-class, multi-label, we can optimize the GNN model by binary cross entropy loss with combing a sigmoid layer.

$$L(x) = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \sigma(h_i) + (1 - y_i) \log(1 - \sigma(h_i)), \quad (3)$$

where N is the number of training nodes, h is an embedding vector of training nodes, σ is a sigmoid function and y is a label vector of training nodes. Once the GNN model training is complete, it generates embedding vectors for all nodes. These vectors, along with metadata such as each node's ID, title, and abstract, are stored in a vector database for retrieving in Phase 2.

Phase 2. First, when a query $\mathbf{q}$ is provided as input, it is transformed into an embedding vector $\mathbf{v}_q$ by the GNN model GNN trained in Phase 1.

$$\mathbf{v}_q = GNN_{\text{embed}}(q) \quad (4)$$

Specifically, GNN_{embed} is the embedding function of the GNN model G, mapping the query $\mathbf{q}$ into the vector space $\mathbf{v}_q$.

In the retrieval phase, the similarity between the query embedding $\mathbf{v}_q$ and the graph embeddings $\mathbf{v}_{d_i}$ stored in the vector store during Phase 1 is calculated. The similarity between the query embedding $\mathbf{v}_q$ and the graph embeddings $\mathbf{v}_{d_i}$ is calculated using the dot product. The graph embeddings $\mathbf{v}_{d_i}$ stored in the vector store are indexed using HNSW (Hierarchical Navigable Small World) [18], which improves retrieval speed by enabling efficient nearest neighbor search in high-dimensional spaces.

Using this retriever, the top-k nodes most relevant to the query $\mathbf{q}$ are retrieved as $\mathbf{D} = \{d_1, d_2, \ldots, d_k\}$.

$$\mathbf{D} = \text{Retriever}_k(\mathbf{v}_q, \{\mathbf{v}_{d_i}\}) \quad (5)$$

Here, $\{\mathbf{v}_{d_i}\}$ denotes the set of all node embedding vectors in the database, and $\mathbf{D}$ represents the set of top-k documents most similar to the query embedding $\mathbf{v}_q$.

Retrieving the top-k nodes ensures that the system focuses on the most relevant candidates, balancing computational efficiency and response quality. By limiting the retrieval to a fixed number k, the computational overhead in the subsequent generator is reduced, while still maintaining a high likelihood of including the most relevant information.

[14] https://www.w3.org/TR/sparql11-query/

Using the retrieved node set $\mathbf{D}$, the generative model **LLM** generates a response $\mathbf{a}$ for the query $\mathbf{q}$.

$$a = \mathbf{LLM}(\mathbf{v}_q, \mathbf{D}) \tag{6}$$

Each query corresponds to a graph node with associated metadata, and the prompt consists of the query along with the top-k retrieved nodes. By incorporating the retrieved node set $\mathbf{D}$ into the generation process, the model **LLM** can leverage both the query embedding $\mathbf{v}_q$ and structural information from the retrieved graph embeddings $\mathbf{D}$ to generate a more precise and information-rich response. This combination mitigates the limitation of standalone generative models, which rely solely on pre-trained knowledge and may lack domain-specific context.

The generative model **LLM** probabilistically infers the answer based on the query $\mathbf{q}$ and the retrieval results.

$$P(a|\mathbf{v}_q) = \sum_{d \in \mathbf{D}} P(a|\mathbf{v}_q, d) \cdot P(d|\mathbf{v}_q) \tag{7}$$

Here, $P(a|\mathbf{v}_q, d)$ denotes the probability of generating the response a given the selected document d and the query embedding $\mathbf{v}_q$, and $P(d|\mathbf{v}_q)$ represents the probability of selecting the document d given the query embedding $\mathbf{v}_q$.

In conclusion, the proposed framework effectively balances computational efficiency and response quality by retrieving a fixed number of top-k nodes. This approach reduces computational overhead while maintaining high relevance and provides a foundation for integrating the retrieved node set $\mathbf{D}$ into the generative process. By leveraging contextual accuracy from the query embeddings and structural insights from the retrieved graph embeddings, the framework ensures responses are both semantically meaningful and aligned with the underlying graph structure. Additionally, the probabilistic formulation $P(a|\mathbf{v}_q)$ enhances robustness by addressing uncertainty in retrieval and generation, improving system reliability and adaptability.

6 EXPERIMENTS

We present experiments to qualitatively and quantitatively verify the benefits of K-GIST, which reflects real-world contexts in the scientific and technological domain. Furthermore, we evaluate whether the GRAIL model contributes to performance improvement in the node classification task. Our investigation addresses the following five research questions:

- **Q1.** Does K-GIST, by leveraging relationships among heterogeneous data, provide value to users?
- **Q2.** Does the structural information in K-GIST enhance performance in the node classification task?
- **Q3.** Are existing graph representations suitable for the node classification task within K-GIST?
- **Q4.** Does the GRAIL model improve performance in the node classification task?

6.1 Experimental Setup

In our investigation of K-GIST, we selectively extracted data to train graph representation learning models and conduct experiments on various tasks. To ensure a well-structured and meaningful graph, we first removed irrelevant nodes, such as unconnected or small, disconnected components. We then applied extraction criteria focused on national R&D projects conducted between 2011 and 2020, prioritizing those with high research impact and strong alignment with our study objectives. The number of primary entities extracted is presented in Table 4.

Table 4: Extracted Entity Counts for K-GIST.

Entity Type	Count	Notes
Top-Projects	36,230	Unique projects for multi-year R&D initiatives
Sub-Projects	64,883	Annual project IDs associated with Top-Projects
Papers	360,612	International & domestic journal
Patents	102,703	International & domestic published patent application
Reports	21,487	Outputs of national R&D projects
Researchers	70,435	-
Affiliations	2,460	-

Table 5: Category Distribution of Entities.

Entity Type	1st-Level Categories	2nd-Level Categories	Remarks
Paper	8	100	Multi-label
Patent	54	139	Multi-label
Report	17	88	Multi-class

To enable a nuanced classification aligned with the intricate structure of scientific research, we categorized our entities into primary and secondary categories based on their thematic and technical relevance. The distribution of these categories is presented in Table 5.

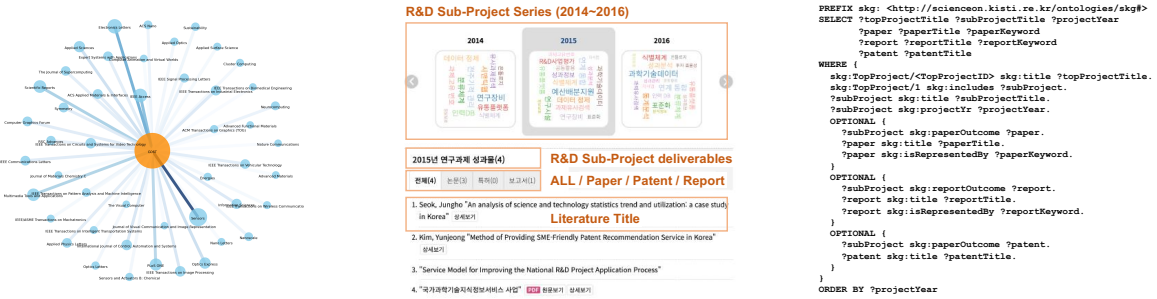
We utilized the following graph representation learning models: GCN [15], GraphSAGE [12], GAT [35] for homogeneous graphs, and Metapath2vec [9], HAN [36], HGT [13] for heterogeneous graphs. Most hyperparameters were adopted from existing GNN literature to ensure consistency and comparability. We utilize trained graph representation learning to transform all nodes into embedding vectors, which are then stored in a vector store implemented using Chroma [15]. For vector similarity search, we employ the HNSW [18] index and calculate distances using the squared L2 norm. For in-context learning, we utilized state-of-the-art large language models (LLMs), such as GPT-4o. These models were chosen to evaluate the complementary capabilities of graph representation learning and large language models in handling complex and large-scale datasets, particularly in tasks requiring both structured and unstructured data integration. All graph representation learning models models were implemented using the PyTorch Geometric (PyG) package [10].

6.2 Q1. Value of Heterogeneous Relationships

We developed K-GIST to infer relationships among heterogeneous content that traditional graphs cannot capture, supporting tasks like helping patent examiners classify journals related to specific patents. By analyzing co-occurrence frequencies between node types (e.g., IPC and journal nodes), we measured how often a target node (e.g., a journal) appears alongside a source node (e.g., an IPC). Higher co-occurrence frequencies indicate stronger relational closeness, as illustrated in Figure 4a, where link thickness represents the degree of proximity. For example, the IPC "Image data processing or generation" (G06T) strongly connects to journals such as “Sensors,” “Electronics Letters,” and “IEEE Access,” highlighting thematic commonalities between patents and academic papers.

Additionally, through K-GIST, we leverage relationships between entities and classify content using node classification to develop applications that provide access to heterogeneous scientific and technological literature. These applications are made available to the public via the ScienceON web service. For example, one functionality retrieves national R&D projects by year

[15] https://github.com/chroma-core/chroma



(a) (b) (c)

Figure 4: Various applications based on K-GIST: (a) Frequency of co-occurrence, (b) Timeline visualization service of national R&D projects on ScienceON, (c) Example SPARQL query for the service.

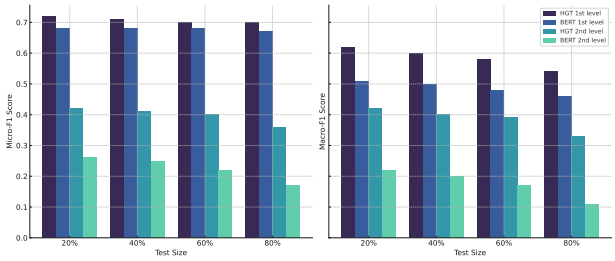


Figure 5: Comparison of document classification performance with and without K-GIST on the 1st-level categories of papers.

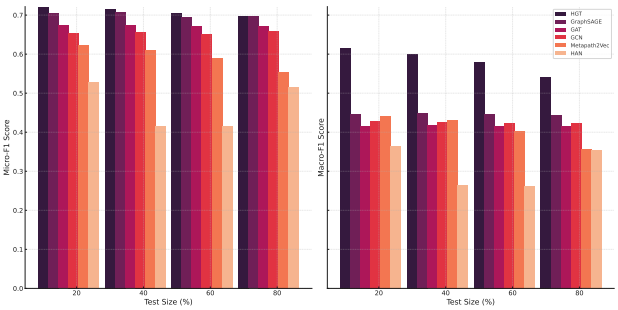
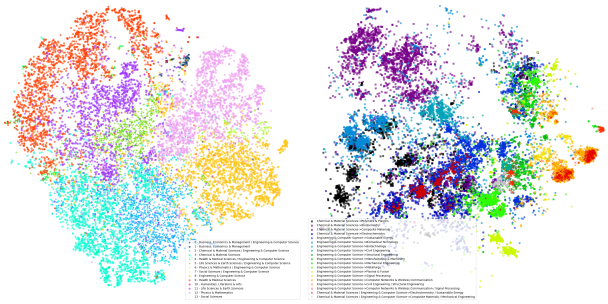


Figure 6: Node classification performance on the 1st-level categories of papers.

and displays their resulting outputs. Figure 4b illustrates projects from 2014 to 2016, along with their annually published papers, patents, and reports in a timeline format. Another feature, depicted in Figure 4c, uses a SPARQL query to locate all Sub-Projects under a Top-Project and retrieve the research outputs produced by these Sub-Projects.

6.3 Q2. Impact of Structural Information

To assess K-GIST's contribution to node classification tasks, we evaluated its impact on document classification efficacy. The comparison of paper classification performance, both with and without K-GIST integration, is illustrated in Figure 5. We employed two distinct models for this experiment: the baseline sentenceBERT [24, 25], and HGT [13], which utilizes sentenceBERT's text embeddings as input features. The baseline sentenceBERT model relies solely on abstract text for embedding and uses logistic



(a) (b)

Figure 7: Visualization of graph embeddings: (a) Embeddings labeled with the paper's 1st-level categories, (b) Embeddings labeled with the paper's 2nd-level categories.

regression for classification. In contrast, HGT enhances this approach by incorporating graph-structured data as an additional feature set alongside the textual information.

The evaluation, depicted in Figure 5, benchmarks the F1 score across different test sizes. Across all test sizes, HGT consistently outperforms sentenceBERT, with a particularly significant performance gap observed in the paper 2nd-level category, where classification granularity is higher. This demonstrates that the integration of graph information provides a substantial performance advantage, highlighting the value of incorporating K-GIST into the model. These results underscore the advantage of integrating graph-structured data through K-GIST, enabling HGT to achieve superior predictive accuracy over the text-only baseline.

6.4 Q3. Suitability of Existing Graph Representations

We report the F1 scores across classification tasks. HGT consistently outperforms other baselines in all node classification tasks. The results, detailed in Figure 6, illustrate the node classification performance when categorizing paper nodes into their 1st-level categories. We present experimental results only for node classification in the paper's 1st-level categories, as the F1 scores of methods other than HGT converge to nearly 0 in more complex node classification tasks, making meaningful comparisons infeasible.

For an intuitive understanding of the embedding generated by HGT, we employed t-distributed Stochastic Neighbor Embedding (t-SNE) [33]. Figure 7a shows the node embeddings labeled

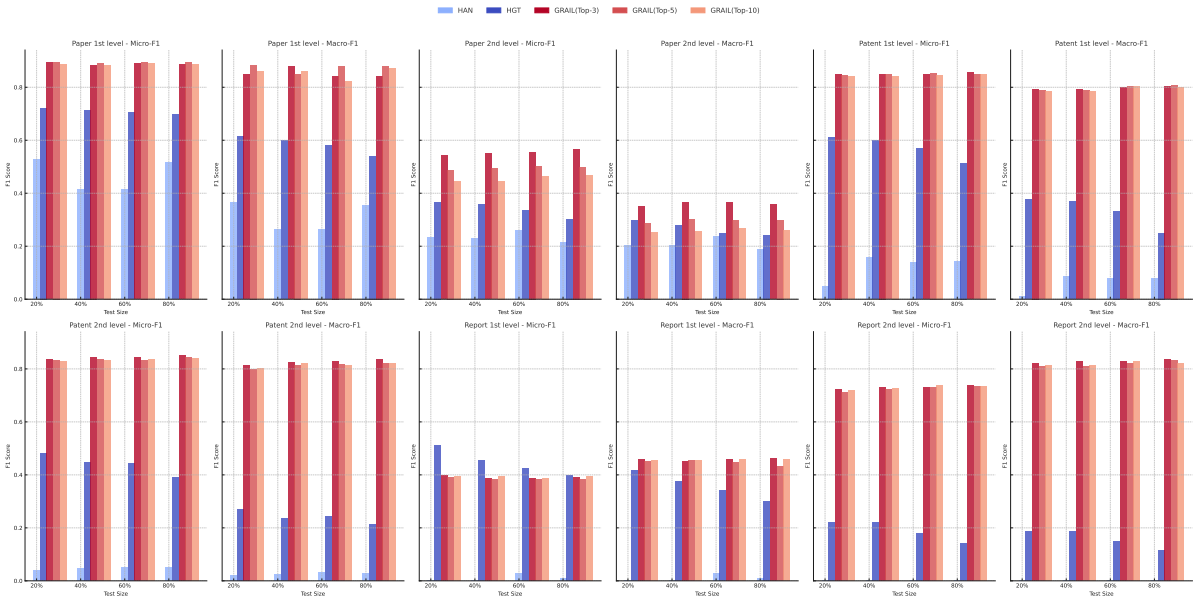


Figure 8: Node classification performance on all tasks.

with the paper 1st-level categories. Since the paper 1st-level task involves only eight labels, it is relatively simple, and the embedding visualization shows clearly separated regions for each label. In contrast, Figure 7b also presents node embeddings labeled with paper 2nd-level categories, where the number of labels is significantly larger and the classification task is more complex. Consequently, overlapping regions between labels are more frequently observed in the visualization, reflecting the challenges of distinguishing between a significantly larger number of categories in a more complex classification task.

In conclusion, existing graph embedding models perform well on relatively simple node classification tasks but struggle to perform effectively in real-world TAGs. This highlights the need for more advanced node classification techniques capable of handling the complexity and diversity inherent in real-world scenarios. Therefore, we confirm the limitations of relying solely on graph structure for node classification performance.

6.5 Q4. Effectiveness of the GRAIL

We present the performance of GRAIL and GNNs across all node classification tasks. The F1 score is measured for classifying the 1st-level and 2nd-level categories of all entities (papers, patents, and reports). As shown in Figure 8, GRAIL, enhanced with LLM, consistently achieves higher performance compared to traditional GNNs. The average F1 score of GRAIL (Top-3) is 0.698, while the average F1 score of HGT is 0.387, demonstrating an improvement of 0.311. This improvement is particularly notable in the patent classification task, where GRAIL shows a distinct advantage by effectively utilizing the textual features of the target node and the detailed IPC descriptions to enhance node classification performance.

Moreover, the improvement in classification performance for 2nd-level categories is more significant than for 1st-level categories. The 2nd-level categories, such as "Artificial Intelligence" or "Signal Processing" under "Engineering & Computer Science," involve finer-grained distinctions that rely heavily on contextual and descriptive textual information. As a result, text features contribute more significantly to classification performance than graph structure in these detailed categories. However, we still observed challenges in classifying nodes with sparse labels, likely due to the limited availability of training data.

Additionally, we examined the effect of top-k retrieval settings on performance when using a graph retriever to identify classification candidates and reasoning with the LLM. As shown in the Figure 8, the highest performance is observed with top-3 retrieval, but the difference between top-3 and top-10 is minimal, at approximately 0.02. This indicates that while top-3 is optimal, expanding to top-10 does not significantly degrade performance.

7 CONCLUSION

In this paper, we constructed K-GIST, a TAG that reflects real-world scenarios in the scientific and technological domain, and proposed GRAIL, a novel method for improving node classification performance on the graph. K-GIST integrates diverse scientific and technological information from KISTI, incorporating multiple stages of validation and standardization to ensure future usability. GRAIL significantly improves node classification on K-GIST by leveraging GNNs as retrievers to enhance the efficiency and inference capabilities of LLMs. Extensive experiments identified limitations in existing graph representation learning models and demonstrated that our method outperforms state-of-the-art baselines. By providing integrated information services for stakeholders—including students, researchers, and policymakers—our method facilitates more efficient knowledge discovery and decision-making. Furthermore, GRAIL is expected to be effective for context-based, complex classification in other domains.

ACKNOWLEDGMENTS

This research was supported by Korea Institute of Science and Technology Information (KISTI) (No. K25L1M1C1), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00214065, RS-2023-00278009).

REFERENCES

[1] Melissa S. Anderson. 2001. The Complex Relations between the Academy and Industry. The Journal of Higher Education 72, 2 (2001), 226–246.

[2] Simone Angioni, Angelo Salatino, Francesco Osborne, Diego Reforgiato Recupero, and Enrico Motta. 2021. AIDA: A knowledge graph about research dynamics in academia and industry. Quantitative Science Studies 2, 4 (2021), 1356–1398.

[3] Michaël Bikard, Keyvan Vakili, and Florenta Teodoridis. 2019. When collaboration bridges institutions: The impact of university-industry collaboration on academic productivity. Organization Science 30, 2 (2019), 426–445.

[4] Sebastian Björkqvist and Juho Kallio. 2023. Building a Graph-Based Patent Search Engine. In SIGIR. 3300–3304.

[5] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. 2017. Geometric Deep Learning: Going beyond Euclidean data. IEEE Signal Processing Magazine 34, 4 (2017), 18–42.

[6] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. 2014. Spectral Networks and Locally Connected Networks on Graphs. In ICLR.

[7] Sadaf Charkhabi, Peyman Samimi, Sikha S. Bagui, Dustin Mink, and Subhash C. Bagui. 2024. Node Classification of Network Threats Leveraging Graph-Based Characterizations Using Memgraph. Computers 13, 7 (2024), 171.

[8] Han Chen and Weiwei Deng. 2023. Interpretable patent recommendation with knowledge graph and deep learning. Scientific Reports 13, 1 (2023), 2586.

[9] Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. 2017. Metapath2vec: Scalable Representation Learning for Heterogeneous Networks. In KDD. 135–144.

[10] Matthias Fey and Jan E. Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. In ICLR (RLGM Workshop).

[11] Christoph Grimpe and Katrin Hussinger. 2013. Formal and Informal Knowledge and Technology Transfer from Academia to Industry: Complementarity Effects and Innovation Performance. Industry and Innovation 20, 8 (2013), 683–700.

[12] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In NIPS. 1024–1034.

[13] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous Graph Transformer. In WWW. 2704–2710.

[14] Mu-Hsuan Huang, Hsiao-Wen Yang, and Dar-Zen Chen. 2015. Industry-academia collaboration in fuel cells: A perspective from paper and patent analysis. Scientometrics 105 (2015), 1301–1318.

[15] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In ICLR.

[16] Guangtong Li L. Siddharth and Jianxi Luo. 2022. Enhancing patent retrieval using text and knowledge graph embeddings: a technical note. Journal of Engineering Design 33, 8-9 (2022), 670–683.

[17] Vincent Larivière, Benoit Macaluso, Philippe Mongeon, Kyle Siler, and Cassidy R Sugimoto. 2018. Vanishing industries and the rising monopoly of universities in published research. PLOS ONE 13, 8 (2018), e0202120.

[18] Yury A. Malkov and Dmitry A. Yashunin. 2020. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 42, 4 (2020), 824–836.

[19] Paolo Manghi, Alessia Bardi, Claudio Atzori, Miriam Baglioni, Natalia Manola, Jochen Schirrwagen, and Pedro Principe. 2019. The OpenAIRE Research Graph Data Model. https://zenodo.org/records/2643199

[20] Concepta McManus, Abilio Afonso Baeta Neves, and Alvaro Toubes Prata. 2021. Scientific publications from non-academic sectors and their impact. Scientometrics 126, 11 (2021), 8887–8911.

[21] Aristides Milios, Siva Reddy, and Dzmitry Bahdanau. 2023. In-Context Learning for Text Classification with Many Labels. In GenBench Workshop on (Benchmarking) Generalisation in NLP. 173–184.

[22] Natasha Noy, Yuqing Gao, Anshu Jain, Anant Narayanan, Alan Patterson, and Jamie Taylor. 2019. Industry-Scale Knowledge Graphs: Lessons and Challenges: Five Diverse Technology Companies Show How It's Done. Commun. ACM 62, 8 (2019), 36–43.

[23] Silvio Peroni and David Shotton. 2020. OpenCitations, An Infrastructure Organization for Open Scholarship. Quantitative Science Studies 1, 1 (2020), 428–444.

[24] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In EMNLP/IJCNLP. 3980–3990.

[25] Nils Reimers and Iryna Gurevych. 2020. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In EMNLP. 4512–4525.

[26] Angelo Salatino, Francesco Osborne, and Enrico Motta. 2020. Researchflow: Understanding the knowledge flow between academia and industry. In EKAW. 219–236.

[27] Serhad Sarica, Binyang Song, En Low, and Jianxi Luo. 2019. Engineering Knowledge Graph for Keyword Discovery in Patent Search. ICED (2019), 2249–2258.

[28] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. Collective classification in network data. AI Magazine 29, 3 (2008), 93–93.

[29] Jae-Wook Seol, Seok-Hyoung Lee, and Kwang-Young Kim. 2016. Author Disambiguation Using Co-Author Network and Supervised Learning Approach in Scholarly Data. International Journal of Software Engineering and Its Applications 10, 4 (2016), 73–82.

[30] L. Siddharth, Lucienne T. M. Blessing, Kristin L. Wood, and Jianxi Luo. 2021. Engineering Knowledge Graph From Patent Database. Journal of Computing and Information Science in Engineering 22, 2 (2021), 021008.

[31] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. 2015. An Overview of Microsoft Academic Service (MAS) and Applications. In WWW. 243–246.

[32] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. ArnetMiner: Extraction and Mining of Academic Social Networks. In KDD. 990–998.

[33] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. Journal of Machine Learning Research 9, 86 (2008), 2579–2605.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In NIPS. 5998–6008.

[35] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In ICLR.

[36] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous Graph Attention Network. In WWW. 2022–2032.

[37] Desheng Wu, Quanbin Wang, and David L. Olson. 2023. Industry classification based on supply chain network information using Graph Neural Networks. Applied Soft Computing 132, C (2023).

[38] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. 2022. Graph Neural Networks in Recommender Systems: A Survey. Comput. Surveys 55, 5 (2022), 97.

[39] Yan Xiao, Congdong Li, and Matthias Thürer. 2023. A Patent Recommendation Method based on KG Representation Learning. Engineering Applications of Artificial Intelligence 126, A (2023), 106722.

[40] Jian Xu, Sunkyu Kim, Min Song, Minbyul Jeong, Donghyeon Kim, Jaewoo Kang, Justin F Rousseau, Xin Li, Weijia Xu, Vetle I Torvik, Yi Bu, Chongyan Chen, Islam Akef Ebeid, Daifeng Li, and Ying Ding. 2020. Building a PubMed Knowledge Graph. Scientific Data 7, 1 (2020), 205.

[41] Hao Yan, Chaozhuo Li, Ruosong Long, Chao Yan, Jianan Zhao, Wenwen Zhuang, Jun Yin, Peiyan Zhang, Weihao Han, Hao Sun, Weiwei Deng, Qi Zhang, Lichao Sun, Xing Xie, and Senzhang Wang. 2023. A Comprehensive Study on Text-attributed Graphs: Benchmarking and Rethinking. In NeurIPS. 17238–17264.

[42] Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit Singh, Guangzhong Sun, and Xing Xie. 2021. GraphFormers: GNN-nested transformers for representation learning on textual graph. In NeurIPS. 28798–28810.

[43] Fanjin Zhang, Xiao Liu, Jie Tang, Yuxiao Dong, Peiran Yao, Jie Zhang, Xiaotao Gu, Yan Wang, Bin Shao, Rui Li, and Kuansan Wang. 2019. OAG: Toward Linking Large-Scale Heterogeneous Entity Graphs. In KDD. 2585–2595.

[44] Jianan Zhao, Meng Qu, Chaozhuo Li, Hao Yan, Qian Liu, Rui Li, Xing Xie, and Jian Tang. 2023. Learning on Large-scale Text-attributed Graphs via Variational Inference. In ICLR.

[45] Cheng Zheng, Bo Zong, Wei Cheng, Dongjin Song, Jingchao Ni, Wenchao Yu, Haifeng Chen, and Wei Wang. 2020. Node Classification in Temporal Graphs Through Stochastic Sparsification and Temporal Structural Convolution. In ECML/PKDD. 330–346.

[46] Jason Zhu, Yanling Cui, Yuming Liu, Hao Sun, Xue Li, Markus Pelger, Tianqi Yang, Liangjie Zhang, Ruofei Zhang, and Huasha Zhao. 2021. TextGNN: Improving Text Encoder via Graph Neural Network in Sponsored Search. In WWW. 2848–2857.

[47] Haoyu Zuo, Yuan Yin, and Peter Childs. 2022. Patent-KG: Patent Knowledge Graph Extraction for Engineering Design. Proceedings of the Design Society 2 (2022), 821–830.