University of Southampton

Faculty of Engineering and Physical Sciences

Electronics and Computer Science

Cross-domain research paper recommender system using Graph neural networks

Ji Hwan Kim

15th Nov 2023

Supervisor: Dr Bahar Rastegari

Second Examiner: Prof Mahesan Niranjan

A dissertation submitted in partial fulfilment of the degree

of MSc Data Science

Abstract

In recent years, as the complexity of the problems we face increases, researchers are increasingly collaborating to tackle them. The challenges we face today often require knowledge and expertise from multiple fields to be effectively addressed. As a result, research collaborative networks are growing in importance, facilitating interdisciplinary research. These networks allow researchers from different disciplines to share ideas, resources, and expertise, leading to new insights and discoveries. Thus, this study will perform research on cross-domain research paper recommender systems, particularly using graph neural networks and utilising the data from ArnetMiner. In spite of the initial problem of having no ground truth, to mitigate this issue contrastive learning has been used. However, this research finished only comparing the trained result to the original one and pointing out how they were influenced and how they can be improved. Thus, this research becomes a cornerstone, encouraging other researchers to keep improving this approach.

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1.Background

Research has been conducted along with human society. Ever since the society was civilised, research has been one of the significant factors which made the society developed. As many problems were solved via research, the significance of research has grown. However, in recent years, as the complexity of the problems we face increases, researchers are increasingly collaborating to tackle them. The challenges we face today often require knowledge and expertise from multiple fields to be effectively addressed. As a result, research collaborative networks are growing in importance, facilitating interdisciplinary research. These networks allow researchers from different disciplines to share ideas, resources, and expertise, leading to new insights and discoveries.

Research from other disciplines is sometimes significantly valuable for solving challenging problems. However, it is not common for researchers to actively read or follow the one from the different fields. As a result, they are frequently stuck and waste lots of time to come up with new solutions, even though a similar problem they are struggling with was already solved in other areas. This occasional problem shows the importance and needs of collaborating with researchers from different fields.

Moreover, in many cases, research from other domains sometimes gives researchers a creative insight they have never thought of, which is also called “cross-domain knowledge transfer”. In fact, the most common example of cross-domain transfer learning is imitating the phenomena from nature. Many neural network algorithms were examples; for instance,

the initial thought of a perceptron used in deep learning is from a neuron of the brain. In addition, the neocognitron, the concept utilised for convolutional neural networks, was also motivated by human visual cortices [1].

In terms of information transfer or knowledge transfer, this leads to interdisciplinary research, the collaboration of researchers from different fields. Some researchers precisely treat the research collaboration in two ways: interdisciplinary and transdisciplinary research. However, as the significance is in that researchers are becoming more collaborative with each other, this study will neither discuss the difference between the two nor accurately differentiate them. One of the goals to achieve by conducting interdisciplinary research is to solve a complex problem, which is particularly not easy to solve within the discipline.

According to Van Noorden [2], as a number of researchers reckon the importance and effectiveness of interdisciplinary research, the number of these research collaborations has increased in recent decades. Nonetheless, the infrastructure for interdisciplinary research has still not yet fully functioned. As an example, for ordinary research, every researcher uses search platforms such as Google Scholar. This type of service even recommends research papers in which users are likely to be interested. However, there is no research recommendation tool for interdisciplinary research specific, and research on this topic has not been conducted.

Thus, this study will perform research on cross-domain research paper recommender systems, particularly using graph neural networks. The system is expected to suggest the most relevant research from other domains. In this research, the data from ArnetMiner will be used to construct the citation network where the nodes represent research papers and edges represent

citations. Also, detailed information on the title, keywords, and abstract of each research paper will be added. Then, this study will compare the similarity between two different research and update the weights of the edges with this knowledge using the graph neural networks.

1.2. Objectives and motivation

Research paper recommender systems have become more pivotal in the academic landscape [3]. These systems largely function within a specific discipline, recommending research papers for researchers to utilise to explore the former studies. However, as the complexity of the world issues escalates, interdisciplinary approaches drawing knowledge across various fields is in need. In spite of this, the existing research paper recommender systems have not yet fully equipped or supported the capability of recommending papers across different research domains.

Moreover, even representing the problem is sometimes challenging due to its complexity. The best data structure to represent the information, which includes the relationship between entities, is a graph. Therefore, this research actively utilised the graph data structure to represent this citation network. Also, although many ways have been proposed and used to build recommender systems, the method of neural networks, which is the state-of-the-art technique in most fields, was chosen. In particular, as the dataset this research uses is in a graph format, graph neural networks were used. In fact, graph neural networks particularly show their power when training the graph-structured data.

This system will be significantly helpful for researchers to do preliminary research, such as finding literature on related works, because they are usually not familiar with finding useful

research papers. Furthermore, the system can be used for finding prospective collaborating researchers. Scholars have been looking for co-researchers via word-of-mouth, other fellow researchers' recommendations or by reading other research papers. These conventional ways are, in fact, substantially subjective, but this research is expected to give more objective and reliable results. Also, researchers can utilise the system to get some fresh insights when they are stuck or struggling with finding a proper solution.

Thus, this study aims to fill these research gaps by proposing a novel system for cross-domain paper recommendation.

The objectives of this study are like below:

- To collect and preprocess research data of multiple domains.
- To create a citation network graph that captures the similarity between research in different domains.
- To select and implement a suitable GNNs architecture that can effectively learn embeddings from the citation network.
- To explore the combination of natural language processing techniques and GNNs for extracting relevant features from research items and improving the quality of cross-domain recommendations.
- To develop a recommender system that uses the learned embeddings to suggest prospective research from other disciplines.

By achieving these objectives, this project aims to contribute to the interdisciplinary research field, enabling researchers to discover and apply useful research from other domains in their work.

1.3. Research questions

- How to generate the graph (citation network)?
- How to define the distance function (similarity function)?
- How to evaluate the result?

1.4. Outline of this work

The ultimate goal of this project is to promote interdisciplinary research and encourage creative solutions to problems. This report is structured as follows. In Chapter 1, the background, objectives and motivation, and research questions of this research are introduced. Former literature regarding the topic of "cross-domain research paper recommendation" will be critically analysed in the following chapter. Moreover, in Chapter 3, all data as well as methodologies used in this research will be introduced and explained how they were used. Then the chapter 4 shows the result of this research with explanations. Chapter 5 will discuss this research. The report will be concluded in the last chapter, followed by the bibliography and appendix.

2. Literature reviews

2.1. Interdisciplinary research

Traditionally, research has usually been conducted within a single discipline because this way can help scholars dedicate and study deeper. However, as the problem researchers argue is getting complex, another approach is required. Unlike traditional research, interdisciplinary research draws on knowledge and methods from multiple disciplines and helps scholars understand more complex topics better [4]. Therefore, the importance of collaborating with researchers in different domains has been increasing [5].

To be one of the effective ways of being admitted by fellow experts, publishing research work has dominated academia [6]. As the number of publications in the research field increases, scholars are hardly possible to follow every novel research instantaneously, and it is more evident if the research is from other disciplines [7, 8]. Therefore, it is challenging to understand the most advanced methods or concepts on the specific topic from the perspective of other domains, although there are increasing demands on collaborating with experts studying another subject. This sometimes leads to misunderstandings or misinterpretations of the context [9].

The most fundamental way to understand other scholars in academia is reading each other's literature, but it is always challenging to read the research papers with which they are not familiar [10]. Identifying the significant research is also not an easy task as they are usually in different vocabulary [11]. In addition, researchers can learn academic knowledge from conferences or seminars as well. However, similar to reading the research papers, it is not simple to distinguish whether the research will be useful. Therefore, it is required to have

more objective methods of research recommendation when conducting interdisciplinary research.

In order to find collaborators, Clarivate [12], one of the leading companies in providing solutions and services for research and innovation, suggests answering the following questions: which authors are writing the most papers on the specific topic, who are the authors of hot and highly cited papers on the specific topic, which authors have cited you, and who are the authors writing in the specific field that have the most overall citation impact. This approach has been widely used by researchers to find their co-workers. However, as this approach is subjective, the answers to those questions have been different by researchers. This, in addition, limits the finding to the specific domain. Also, Mansilla [11] pointed out that most researchers build networks within their own discipline. As a result, finding collaborators from other fields is time-consuming as well as requires additional effort.

Collaborating with researchers from another domain can also give some new insights that they have never thought of. Pohl [13] insists that the future of knowledge will be in interdisciplinary or transdisciplinary research. From his own experience of participating in the transdisciplinary research, although he was frustrated with inefficient discussion, he gained many new insights on the particular topic. Also, Bruun, et al. [14] evaluate the reason of scholars selecting interdisciplinary research is that this method can produce new and broad learnings on a specific topic.

2.2.Citation networks

Reading fellow researchers' works plays a significant role in understanding the field and following the trends of research simultaneously [15]. Citation networks are essential to represent the relationship between research papers to show the brief landscape of the research field. The concept of citation networks was first introduced by Garfield [16] in his study of citation analysis. He also proposed the idea of the science citation index (SCI), which has been used as the norm to evaluate the reputation or influence of journals. In addition, Greenberg [17] claims that the network is a robust structure of social communication and also a well-balanced research method. Therefore, a number of studies started to focus on its analysis.

As analysing citation networks has been one of the active research topics, this analysis is utilised in many areas, such as to measure the impact of one discipline on society or the influence of the research [18]. Although it is the author's own decision which research to cite, in the end, if the number of authors is meaningfully considerable, the citation count tells the impact of the research [19]. Additionally, in terms of metrics, an H-index [20] is commonly used to analyse the citation networks quantitatively. Also, the Eigenfactor score [21], widely used to show the journal impact, is another crucial measure. These metrics have been actively utilised from tools and platforms such as Google Scholar or Web of Science.

2.3.Recommender system

In modern society, it is not deniable that people become more reliant on recommender systems. Many popular platforms such as Netflix, YouTube and Spotify, for instance, highly utilise the system, and people use them without much attention. Thanks to its growing use,

the system has evolved dramatically. In particular, the methodology the engineers and researchers used to develop the system has been utilised, from neighbourhood methods such as content-based filtering and collaborative filtering [22, 23] to representation-based learning methods [24].

The content-based filtering algorithm is one of the neighbourhood methods for recommender systems. As this algorithm is based on a description of items and a profile of users' interests, it recommends items similar to those a given user has liked in the past [25]. This feature is one of the benefits of using this approach; however, at the same time, it is also a drawback in that only similar items can be recommended. Another advantage of this algorithm is that it can react to new items which have never been introduced to the model [26]. This aspect seems simple, but the potential of it plays its role in preventing cold start problems, which is typical for many recommender systems.

The collaborative filtering approach is another neighbourhood method used in recommender systems, which became more widely used after it proved the quality of the result via the Netflix prize competition [27]. The concept of this algorithm is simple. It fundamentally uses the data from other users who have similar preference and suggest items based on the analysed data. There are two main approaches to this method: one is user-based, and the other is item-based collaborative filtering [28, 29]. The user-based collaborative filtering method, first introduced by Resnick, et al. [30] in 1994, focuses on the similarity of preferences between users. The algorithm suggests what other users chose before since the item would be likely to be selected by the active user based on their preference affinity. Therefore, a more accurate personalised recommendation becomes available, and the method fosters serendipity in recommendations [23].

On the other hand, the item-based algorithm, which generally yields better results compared to the user-based approach, leverages the associations between items. These associations can be systematically created through the use of a user-item interaction matrix. It recommends the paired items once the users choose one another [31]. In addition, the item-based filtering typically shows more stable results than the user-based one because item-item relationships are observed to be more reliable than user-user relationships [32].

Despite its excellence in predicting users' interests, the data, specifically a user-item matrix, this approach uses is highly sparse in general, which leads to a known problem called the *cold start* [33]. Although many techniques have been introduced in order to resolve the cold start problem, it is challenging to overcome the fundamental problem this approach has. In addition, this approach can cause the scalability issue as the size of the data increases [34].

According to Beel, et al. [3], the majority of the recommendation algorithm used was content-based filtering, and it is followed by collaborative filtering and a graph-based recommender system. Although much research has been done using the content-based filtering method, its performance has not always been better than the others. Also, Bai, et al. [35] pointed out some known issues of the research paper recommender system. The problems are actually similar to the ones ordinary recommender systems have. One specific issue with this particular system is the standards of unified scholarly data. Due to the absence of standards in terms of scholarly data, platforms providing this research information, such as Google Scholar or Web of Science, use all different forms of data. It sometimes matters when collecting the data from them.

Another approach commonly used for recommender systems is representation-based learning. To be more specific, matrix factorisation or deep learning models are well-known examples of the representation-based model [24]. The method utilising the matrix factorisation tries to combine the data of users and items together and embeds them into continuous vectors [36]. In addition, the research area on recommender systems based on deep learning has been burgeoning lately [37]. A number of studies using neural networks show outstanding output compared to the traditional approach. Unlike matrix factorisation, which is limited to working linearly, this approach can perform nonlinearly [38]. It also shows the improved result on such non-linear and non-trivial user-item relationships.

2.4.Graph Neural Networks

As the complexity of the data increases in the real world, it becomes challenging to represent the data in an applicable format. One of the effective ways to represent data is using a graph data structure. The graph data structure consists of a set of nodes and edges. The nodes represent the entity of the dataset, and the edges represent the relationship between them. Thanks to its versatility, any type of information, even from the molecular structure to the social network, can be stored as a graph data structure.

There are several types of graphs representing the network of research papers. Some researchers include papers, authors, venues, the paper published year and titles in the network, and therefore, the connections can be citations, authorships or the relation of published depending on its nodes [3, 39, 40]. On the other hand, Huang, et al. [41] use the entity's attribute information and calculate the similarity between the entities. However, although those studies used a graph as the data structure representing the information, they

are still based on conventional recommender system algorithms such as content-based or collaborative filtering.

As neural networks are widely used in many places, a number of studies are actively ongoing about graph neural networks, which exceptionally focus on the data formed in graphs. There are many types of algorithms have been proposed which are similar to ordinary neural networks. Graph convolutional networks, which were introduced by Kipf and Welling [42], share the idea of utilising convolution with convolutional neural networks (CNNs) (2-1). This method is one of the frequently used graph neural network architectures and shows reasonable results to researchers. Similar to CNNs, the GCNs have also been largely utilised for classification tasks, but these neural networks can be used for the task of node regression, which works well with tabular data [43].

$$h_i = \frac{1}{\sqrt{\deg(i)}\sqrt{\deg(j)}} \sum_{j \in N_i} x_j W^T \tag{2-1}$$

Graph attention networks, which utilise the weight factors using *self-attention*[1], were proposed by Veličković, et al. [44]. The fundamental concept of this model is that some neighbour nodes are more influential than others. The advanced factor this method has compared to the graph convolutional networks is that it utilises attention scores based on self-attention rather than degrees of nodes [43]. However, Brody, Alon and Yahav [45] claim that the original GATs have the fundamental limitation of only using a static attention mechanism. They, therefore, proposed the GATv2, which outperforms the previous networks. In addition, in terms of classification tasks, Labonne [43] claims that the insufficiently connected nodes may influence to the model performance adversely (2-2). The author

[1] The core method used for the transformer.

showed that the more the number of degrees increases, the higher the accuracy score the model achieves.

$$h_i = \sum_{j \in N_i} \alpha_{ij} \mathbf{W} x_i$$

$$\alpha_{ij} = \frac{\exp\left(W_{att}^T LeakyReLU\left(\mathbf{W}[x_i || x_j]\right)\right)}{\sum_{k \in N_i} \exp\left(W_{att}^T LeakyReLU(\mathbf{W}[x_i || x_k])\right)} \quad (2\text{-}2)$$

2.5. Contrastive learning

A loss function is one of the key concepts used in the field of artificial intelligence to improve the model's performance. There are a few popularly utilised functions, such as a mean squared error (MSE), a root mean squared error (RMSE), a cross-entropy loss (CE) or a binary cross-entropy loss (BCEL). In addition, each function is usually used for some specific tasks. MSE, for instance, predominantly works with regression tasks, and BECL, on the other hand, is used for classification tasks most [46].

In supervised learning, the loss function usually compares the result of the model (prediction) to the true labelled data (answer) and computes the loss during training [47]. On the other hand, in unsupervised learning, it is not possible to compare the predicted result to the true labels using the loss function, as the true data does not exist in the training phase [48]. The loss function, instead, encourages the model to learn the useful data that researchers set. Therefore, in the task of unsupervised learning, the model tends to focus on finding the task's own patterns by minimising the loss. Supervised learning has successfully solved tremendous problems thanks to its simple and straightforward concept; however, as it heavily depends on generating the labelled dataset, researchers were looking for alternatives [47, 49].

As a result, self-supervised learning, as one of the alternative machine learning paradigms, has recently been gaining popularity, where it is distinguishable from supervised learning with regard to the labelling of the dataset [50]. The most significant benefit of using this approach is that it does not require the labelled data. The objective of the self-supervised learning approach is to generate the representations using pretext tasks which do not require to have labelled data [51]. Contrastive learning is one of the successful self-supervised learning techniques introduced by Hadsell, Chopra and LeCun [52], widely used in recent research fields. This approach requires two pairs, positive and negative pairs, and is fundamentally trained based on minimising the contrastive loss, which makes the similar pairs closer. However, dissimilar pairs move apart in their representation space (2-3).

$$\begin{aligned}\mathcal{L}_{count}(x_i, x_j, \theta) \\ &= 1[y_i = y_j]\|f_\theta(x_i) - f_\theta(x_j)\|_2^2 \\ &+ 1[y_i \neq y_j]\max(0, \epsilon - \|f_\theta(x_i) - f_\theta(x_j)\|_2^2\end{aligned} \tag{2-3}$$

However, as contrastive learning is sensitive to hyperparameters, its performance is highly dependent on the hyperparameter tuning [53]. Similarly, selecting the negative samples also vitally affects its result, and as the quality of the data representation relies on its initial form, valid pre-processing is crucial [53, 54]. Despite these known drawbacks, its reputation is improving, and it is dominating the research field.

3. **Methodology**

3.1.Environment

This research was largely conducted on the Linux system of Iridis5, which is the high performance computing cluster the university provided. Some preliminary research and light tasks have been done on MacOS, which is my personal computer. Also, across the whole process, the anaconda virtual environment, whose version is 23.7.3, was mainly utilised, and every script was written in Python version 3.10.12. In addition to this, a number of external Python libraries, such as PyTorch[2], DGL[3], Pandas[4], NumPy[5], Scikit-learn[6], and PyMongo[7], were used. Furthermore, in particular, for the dataset management, the MongoDB system was used as the raw data collected was in MongoDB Extended JSON format.

3.2.Dataset

The dataset used in this project is originally downloaded from ArnetMiner database system, which stores various data [55]. According to its website[8], the database has been used by over 68 million researchers for over 295 million publications. Furthermore, this database has been cited over 1.2 billion times by fellow researchers for various reasons. This citation network dataset comprises information from 5,259,858 papers and 36,630,661 citation relationships**.** **Table 3-1** shows the field of the dataset. Among these attributes, id, title, keywords,

[2] PyTorch is an optimised tensor library for deep learning using GPUs and CPUs.
[3] Deep Graph Library (DGL) is the Python library mainly for deep learning on graph structured data.
[4] Pandas is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool.
[5] The fundamental package for scientific computing with Python.
[6] Scikit-learn is simple and efficient tools for predictive data analysis.
[7] PyMongo is a Python distribution containing tools for working with MongoDB.
[8] https://www.aminer.org/

fos.name, fos.w, references, n_citation and abstract have only been selected for this study (**Table 3-2**; **Table 3-3**)

Field Name	Field Type	Description
id	string	paper ID
title	string	paper title
authors.name	string	author name
author.org	string	author affiliation
author.id	string	author ID
venue.raw	string	paper venue name
year	int	published year
keywords	list of strings	keywords
fos.name	string	paper fields of study
fos.w	float	fields of study weight
references	list of strings	paper references
n_citation	int	citation number
page_start	string	page start
page_end	string	page end
doc_type	string	paper type:journal,conference
lang	string	detected language
volume	string	volume
issue	string	issue
issn	string	issn
isbn	string	isbn
doi	string	doi
url	list	external links
abstract	string	abstract
indexed_abstract	dict	indexed abstract
v12_id	int	v12 paper id
v12_authors.name	string	v12 author name
v12_authors.org	string	v12 author affiliation
v12_authors.id	int	v12 author ID

Table 3-1. Data schema

3.3. Pre-processing

Pre-processing is one of the steps that must be done in order to train the model effectively. In general, regardless of how data is collected, the raw data should be cleaned in accordance

with the research plan. Likewise, although the data has been collected from ArnetMiner database, which is mentioned above, it was filtered and converted into a different format accordingly.

Field Name	Field Type	Description
id	string	paper ID
title	string	paper title
keywords	list of strings	keywords
fos.name	string	paper fields of study
fos.w	float	fields of study weight
references	list of strings	paper references
n_citation	int	citation number
abstract	string	abstract

Table 3-2. Used data schema

id	title	keywords	abstract	field of study
53e997e9b7602d9701fe682c	Animating soft objects.	['computer animation']	Soft objects change shape as they move. Specif...	[{'name': 'Computer graphics (images)', 'w': 0...
53e997f1b7602d9701fef278	Attribute coupled grammars	['composition operator', 'lisp', 'attribute gr...	In this paper, attribute grammars are viewed a...	[{'name': 'Tree-adjoining grammar', 'w': 0.640...
53e997ecb7602d9701fe8352	Avoiding object misconceptions	['programming language', 'distance education']	This paper identifies and describes a number o...	[{'name': 'Computer science', 'w': 0.43557}, {...
...	...	...	...	...
5e6b5bd293d709897c9ba7a6	Machine Learning Models for Secure Data Analyt...	['Big data', 'Secure Data Analytics', 'Data re...	In recent years, rapid technological advanceme...	[{'name': 'Architecture', 'w': 0.40234}, {'nam...
5e72105293d709897cf03789	Deep Reinforcement Learning for 5G Networks: J...	['Reinforcement learning (RL)', 'deep learning...	The fifth generation of wireless communication...	[{'name': 'Link adaptation', 'w': 0.54007}, {'...
5a260c0c17c44a4ba8a1e1b0	Nonlinear random matrix theory for deep learning	['machine learning']	Neural network configurations with random weig...	[{'name': 'Nonlinear system', 'w': 0.51368}, {...

Table 3-3. Actual dataset used

3.3.1. Filtering the data

As the size of this dataset is tremendously huge, accounting for about 15GB, particularly for the task of keyword extraction from paper abstracts, it was not feasible to conduct the research using all data; therefore, filtering the data was necessary. In order to filter the data meaningfully, a threshold of the number of citations was decided to be used to identify the significant data for this research. Furthermore, the data, which consists of null and meaningless values, was filtered out as well. For example, some abstracts were filled with special characters, and some data did not have titles at all. Additionally, the link information did not explicitly exist as separate files, but it was included in the dataset as a name of reference. Therefore, another file of the relationship was created for easy use of the link information.

3.3.2. Keyword extraction and word embedding

After filtering out some impractical data, all chosen fields of data: title, keywords, abstract and fields of study, have to be converted into embedding format so as to be used for model training. For this task, *fasttext*, backed up by the Facebook Open Source and one of the state-of-the-art Python libraries in the natural language processing (NLP) field [56], was utilised. The advantage of using this library is, as it is recognisable from its name, the speed of its process. Although there is some trade-off for the accuracy, it is considerably faster than other NPL libraries, keeping its accuracy with a large corpus for word representation. Since the focus of this research is not on this NLP task, the most advanced library was chosen to be used. Therefore, all data was transformed into a numerical format using this library.

However, as the abstract, in particular, is usually written within around 200 words, in order to embed the information effectively, a keyword extraction task has been performed prior to the

embedding task. This task selected ten meaningful keywords, and only the ones whose weight is greater than the average weight, practically calculated by 0.3 in this research, were picked between them. For this specific task, the Python library called *keyBERT*, also one of the leading NLP libraries and particularly tuned for keyword extraction tasks [57], was utilised. While other data such as titles, keywords and domains are straightforward, abstract is the only data which includes much information in it, which could be translated in many ways. Therefore, as the method is based on the *BERT*[9], which is the most famous NLP model using a *transformer*, and therefore the quality of the result could be assured, this library was chosen, although this library takes time to get the result.

Moreover, in order to get the total embedding value of a list of words, the way of computing the average has been generally performed. This method was applied to calculate the embedding of keywords as well. Also, the data representing the domain of research, in particular, contains not only the name of it but also the weight of each as well. Therefore, for more precise embedding, weighted arithmetic mean, Equation (3-1), was used.

$$total\ embedding = \frac{\sum w_i e_i}{\sum w_i} \tag{3-1}$$

3.4.Main Task

3.4.1. Build a graph

With the pre-processed data, the next step was to build a citation network graph. The graph consists of nodes, which are research papers, with four-node features: the word embedding of title, abstract, keywords, and domains. Each embedding had 300 dimensions, and therefore,

[9] Bidirectional Encoder Representations from Transformers

they were in (162207, 300) forms. In addition, for computational efficiency, they were used in a combined form, whose shape was in (162207, 1200). Also, the pre-processed edge data, shaped of (1273175, 2), was used to construct the graph. As a result, in spite of filtering out the data, the graph is still in a vast form.

3.4.2. Train the graph using graph attention networks (GAT)

In this research, the graph attention networks have been used in order to train the citation network graph. For this network, two graph attention layers were utilised along with the exponential linear unit (ELU) as an activation function to avoid simplifying the structure [58]. ELU is, in fact, one of the commonly used activation functions for the task of graph neural networks, thanks to its excellence in fixing the dying ReLU problem [59]. This architecture consisted of 1200 dimensions of the input layer, 300 dimensions of the hidden layer with four attention heads, and 50 dimensions of the output layer (**Figure 3-1**).

However, this research had a limitation of the absence of the ground truth for the validation in the phase of training, which usually plays a significant role in supervised learning tasks. Therefore, in this case, contrastive learning is used to mitigate this drawback of the absence. Also, among several options of loss that the contrastive learning can use, *the contrastive loss* is chosen. This type of self-supervised learning with contrastive loss requires defining the positive and negative pairs to calculate the loss. In order to generate them, randomly selected pairs were used with customised similarity metric, which used the cosine similarity in basic to calculate the distance between each feature representation; if the distance is greater than the higher threshold, it will be categorised as the positive pair, and if the distance is less than the lower threshold, the pair will go to the negative pair. As the higher threshold and lower threshold are the hyperparameters, they are set to 0.5 and 0.35, respectively. In the

process of computing the total similarity, each feature distance is weighted differently[10], and if the pair are linked, they gained more value by 0.05.

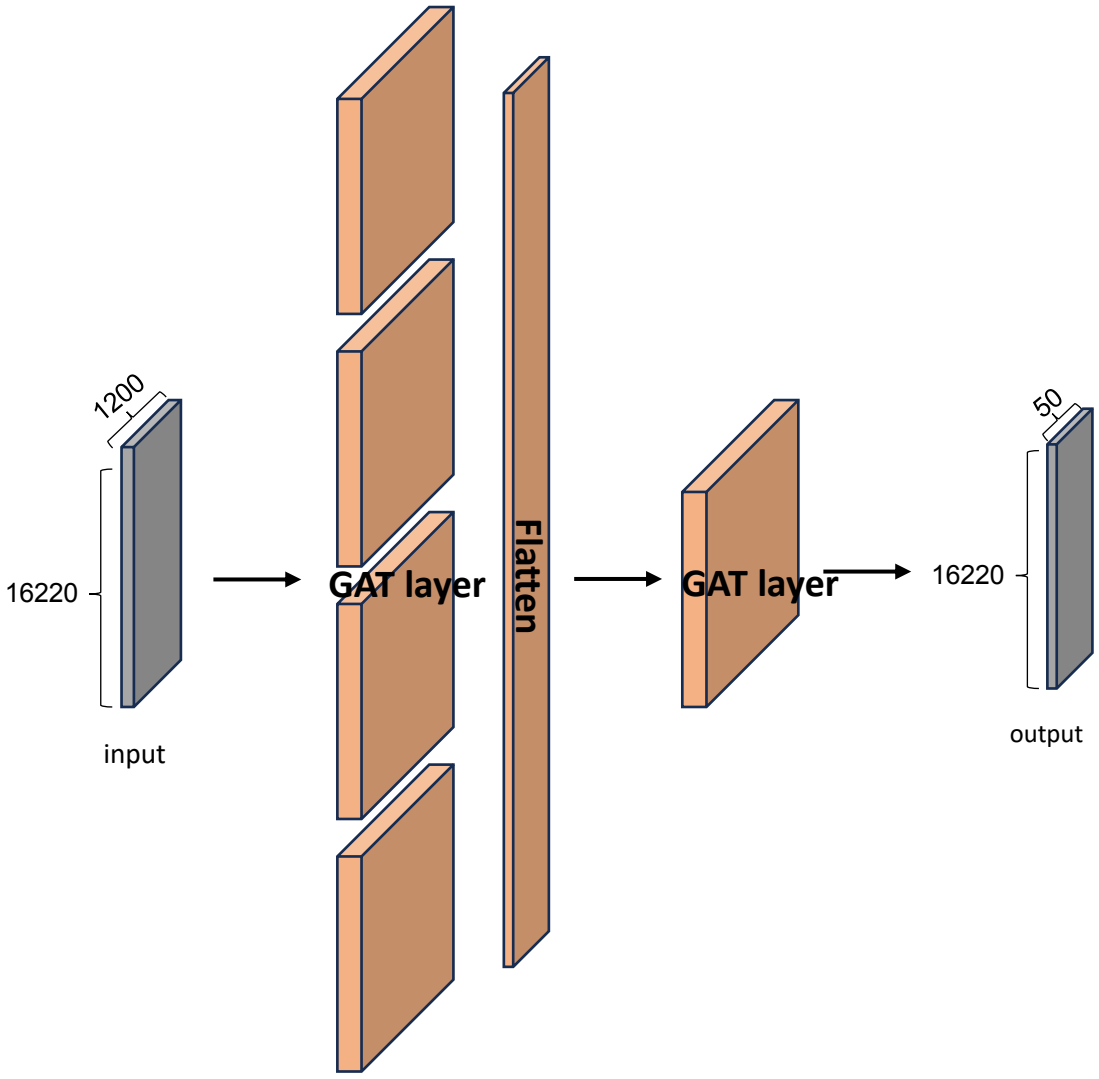

Figure 3-1. Neural network architecture

Next, using the result of the trained graph attention networks, which are the new node embeddings, the model will predict the links. The process of predicting the links will utilise the distance function, which was defined earlier for the contrastive learning task. With the trained node embedding and the distance function, the system will calculate the similarity and then choose a certain number of potential links with high similarity scores. The updated

[10] The weights are set to 0.2 for title, abstract and keywords, and 0.4 for domains.

relationship will be applied to the original graph. In addition, after repeating the training and predicting tasks, once the user inputs a certain name of the research paper, this system will recommend the five most relevant research papers.

4. Results

This chapter will discuss the findings and the results of the research.

In the phase of preprocessing, the first step performed was filtering out and cleaning the data. The data was initially filtered out if it had been cited more than 100 times. The number of original data was 5,259,858 nodes and 36,630,661 links, and it was decreased by 162,227 entities and 1,384,372 edges. However, there were some invalid data in the domain, so after cleaning those as well, the number of data used was decided to be 162,207 and 1,273,175 (**Table 4-1**).

data	count
original_node	5,259,858
filtered_node	162,207
original_edge	36,630,661
filtered_edge	1,273,175

Table 4-1. The number of nodes

Then, the filtered data was transformed into the representation data form with 300 dimensions. As mentioned in the previous chapter, the Python library *fasttext* was mainly used for the embedding, and *keyBERT* was applied to extract the keywords from the text. Also, from the raw dataset, the attributes used in this research were the title, abstract, keywords and the fields of study. In addition, with this transformed data the similarity scores have been computed, which were based on the cosine similarity and weighted sum, in order to analyse the data's characteristics and trends (4-1).

$$total\ similarity = \frac{\sum w_i\ \text{similarity}_i}{\sum w_i} \tag{4-1}$$

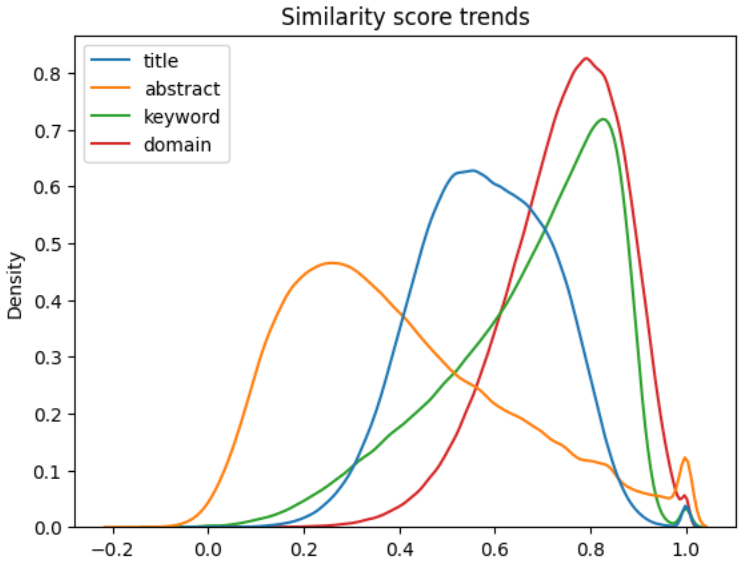

Figure 4-1. Trends of similarity scores

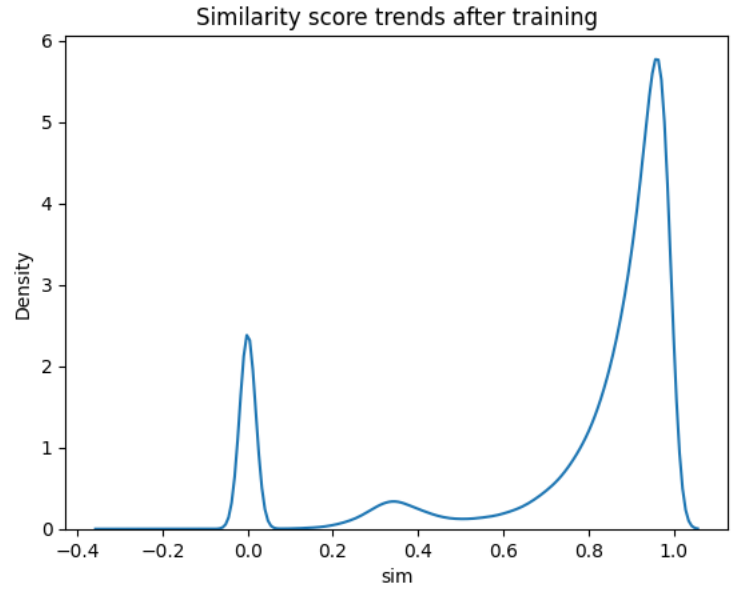

Figure 4-2. Trends of similarity scores after training

Furthermore, after the model training phase, the model returns the trained node embeddings with 50 dimensions. In order to see the trends of this data, similarity scores were calculated. Additionally, they were compared to the one of the original data. During the computation, while most results were shown to have specific values nevertheless, what they mean, 20,777 results, which account for 12.8%, were filled with all zeros.

5. Discussion

This research has not honestly gone as if it was planned in terms of implementation, and therefore, the result was a bit limited. This chapter will discuss the reflection on this research and the possible future plans to improve this study.

The result shows the implication of training under the criteria provided, where the model gains values if the title, abstract and keywords of nodes are similar, and it loses points if the domains of two nodes are closely related. It is because the main research theme is to find research papers which are written about similar topics or are similarly studied in terms of methods but are, more importantly, from different fields of study. As a result of the model training, 79.85% of links lost their distance; on the other hand, 20.12% of initial edges was departed away (Figure **4-1**; Figure **4-2**). This means, in other words, nearly 80% of relationships among the existing links became closer after the training. One of the examples which shows the significant difference is the link of '*CONTRAfold: RNA secondary structure prediction without physics-based models'* and '*Rfam: An Rna Family Database*'. They seem quite related, but from a similar domain as well. The possible reason could be that the data of fields of study is too specific, and therefore, it could be better if the domain data were a bit higher level.

Interestingly, in spite of many times of endeavour, the trained mode returns significant numbers of embeddings with the result of zeros. This turns out that those nodes have no in-degrees while have several out-degrees, which means that they can affect other neighbour nodes, but they cannot be influenced by their neighbours. However, this issue could affect the whole graph in the end; therefore, the nodes without in-degrees are better filtered out for future research.

In the phase of preprocessing, all node feature data was supposed to be converted into the embedding format to be used for the model training. However, converting the format is quite a time-consuming and computationally expensive task. In addition, if the length of the text is too long, the embedding could include some noise in it, which is unnecessary. Under these known problems, in particular, transforming the whole abstract was inefficient as well as ineffective. Therefore, a keyword extraction task was decided to be added prior to the embedding step. This led to achieving more precise and robust embedding data. For this specific task of conversion, *fasttext* and *keyBERT* were utilised, as mentioned in Chapter 4. One of the primary issues faced while conducting this research was the size of the graph. As its number of nodes and edges is too huge, several tasks, such as calculating the similarity between all nodes for the link prediction, were not feasible. Unfortunately, it was hardly possible to proceed further with the research due to the scalability issue, as this study was designed to use link prediction to build the recommender system. However, there will be a particular solution to handle this type of scalability issue as the data will become more complex and scalable.

Subsequently, another challenging point encountered was the fact that the ground truth did not exist. It was even, at first, not easy to train the model without the ground truth as the supervised learning tasks have been dominating this field; therefore, validating the predicted result was common. However, there were several ways to solve this issue, and among them, this research used the self-supervised learning technique, as mentioned in Chapter 4. In this particular study, contrastive learning has been utilised with contrastive loss. In spite of this alternative method, for the model evaluation, any form of ground truth was required. The first idea was to do a survey or interview experts to ask about the quality of the results. This

approach, however, was not practical and less scientifically objective but more qualitative. As a result, this is also one of the limitations of this research.

On the other hand, there were some new ideas to consider for future research. The first one was the use of authors' information. This study used a homogeneous graph; however, in the field of graph neural networks, researchers often use a bipartite graph to represent more information in one data structure so that they can analyse more complex relationships. Therefore, the information of the authors could be dealt with.

In addition, it could be more beneficial if the model can manage the multiple features separately. For this work, the combined node feature was utilised in the training session. Therefore, it used the same weight while training the model. However, if the node features were handled respectively, the model could use separate weights for each node feature: title, abstract, keywords, and domains. This could lead to more accurate training results; however, due to the limited time, this idea was only considered but not applied to the actual model (5-1).

$$h_i = \sum_{j \in N_i} \alpha_{ij} \mathbf{W} x_j$$
$$h_i = \sum_{j \in N_i} \alpha_{ij} W_{title} x_j + \beta_{ij} W_{abstract} x_j + \gamma_{ij} W_{keywords} x_j + \delta_{ij} W_{domain} x_j \quad (5\text{-}1)$$

Furthermore, the direction of edges is also one aspect to discuss. The citation networks are fundamentally to be directed graphs as they consist of a number of research papers which are referencing other papers, the cited papers. Therefore, the direction from the source to the destination exists. While considering the method of link prediction, the direction became an obstacle because the direction was not considered to compute the similarity score between

two nodes. However, for the usual link prediction task, the direction of the edge is not the focal point, but the connection is. Therefore, although it is meaningful for the network to be in a directed form, in terms of analysing the graph, it is less crucial as the nodes cannot cite each other. This means that the graph can be considered an undirected graph since one should be published earlier than the other; therefore, they cannot mutually reference each other. Nonetheless, this assumption seems reasonable, considering directed to undirected showed different results. **Figure 5-1** illustrated that two graphs showed a bit different result in terms of the difference between similarity using trained result and original similarity particularly. Thus, considering the directed graph to undirected for the model training should be reconsidered.

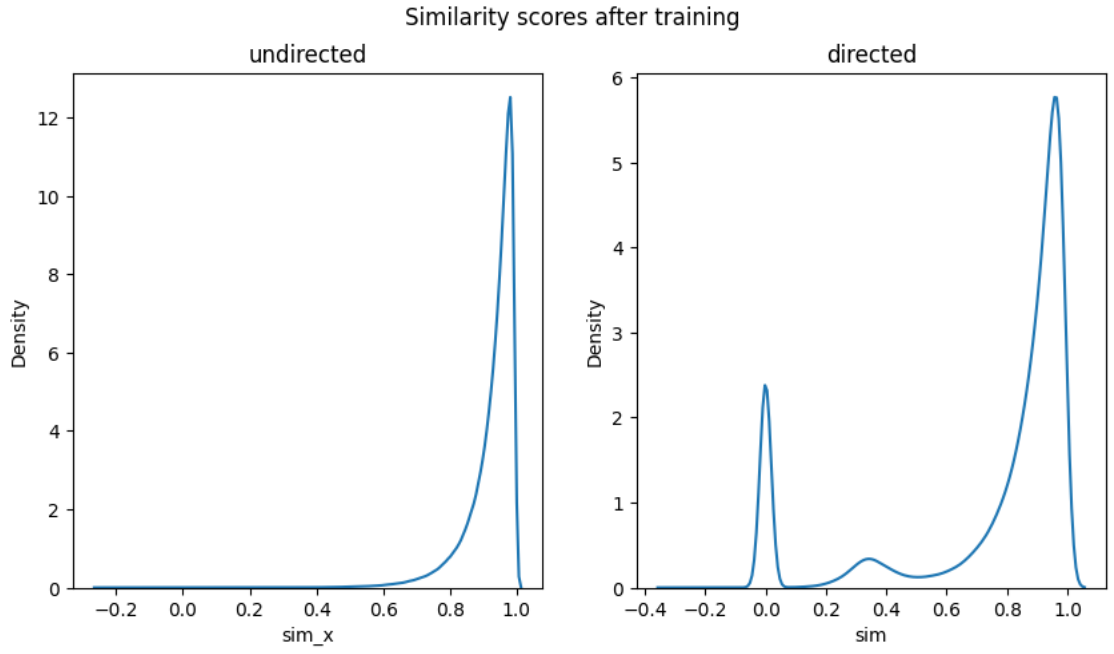

Figure 5-1. Similarity scores comparison (undirected and directed)

Also, tuning hyperparameters sometimes makes a considerable difference for deep learning or machine learning tasks. Although tweaking them does not change the model structure, it can make the model perform better or worse. While training the model in this study also,

several hyperparameters are to be set. Initially, in the preprocessing phase, the number of citations has been decided by 100 to filter the raw data according to the heuristic approach. It could be better if other attributes were considered, for instance, the published year or the list of authors, as previously mentioned, for more precise data filtering. Also, the thresholds of similarity in the method of generating positive and negative pairs for contrastive learning needed to be set. They were set by comparing their distributions. The mean of the similarity between randomly selected pairs was roughly 0.4, with a standard deviation of 0.05. Therefore, the higher and lower thresholds were chosen by 0.5 and 0.35, respectively. However, although there are some hyperparameters that are well known how to be selected, the majority of hyperparameters still strongly depend on the researcher's experience and research condition. Also, this research does not even have the ground truth with it, so it was more challenging to tune the hyperparameter. Thus, this research also used many hyperparameters from conventional knowledge as well as heuristic assumptions.

6. Conclusion

This study was designed to create a system that would help researchers find papers from different fields more easily, helping to boost collaborative and interdisciplinary research. By using a technique called graph attention networks (GATs), this research tried to build a system that could analyse large amounts of data and suggest related research papers to users.

However, the absence of a ground truth, a standard or a reference point to check if the suggestions made by the system were actually true matches, was a big challenge to conducting this study deeper. Although a method called contrastive learning was used to mitigate this issue, it could not fully solve the problem.

Working with such a large set of data was another challenge this research faced. Calculating similarities of all possible pairs was hardly possible, and transforming all the detailed information of research papers into a format the system could work with was very time-consuming. This process showed the need for more efficient ways to handle big data and suggested that future projects should consider a way that utilise smaller, more focused parts for better results not losing much information.

However, despite these limitations, this study opened up new possibilities in the way we look at large networks of information, suggesting that we might not always need to follow traditional rules, like considering the direction of connections between papers. This research is a cornerstone, encouraging other researchers to keep improving this approach.

To sum up, this study took an important first step towards building a system that can encourage collaboration and creativity in research by suggesting related works from different

fields. Although it faced challenges and showed areas that needed improvement, it pointed out the need for this research. The further work will be about refining the method, making it more intelligent and efficient, and opening up a world of collaborative possibilities for researchers around the world, one well-informed suggestion at a time.

7. Appendix

7.1.Python files

7.1.1. mongo_filtering.py

This is the Python file used for filtering data using MongoDB.

7.1.2. embedding.zip

This is the Python file used for embedding and keyword extraction.

(abstract_keyword_embedding.py, domain_embedding.py, keyword_embedding.py,

title_embedding.py)

7.1.3. GAT_training.py

This is the Python file used for neural network training.

7.2.Dataset (v14)

This is the raw data used for this research.

8. Reference

[1] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics,* vol. 36, no. 4, pp. 193-202, 1980/04/01 1980, doi: 10.1007/BF00344251.

[2] R. Van Noorden, "Interdisciplinary research by the numbers," *Nature,* vol. 525, no. 7569, pp. 306-307, 2015.

[3] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Paper recommender systems: a literature survey," *International Journal on Digital Libraries,* vol. 17, pp. 305-338, 2016.

[4] R. Szostak, "The state of the field: Interdisciplinary research," 2013.

[5] S. W. Aboelela *et al.*, "Defining interdisciplinary research: Conclusions from a critical review of the literature," *Health services research,* vol. 42, no. 1p1, pp. 329-346, 2007.

[6] S. Rawat and S. Meena, "Publish or perish: Where are we heading?," (in eng), *J Res Med Sci,* vol. 19, no. 2, pp. 87-9, Feb 2014.

[7] A. F. Repko and R. Szostak, *Interdisciplinary research: Process and theory*. Sage Publications, 2020.

[8] L. Bromham, R. Dinnage, and X. Hua, "Interdisciplinary research has consistently lower funding success," *Nature,* vol. 534, no. 7609, pp. 684-687, 2016.

[9] K. L. Hall, A. L. Vogel, G. C. Huang, K. J. Serrano, E. L. Rice, S. P. Tsakraklides, and S. M. Fiore, "The science of team science: A review of the empirical evidence and research gaps on collaboration in science," *American psychologist,* vol. 73, no. 4, p. 532, 2018.

[10] K. Huutoniemi, J. T. Klein, H. Bruun, and J. Hukkinen, "Analyzing interdisciplinarity: Typology and indicators," *Research policy,* vol. 39, no. 1, pp. 79-88, 2010.

[11] V. B. Mansilla, "Assessing expert interdisciplinary work at the frontier: an empirical exploration," *Research evaluation,* vol. 15, no. 1, pp. 17-29, 2006.

[12] Clarivate. "Authors / Researchers: Find collaborators." https://clarivate.libguides.com/authors/collaborators (accessed Sep 3, 2023).

[13] C. Pohl, "What is progress in transdisciplinary research?," *Futures,* vol. 43, no. 6, pp. 618-626, 2011.

[14] H. Bruun, J. I. Hukkinen, K. I. Huutoniemi, and J. Thompson Klein, "Promoting interdisciplinary research: The case of the Academy of Finland," 2005.

[15] C. D. McLaren and M. W. Bruner, "Citation network analysis," *International Review of Sport and Exercise Psychology,* vol. 15, no. 1, pp. 179-198, 2022.

[16] E. Garfield, "Citation indexes for science: A new dimension in documentation through association of ideas," *Science,* vol. 122, no. 3159, pp. 108-111, 1955.

[17] S. A. Greenberg, "How citation distortions create unfounded authority: analysis of a citation network," *Bmj,* vol. 339, 2009.

[18] S. Dawson, D. Gašević, G. Siemens, and S. Joksimovic, "Current state and future trends: A citation network analysis of the learning analytics field," in *Proceedings of the fourth international conference on learning analytics and knowledge*, 2014, pp. 231-240.

[19] Y. Ding, X. Liu, C. Guo, and B. Cronin, "The distribution of references across texts: Some implications for citation analysis," *Journal of Informetrics,* vol. 7, no. 3, pp. 583-592, 2013.

[20] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proceedings of the National academy of Sciences,* vol. 102, no. 46, pp. 16569-16572, 2005.

[21] C. Bergstrom, "Eigenfactor: Measuring the value and prestige of scholarly journals," *College & research libraries news,* vol. 68, no. 5, pp. 314-316, 2007.

[22] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 230-237.

[23] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The adaptive web: methods and strategies of web personalization*: Springer, 2007, pp. 291-324.

[24] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, "Graph Neural Networks in Recommender Systems: A Survey," *ACM Comput. Surv.,* vol. 55, no. 5, p. Article 97, 2022, doi: 10.1145/3535101.

[25] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The adaptive web: methods and strategies of web personalization*: Springer, 2007, pp. 325-341.

[26] P. B. Thorat, R. M. Goudar, and S. Barve, "Survey on collaborative filtering, content-based filtering and hybrid recommendation system," *International Journal of Computer Applications,* vol. 110, no. 4, pp. 31-36, 2015.

[27] J. Bennett and S. Lanning, "The netflix prize," in *Proceedings of KDD cup and workshop*, 2007, vol. 2007: New York, p. 35.

[28] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *arXiv preprint arXiv:1301.7363,* 2013.

[29] J. L. Herlocker, J. A. Konstan, and J. Riedl, "Explaining collaborative filtering recommendations," in *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, 2000, pp. 241-250.

[30] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994, pp. 175-186.

[31] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 285-295.

[32] C. Desrosiers and G. Karypis, "A comprehensive survey of neighborhood-based recommendation methods," *Recommender systems handbook,* pp. 107-144, 2010.

[33] A. Schein, R. Popescul, L. Ungar, and D. Pennock, "Generative Models for Cold-Start Recommendations," 11/01 2001.

[34] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in artificial intelligence,* vol. 2009, 2009.

[35] X. Bai, M. Wang, I. Lee, Z. Yang, X. Kong, and F. Xia, "Scientific paper recommendation: A survey," *Ieee Access,* vol. 7, pp. 9324-9339, 2019.

[36] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer,* vol. 42, no. 8, pp. 30-37, 2009.

[37] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM computing surveys (CSUR),* vol. 52, no. 1, pp. 1-38, 2019.

[38] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 173-182.

[39] N. Lao and W. W. Cohen, "Relational retrieval using a combination of path-constrained random walks," *Machine learning,* vol. 81, pp. 53-67, 2010.

[40] D. Zhou, S. Zhu, K. Yu, X. Song, B. L. Tseng, H. Zha, and C. L. Giles, "Learning multiple graphs for document recommendations," in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 141-150.

[41] Z. Huang, W. Chung, T.-H. Ong, and H. Chen, "A graph-based recommender system for digital library," in *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, 2002, pp. 65-73.

[42] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907,* 2016.

[43] M. Labonne, *Hands-On Graph Neural Networks Using Python*. Packt Publishing Ltd, 2023, pp. 106-121.

[44] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903,* 2017.

[45] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?," *arXiv preprint arXiv:2105.14491,* 2021.

[46] Q. Wang, Y. Ma, K. Zhao, and Y. Tian, "A comprehensive survey of loss functions in machine learning," *Annals of Data Science,* pp. 1-26, 2020.

[47] P. Cunningham, M. Cord, and S. J. Delany, "Supervised learning," in *Machine learning techniques for multimedia: case studies on organization and retrieval*: Springer, 2008, pp. 21-49.

[48] Z. Ghahramani, "Unsupervised learning," in *Summer school on machine learning*: Springer, 2003, pp. 72-112.

[49] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE transactions on knowledge and data engineering,* vol. 35, no. 1, pp. 857-876, 2021.

[50] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies,* vol. 9, no. 1, p. 2, 2020.

[51] I. Misra and L. v. d. Maaten, "Self-supervised learning of pretext-invariant representations," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6707-6717.

[52] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, 2006, vol. 2: IEEE, pp. 1735-1742.

[53] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, 2020: Springer, pp. 776-794.

[54] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, 2020: PMLR, pp. 1597-1607.

[55] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "ArnetMiner: extraction and mining of academic social networks," presented at the Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, Las Vegas, Nevada, USA, 2008. [Online]. Available: https://doi.org/10.1145/1401890.1402008.

[56] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the association for computational linguistics,* vol. 5, pp. 135-146, 2017.

[57] M. Grootendorst, *KeyBERT: Minimal keyword extraction with BERT*. Zenodo.

[58] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci,* vol. 6, no. 12, pp. 310-316, 2017.

[59] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289,* 2015.