Masters Thesis

GAPRS: A Graph-based Academic Paper Recommender System

William Ikenna-Nwosu
School of Computer Science
University of Galway

Supervisor(s)
Colm O'Riordan

In partial fulfillment of the requirements for the degree of
MSc in Computer Science (Artificial Intelligence)

August 31, 2023

DECLARATION I, WILLIAM IKENNA-NWOSU, hereby declare that this thesis, titled “GAPRS: A Graph-based Academic Paper Recommender System”, and the work presented in it are entirely my own except where explicitly stated otherwise in the text, and that this work has not been previously submitted, in part or whole, to any university or institution for any degree, diploma, or other qualification.

Signature: 

Student ID: 22222642

Acknowledgements

Firstly, I would like to thank God for making this possible, watching over me and guiding my journey to this point and beyond. Secondly, I would like to thank my family for their support, advice, opinions, love and belief in me to always do my best. Thirdly, I would like to thank my industry mentors Cristian Olariu and Pratik Mondkar for sharing their deep wells of experience with me and providing me with invaluable mentorship. Finally, I would like to thank my supervisor Colm O'Riordan for making himself available, approachable, being understanding, providing guidance on the research process, supporting my ideas and believing in my capabilities.

Abstract

This project proposes GAPRS, a graph-based academic paper recommender system with search engine functionality. GAPRS is targeted towards undergraduate and masters students who are writing their theses or capstone projects. It aims to be the research tool of choice when it comes to finding recommendations for academic papers that are relevant, novel, serendipitous and diverse amongst other evaluation metrics. Inspired by the founding stories of Netflix and specifically Spotify, GAPRS seeks to emulate the success bred by the aforementioned recommender systems. GAPRS is driven by the idea that the way people search for academic papers is fundamentally different to how they search for things to buy on amazon, find something to watch on Netflix or music to listen to on Spotify. The common denominator here is that each platform recognised the difference in user behaviour and centred functionality of the recommender system around it. By learning from and building on the knowledge gained by the research paper recommender system community and the wider recommender system community, both in academia and industry, GAPRS can overcome the challenges and obstacles that have plagued research paper recommender systems in the past. Apart from its potential research contribution, GAPRS has wider business value which can be seen as three-fold. First, GAPRS can be a solution to the piracy problem in academia through being a better, safer and legal software product in much the same way as Spotify became a solution to the piracy problem in music during its origin. Second, GAPRS can improve the research process, particularly in the information gathering stage, by making it more engaging, less time-consuming, more diverse, less monotonous, more collaborative and more traceable. Finally, GAPRS can lay the groundwork for monetization of academic research; disrupting the current paywall, journal fee-paying structure that academics currently face to get their work in front of the masses in the academic community and beyond. With the dawn of Generative AI, we need better workflows to help us navigate and gain insight from the plethora of content we produce. GAPRS is a step in that direction which brings many benefits and highlights significant problems in the status quo whilst being a net positive solution that brings innovation to a recently stagnant area.

Keywords: Artificial Intelligence (AI), Recommender System (RS), Information Retrieval (IR), Search Engine, Graph Theory, Network Science, Web Science, Graph Dynamics, Data Visualization (DV), Human-Computer Interaction (HCI), User Experience (UX), User Interface (UI), Explainable Artificial Intelligence (XAI), Optimization, Research Paper Recommender System (RPRS), Graph-based Recommender System, Bibliometrics, Scientometrics

GitHub Repository Link: https://github.com/wiknwo/CT5129_AIProject

Contents

Chapter 1

Introduction

In this chapter, first the motivation for GAPRS is outlined including setting the problem context, carving out the project's research contribution and demonstrating the project's business value. Next, the goals set out for this project are clearly singled out and defined. Thirdly, the research questions which follow from one another in sequence; cumulatively build on the interest of the previous question and drive the project from one stage to the next; from one aim to the next, are clearly posed and listed. Finally, the project specification is presented and this unambiguously lays out what is expected of the project at the core level which pertains to essential requirements and at the advanced level which covers contributions in excess of the core that are beyond the scope of the project.

1.1 Motivation

This section presents the motivation for GAPRS in three following subsections. The first subsection sets the project context within the tripartite relationship between academia, industry and government. The second subsection carves out the gap in the existing RPRS literature which GAPRS seeks to fill by making its research contribution. The third subsection addresses the three points which form the project's business value: solving piracy in academia, improving the research process and monetization of academic papers.

1.1.1 Project Context

This project comes during an AI summer at a time currently known as the dawn of Generative AI. Governments, industries and academia are frantically trying to estimate the extent of the effects AI can have on humanity, the world at large and are desperately trying to regulate it before it gets out of their perceived control. There is significant hype and investment in AI across all industries primarily being driven by LLMs which are the face of the Generative AI boom that is purported to be just the beginning of what is to come for advancements in AI. Where do recommender systems come into the picture? They are more influential than ever. Thanks to the year of Generative AI, the dependency on recommender systems is further cemented and interwoven into daily life. Their discerning capacity lends itself well to processing the prolific output of Generative AI along with humanity's already large data output. Recommender systems are becoming smarter, more ingrained, more ubiquitous and are directing the trajectory of entire industries and shifting popular culture on a daily basis in a way we have never seen. Netflix, Spotify, Amazon, TikTok, Google, etc. These corporations have recommender systems that dictate the entertainment industry, music industry, e-commerce industry, popular culture and organization of human knowledge respectively. However, these recommender systems were created in industry meaning that these companies will not share their techniques, algorithms or system designs openly for profitability reasons unless it is in their interest. Therefore, the lion's share of innovation in recommender systems research takes place in industry and is conducted by a handful on powerful companies. This leaves academia wanting in comparison. There is little consensus in the RS academic community about which recommendation approaches are the most promising. This project endeavours to make a dent in that universe by establishing a baseline to compare different recommendation approaches on a common problem that is well known in academia. With this effort, the hope is that all parties involved can benefit from GAPRS which strives to be a happy medium that solves multiple problems simultaneously. These problems are put forward in the following sections.

1.1.2 Project Research Contribution

This subsection identifies the gap in current RPRS research which GAPRS can fill by making its research contribution.

RPRS literature is well documented. This makes it easier to know what has been accomplished, what is still outstanding and where previous attempts at creating RPRS have faced challenges, obstacles and failure. In short, GAPRS aims to build on the abundant research on RPRS through novel contributions as set out in the aims and research questions; whilst avoiding the common mistakes made by predecessors, overcoming the challenges and obstacles which have stood in the way of progress in the RPRS community and finally, perhaps those novel contributions can be transferred to recommender systems more generally in the RS community. The aforementioned aims and research questions are detailed in the following sections while the SOTA of RPRS and its current shortcomings, challenges, common mistakes and obstacles are discussed in a later chapter.

1.1.3 Project Business Value

This subsection addresses the three points forming the business value of the project and aims to convince the reader of their importance and urgency.

Piracy in Academia: Piracy is a problem that plagues academic research in the name of free and openly accessible science. The intentions of the pirates are good but their execution of those intentions, although effective, is not a considerate solution to academia's slow efforts to make research widely available, free and open to access. With piracy, there are unintended and overlooked consequences which are borne by innocent individuals who are not the pirates' adversaries. These include researchers trying to make a living writing academic papers through innovating intellectual property. Further widening inequality in academia between institutions in the northern and southern hemispheres; although, piracy affects everyone, it affects those with less resources more by siphoning their expected profits. It is a vicious cycle. To get to the heart of this problem, it is important to know the psychology of a pirate. What makes a pirate tick. This is outside the scope of this project. Users who decide to pirate usually do so out of convenience; convenience of price, convenience of catalogue, convenience of download speed, etc. It is like going to the corner store instead of driving to the supermarket to get the same product. To compete with pirates on a technical level does not take much. Usually, their platforms are minimal, barebones, inconsistent and unreliable. Making a better software product is the first step. However, the real challenge is finding, agreeing or negotiating a suitable business model to outcompete piracy and entice users to switch for good.

Improving the Research Process: The research process is overdue for innovation. Education systems at all levels seem to take longer and are more resistant to change, even as technology improves exponentially. Researchers at all levels of qualification can agree that their process for finding relevant papers to their thesis topics at the information gathering stage looks similar at their level of qualification. with the information age upon us, the power of the internet and now the dawn of Generative AI, it is too easy to become clueless as to where to begin the search or go down the wrong rabbit hole and suddenly realise that somebody has already done exactly what they are trying to do. There is information overload and currently, there are no tools to help alleviate these problems. Above all else, it is not an enjoyable process which is the opposite of the pursuit of knowledge and search for truth. In research, curiosity is supposed to be encouraged and inquisitiveness celebrated. The joy of discovery is lost in the monotony of clicking through hundreds of pages of search results looking for a needle in a haystack.

Monetization of Academic Papers: Researchers do not receive compensation for outputting academic papers. In some cases, they have to pay to bypass a paywall to gain access to the latest academic papers which are being gatekept by the most prestigious academic journals. In other cases, they have to pay submission fees to submit their own papers to prestigious journals to get their work recognised by their peers in the peer review process. There is an apparent concentration of power in the prestigious journals in academia across all disciplines. This is purported to serve the purpose of maintaining high standards through peer-reviewed work. However, this is biased, unsustainable and poorly implemented from a scalability standpoint. Disruption to and democratization of the peer review process is of benefit to the academic community given the dawn of Generative AI. Although, this project is not about prestige in academia, it follows from the monetization of academic papers that there will be disruption to the status quo and establishment of a new equilibrium.

1.2 Research Questions

The research questions which form the heart of this project are put forward and numbered in order of increasing complexity.

RQ1. In graph-based recommender systems, which graph properties can be exploited to produce novelty, diversity and serendipity in recommendations?

RQ2. How can visualization and explainability of graph-based academic paper recommender systems be improved to help users understand how recommendations are generated?, i.e., How to indicate to users that recommendations share common attributes?

1.3 Project Specification

This section specifies the expected output of the project at the core and advanced levels in the form of research questions. The former represents contributions satisfying the essential requirements of the project whilst the latter represents contributions that build on and are in excess of the core which extend beyond the scope of the project.

Core: RQ1 and RQ2

Advanced: RQ3 and RQ4

1.4 Aims

In this section, the goals of the project are enumerated and numbered in order of research importance.

A1. Conduct a historical analysis of RPRS as a research problem, an in-depth literature review of the SOTA of RPRS in order to contextualise GAPRS in modern recommender systems research and list the common mistakes, general consensus of the RS community, challenges and shortcomings that have faced previous RPRS so that GAPRS may overcome them and succeed where others have failed.

A2. Establish a baseline for comparison of algorithms and common evaluation framework for current and future researchers in the RPRS community and wider RS community to use and easily extend.

A3. Create an open-source repository containing a framework that bundles SOTA recommendation approaches in a standardised way that is easy for current and future researchers in the RPRS community and wider RS community to use and easily extend.

A4. Build a prototype of GAPRS into an MVP.

A5. Compare the performance of GAPRS against current academic paper recommender systems and search engines according to the baseline and evaluation metrics.

Chapter 2

Groundwork

This chapter is a combination of background and related work sections. The background section focusses on the description of RPRS as a research problem and traces its history from its origin until now. The related work section lists the shortcomings, challenges, obstacles, general consensus of the wider RS community and problems that are preventing the RPRS community from making progress.

2.1 Background

In this section, the research problem tackled by this project is defined and the history of the research problem is traced from its origin to date, in an effort to capture the chain of events that lead to the current state of the RPRS community.

2.1.1 Definition of Research Problem

In this subsection, the key concepts comprising this project are defined and synthesised for the reader's understanding and convenience so the project may be self-contained.

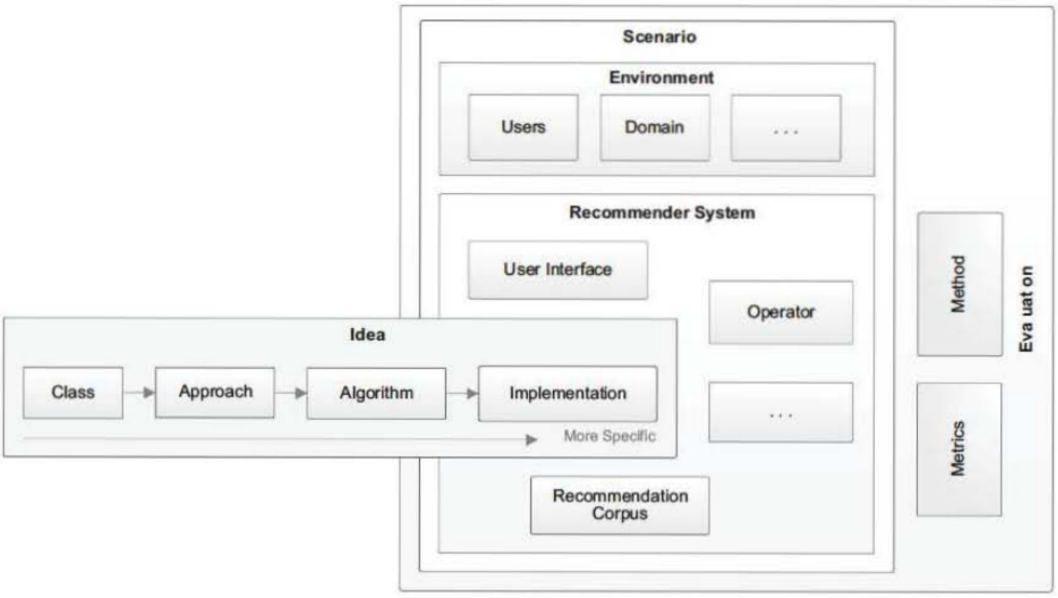

Figure 2.1: Illustration of recommender system terminology and concepts [1].

Idea: "The term *idea* refers to a hypothesis about how recommendations could be effectively generated. To differentiate how specific the idea is, we distinguish between recommendation classes, approaches, algorithms, and implementations." [1]

Recommendation Class: "We define a *recommendation class* as the least specific idea, namely a broad concept that broadly describes how recommendations might be given. For instance, the recommendation classes collaborative filtering (CF) and content-based filtering (CBF) fundamentally differ in their underlying ideas: the underlying idea of CBF is that users are interested in items that are similar to items the users previously liked. In contrast, the idea of CF is that users like items that the users' peers liked. However, these ideas are rather vague and leave room for different approaches." [1]

Recommendation Approach: "A *recommendation approach* is a model of how to bring a recommendation class into practice. For instance, the idea behind CF can be realized with user-based CF, content boosted CF and other various approaches. These approaches are quite different, but are each consistent with the central idea of CF. Nevertheless, these approaches to represent a concept are still vague and leave room for speculation on how recommendations are calculated." [1]

Recommendation Algorithm: "A *recommendation algorithm* precisely specifies a recommendation approach. For instance, an algorithm of a CBF approach would specify whether terms were extracted from the title of a document or from the body of the text, and how terms are processed (e.g., stop-word removal or stemming) and weighted (e.g., TF-IDF). Algorithms are not necessarily complete. For instance, pseudo-code might contain only the most important information and ignore basics, such as weighting schemes. This means that for a particular recommendation approach there might be several algorithms." [1]

Implementation: "Finally, the *implementation* is the actual source code of an algorithm that can be compiled and applied in a recommender system. It fully details how recommendations are generated and leaves no room for speculation. It is, therefore, the most specific idea about how recommendations might be generated." [1]

Recommender System: "A *recommender system* is a fully functional software system that applies at least one implementation to make recommendations. In addition, recommender systems feature several other components, such as a user interface, a corpus of recommendation candidates, and an operator that owns/runs the system. Some recommender systems also use two or more recommendation approaches: CiteULike, a service for discovering and managing scholarly references, lets their users choose between two approaches and Docear randomly selects one of three approaches each time users request recommendations.” [1]

Recommendation Scenario: “The *recommendation scenario* describes the entire setting of a recommender system, including the recommender system and the recommendation environment, i.e., the domain and user characteristics.” [1]

Effectiveness: “By *effectiveness*, we refer to the degree to which a recommender system achieves its objective. The objective of a recommender system from a broad perspective is to provide **good** and **useful** recommendations that make users **happy** by satisfying user needs. The needs of users vary. Consequently, some users might be interested in novel research-paper recommendations, while others might be interested in authoritative research-paper recommendations. Of course, users require recommendations specific to their fields of research. When we use the term *effectiveness*, we refer to the specific objective the evaluator wanted to measure. We use the terms *performance* and *effectiveness* interchangeably.” [1]

Evaluation: “*Evaluation* describes any kind of assessment that measures the effectiveness or merit of a concrete idea or approach. More details about research paper recommender system evaluation methods follow in [later chapters].” [1]

2.1.2 History of Research Paper Recommender Systems

This subsection provides a timeline of events starting from the creation of the first RPRS to the SOTA in RPRS research. Credit to Ritu Sharma et al [2] for putting this together.

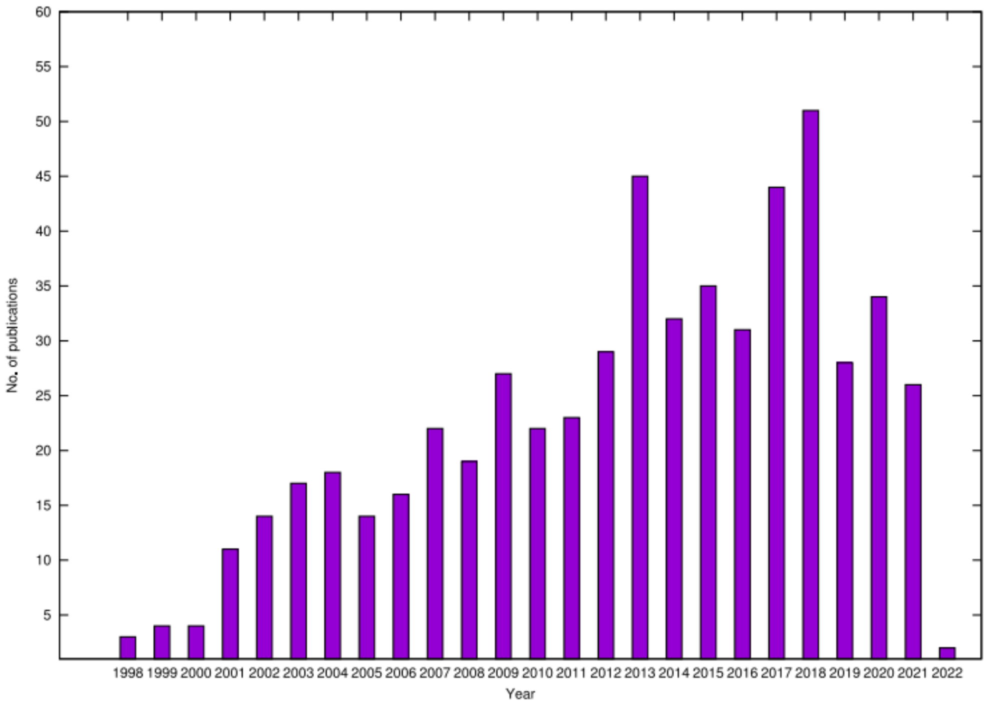

Figure 2.2: Year-wise distribution of publications in the area of RPRS from 1998-2022 [2].

Year-wise list of publications.

Year	References	Percentage (%)
1998	Giles et al. (1998), Bollacker et al. (1998) and Joaquin et al. (1998)	0.5253
1999	Bollacker et al. (1999), Lawrence et al. (1999), NEC Research Institute (1999) and Rocha (1999)	0.7005
2000	Woodruff et al. (2000), Fernández et al. (2000), Pennock et al. (2000) and Bollen and Rocha (2000)	0.7005
2001	Geisler et al. (2001), Geyer-Schulz et al. (2001a,b), Middleton et al. (2001), Rocha (2001), Mateus Rocha (2001), Fuhr et al. (2001), Schwab et al. (2001), Geyer-Schulz et al. (2001c), Lawrence et al. (2001) and Di Giacomo et al. (2001)	1.9264
2002	Renda and Straccia (2002), Nakagawa and Ito (2002), McNee et al. (2002), Middleton et al. (2002a), Straccia (2002), Middleton et al. (2002b), Geyer-Schulz and Hahsler (2002a,b), Geyer-Schulz et al. (2002), Pitkow and Pirolli (2002), Ozono et al. (2002), Ozono and Shintani (2002), Fujimaki et al. (2002) and Zeng et al. (2002)	2.4518
2003	Gross (2003), Hwang et al. (2003), Geyer-Schulz et al. (2003b,d), Böhm et al. (2003), Geyer-Schulz et al. (2003c,a), Middleton (2003), Tang et al. (2003), Callan et al. (2003), Neuhold et al. (2003), Tang and McCalla (2003a,b), Brogan (2003), Yao and Yao (2003), Yao (2003) and Luce and Giacomo (2003)	2.9772
2004	Ozono et al. (2004), Petinot et al. (2004b), Straccia and Thanos (2004), Tang and McCalla (2004a), Erosheva et al. (2004), Meyyappan et al. (2004), Petinot et al. (2004a), Vassileva (2004), Middleton et al. (2004b), Huang et al. (2004), Torres et al. (2004), Tang and McCalla (2004b,c), Nelson et al. (2004), Yao (2004), Middleton et al. (2004a), Tonta (2004) and Theobald and Klas (2004)	3.1523
2005	Watanabe et al. (2005), Cazella and Alvares (2005a), Renda and Straccia (2005), Agarwal et al. (2005), Bollen et al. (2005), Du et al. (2005), Cazella and Alvares (2005b), Yao and Yao (2005), Smeaton and Callan (2005), Tang and McCalla (2005), Konstan et al. (2005), LANL (2005), Ferran et al. (2005) and Franke and Geyer-Schulz (2005)	2.4518
2006	Chirawatkul (2006), Kang and Cho (2006), Agarwal et al. (2006), Frias-Martinez et al. (2006), Franke et al. (2006), Ozono and Shintani (2006), McNee et al. (2006), Councill et al. (2006), Hess (2006), Hess et al. (2006), Gori and Pucci (2006), Yao (2006), Bollen and Van de Sompel (2006), Ishikawa et al. (2006), Li et al. (2006) and Giles (2006)	2.8021
2007	Farooq et al. (2007), Vellino and Zeber (2007), Huang (2007), Matsatsinis et al. (2007), Mao et al. (2007), Burns et al. (2007), Bradshaw and Light (2007), Lin and Wilbur (2007), Avancini et al. (2007), Franke and Geyer-Schulz (2007), Shimbo et al. (2007), Ratprasartporn and Ozsoyoglu (2007), Sánchez et al. (2007), Strohman et al. (2007), Yin et al. (2007), Yang et al. (2007), Yang and Allan (2007), Kapoor et al. (2007), Tang and McCalla (2007b,a), Pohl (2007) and Liu and Hu (2007)	3.8529
2008	Chandrasekaran et al. (2008), Farooq et al. (2008), Henning and Reichelt (2008), Zhang et al. (2008), Lopes et al. (2008), Mönnich and Spiering (2008a), Yao (2008), Bogers and Van den Bosch (2008), Popa et al. (2008), Lakkaraju et al. (2008), Zhou et al. (2008), Nallapati et al. (2008), Mönnich and Spiering (2008b), Naak et al. (2008), Franke et al. (2008), Neumann (2008), Heß (2008), Weng and Chang (2008) and Ritchie et al. (2008)	3.3275
2009	Kodakateri Pudhiyaveetil et al. (2009), Beel et al. (2009), Naak (2009), Tang and Zhang (2009), Morales-del-Castillo et al. (2009), Tang and McCalla (2009a), Breeding (2009), Vellino (2009), Sun et al. (2009), Yang et al. (2009), Gipp and Beel (2009a), Stock et al. (2009), Gipp and Beel (2009b), Arnold and Cohen (2009), Will et al. (2009), Porcel et al. (2009), Gipp et al. (2009), Dong et al. (2009), Vivacqua et al. (2009), Naak et al. (2009), Du et al. (2009), Neumann (2009), Daud et al. (2009), Dattolo et al. (2009), Söldner et al. (2009), Tang and McCalla (2009b) and Parra and Brusilovsky (2009)	4.7285
2010	Ekstrand et al. (2010), Porcel and Herrera-Viedma (2010), Hwang et al. (2010), Choochaiwattana (2010), Bethard and Jurafsky (2010), Vellino (2010), Jomsri et al. (2010), Zhang and Li (2010), Morales-del-Castillo et al. (2010), Shaoping (2010), Logvynovskiy and Dastbaz (2010), Sugiyama and Kan (2010), Cui et al. (2010), Wang et al. (2010), Guan et al. (2010), Lao and Cohen (2010), He et al. (2010), Kataria et al. (2010), Beel and Gipp (2010), Martín et al. (2010), Pan and Li (2010) and Zhang and Koppaka (2010)	3.8529
2011	Wang and Blei (2011), Sugiyama and Kan (2011), He et al. (2011), Pitigala et al. (2011), Pera and Ng (2011), Liang et al. (2011b), Nascimento et al. (2011), Ferrara et al. (2011), Lao and Cohen (2011), Beel et al. (2011), Vaughan (2011), Liang et al. (2011a), Baez et al. (2011), Liu and Belkin (2011), Thomas et al. (2011), Zhou et al. (2011), Chen et al. (2011), Uchiyama et al. (2011), Gottwald (2011), Lu et al. (2011), Ohta et al. (2011), Amini et al. (2011) and Nagori and Aghila (2011)	4.0280
2012	Jain (2012), Zarrinkalam and Kahani (2012a,b), Küçüktunç et al. (2012c), Zhou et al. (2012a), Gautam and Kumar (2012), Hong et al. (2012), Jiang et al. (2012a), Mishra (2012), Winoto et al. (2012), Akbar et al. (2012), Patton et al. (2012), Lao (2012), Wu et al. (2012), Küçüktunç et al. (2012), Jiang et al. (2012b), Küçüktunç et al. (2012b), Huynh et al. (2012), Soulier et al. (2012), Bancu et al. (2012), Tang and Zeng (2012), Kuberek and Mönnich (2012), Doerfel et al. (2012), Lao and Cohen (2012), Xia et al. (2012), He et al. (2012), Yu et al. (2012), Huang et al. (2012) and Bollen and Van De Sompel (2012)	5.0788
2013	Sugiyama and Kan (2013), Beel et al. (2013f), Chen et al. (2013), Vellino (2013), Beel et al. (2013b), Caragea et al. (2013), Zarrinkalam and Kahani (2013a), Lai and Zeng (2013), Yang et al. (2013), Sun et al. (2013), Rokach et al. (2013), Yang and Lin (2013), Meng et al. (2013), Beel et al. (2013e), Hong et al. (2013a), Lee et al. (2013), Nunes et al. (2013), Guo and Chen (2013), Oh et al. (2013), Küçüktunç et al. (2013a), Stock et al. (2013), Yan et al. (2013), Kim (2013), Manouselis and Verbert (2013), Chakraborty and Chakraborty (2013), De Nart et al. (2013), Yao et al. (2013), Hong et al. (2013b), Wang et al. (2013), Tian and Jing (2013), Beel et al. (2013c), Tao et al. (2013), Küçüktunç et al. (2013b), Kucuktunc (2013), Li et al. (2013), Sitthisarn and Rattanabundan (2013), Küçüktunç et al. (2013c,d), Alotaibi and Vassileva (2013), Zarrinkalam and Kahani (2013b), Li and Xiao (2013), Yan (2013), Amini et al. (2013), Jin et al. (2013) and Zhang et al. (2013)	7.8809
2014	Le Anh et al. (2014), Livne et al. (2014), Beel et al. (2014a), Pera and Ng (2014), Sun et al. (2014), Huang et al. (2014), De Nart and Tasso (2014), Tejeda-Lorente et al. (2014b), Philip et al. (2014), Zhou et al. (2014), Amjad et al. (2014), Amini et al. (2014), Ren et al. (2014), Tantanasiriwong and Haruechaiyasak (2014), Guo and Chen (2014), Tang et al. (2014), Kurtz and Henneken (2014), Xia et al. (2014), Liu et al. (2014b,a), Shirude and Kolhe (2014), Tejeda-Lorente et al. (2014a), Beel et al. (2014b), Xue et al. (2014), Akbar et al. (2014), Yang et al. (2014), Ha et al. (2014), Hajra et al. (2014), Jardine and Teufel (2014), Duma and Klein (2014), Omisore and Samuel (2014) and Omisore (2014)	5.6042
2015	Liu and Yang (2015), Tran et al. (2015), Sugiyama and Kan (2015b), Chakraborty et al. (2015), Amini et al. (2015), Kim and Chen (2015), Champiri et al. (2015), Meilian et al. (2015), Sinha et al. (2015), Hanyurwimfura et al. (2015), Liu et al. (2015b), Lee et al. (2015), Liu et al. (2015a), Verma and Dey (2015), Sesagiri Raamkumar et al. (2015), Alotaibi and Vassileva (2015), Fu et al. (2015), Vellino (2015), Tejeda-Lorente et al. (2015), Asabere et al. (2015), Patil and Ansari (2015), Steinert et al. (2015), Beel et al. (2015a,b), Beel (2015), Küçüktunç et al. (2015), Lu et al. (2015), Hsiao et al. (2015), Alzoghbi et al. (2015), Gao (2015), Sugiyama and Kan (2015a), Jiang et al. (2015), Ha et al. (2015), Beel and Langer (2015) and Huang et al. (2015)	6.1296

Figure 2.3: Year-wise list of publications in the area of RPRS from 1998-2022 [2].

Year	References	Percentage (%)
2016	Beel et al. (2016a), Zhao et al. (2016), Steinert and Hoppe (2016), Bean (2016), Raamkumar et al. (2016), Paraschiv et al. (2016), Totti et al. (2016), West et al. (2016), Jadhav and Wankhade (2016), Wang et al. (2016), Alotaibi (2016), Amami et al. (2016), Wolfram (2016), Liu et al. (2016), Shirude and Kolhe (2016), Alotaibi and Vassileva (2016), Rúbio and Gulo (2016), Igbe and Ojokoh (2016), Vargas et al. (2016), Nishioka and Scherp (2016), Allen et al. (2016), Lee et al. (2016), Dhanda and Verma (2016), Tsolakidis et al. (2016), Xia et al. (2016), Chakraborty and Narayanam (2016), Alzoghbi et al. (2016), Duma et al. (2016b), Chakraborty et al. (2016), Duma et al. (2016a) and Beel et al. (2016b)	5.4291
2017	Beel (2017a), Sharma et al. (2017b), Haruna et al. (2017), Kazemi and Abhari (2017), Amami et al. (2017), Wang et al. (2017b), Langer and Beel (2017), Liu and Chien (2017), Xia et al. (2017), Hristakeva et al. (2017), Guo et al. (2017b), Ravi et al. (2017), Sharda and Dawgotra (2017), Guo et al. (2017a), Hassan (2017), Feyer et al. (2017), AlShebli et al. (2017), Tsolakidis et al. (2017), Beel et al. (2017a), Neethukrishnan and Swaraj (2017), Habib and Afzal (2017), Yin and Li (2017), Beel and Dinesh (2017), Chaitanya and Singh (2017), Guseva et al. (2017), Gupta and Varma (2017), Al-Natsheh et al. (2017), Wang et al. (2017a), Beel et al. (2017c), Hwang et al. (2017), Magara et al. (2017), Sahijwani and Dasgupta (2017), Raamkumar et al. (2017), Beel (2017b), Beierle et al. (2017), Jia and Saule (2017), Roy (2017), Anand et al. (2017), Ebesu and Fang (2017), Beel et al. (2017b), Al Alshaikh et al. (2017c,a,b) and Ahmad and Afzal (2017)	7.7058
2018	Kaya (2018), Yang et al. (2018), Collins et al. (2018), White (2018), Li et al. (2018a), Dai et al. (2018b,a), Liang et al. (2018), Mu et al. (2018), Wenige and Ruhland (2018), Cai et al. (2018b), Zhang et al. (2018b), Bulut et al. (2018), Cai et al. (2018a), Fernández-Isabel et al. (2018), Haruna et al. (2018), Son and Kim (2018), Zhao et al. (2018), Jiang et al. (2018), dos Santos and Machado (2018), Ayala-Gómez et al. (2018), Jia and Saule (2018), Wang et al. (2018a), Supriyanto et al. (2018), Mayr et al. (2018), Ollagnier et al. (2018b), Beel et al. (2018b), Sesagiri Raamkumar and Foo (2018), Chughtai et al. (2018), Ziegler and Shrake (2018), Lu et al. (2018), Bertin and Atanassova (2018), Haruna and Ismail (2018), Nishioka and Ogata (2018), Porcel et al. (2018), Zhang et al. (2018a), Beel et al. (2018a), Magara et al. (2018), Ollagnier et al. (2018a), Kong et al. (2018), Beel et al. (2018d), Kobayashi et al. (2018), Bhagavatula et al. (2018), Dinesh (2018), Beel et al. (2018c), Wang et al. (2018b), Raamkumar et al. (2018), Ullah (2018), Al Alshaikh (2018), Färber et al. (2018) and Li et al. (2018b)	8.9317
2019	Dai et al. (2019), Li et al. (2019a), Waheed et al. (2019), Bai et al. (2019), Cai et al. (2019), Yang et al. (2019c), Beel et al. (2019), Li et al. (2019b), Rahdari and Brusilovsky (2019), Yang et al. (2019b), Samad (2019), Alshareef (2019), Collins and Beel (2019), Hassan et al. (2019), Maake et al. (2019a), Murali et al. (2019), Kanakia et al. (2019), Habib and Afzal (2019), Ma et al. (2019), Ma and Wang (2019), Li and Zou (2019), Chen et al. (2019b), Maake et al. (2019b), Chen et al. (2019a), Zhao et al. (2019), Yang et al. (2019a), Duma (2019) and Nishioka et al. (2019)	4.9036
2020	Ali et al. (2020b), Färber and Jatowt (2020), Färber and Sampath (2020), Färber et al. (2020), Haruna et al. (2020), Liu et al. (2020), Alfarhood and Cheng (2020), Jeong et al. (2020), Cohan et al. (2020), Ma et al. (2020), Wang et al. (2020a), Nogueira et al. (2020b), Ostendorff (2020), Ali et al. (2020a,c), Guo et al. (2020), Bulut et al. (2020), Sakib et al. (2020), Du et al. (2020), Khadka (2020), Medić and Šnajder (2020), Nair et al. (2020), Nishioka et al. (2020), Li et al. (2020), Medić and Snajder (2020), Saier and Färber (2020), Nogueira et al. (2020c), Tao et al. (2020), Nogueira et al. (2020a), Khadka et al. (2020), Alkhatib and Rensing (2020), Dai et al. (2020), Wang et al. (2020b) and Choi et al. (2020)	5.9545
2021	Zhu et al. (2021), Ali et al. (2021b), Qiu et al. (2021), Ali et al. (2021a), Tang et al. (2021), Wang et al. (2021), Dai et al. (2021), Kang et al. (2021), Chaudhuri et al. (2021), Ali et al. (2021d), Kieu et al. (2021a), Xie et al. (2021), Kieu et al. (2021b), Boudin (2021), Zhang et al. (2021a), Portenoy (2021), Ali et al. (2021c), Sakib et al. (2021), Nair et al. (2021), Hao et al. (2021), Shyani (2021), Meyer et al. (2021), Chen (2021), Zhang et al. (2021b), Ma et al. (2021) and Bhowmick et al. (2021)	4.5534
2022	Zhang and Zhu (2022) and Nair et al. (2022)	0.3502

Figure 2.4: Year-wise list of publications in the area of RPRS from 1998-2022 [2].

2.2 Literature Review

This section captures the major challenges facing new contributions to the RPRS space from the perspective of researchers in the community and summarizes related research fields from which RPRS may draw inspiration. Challenges are organised by research topic and community and enumerated for clearer reading.

2.2.1 Recommender Systems Research Community Consensus

The wider RS community's general consensus on the limitations facing their research is noted in this subsection.

1. **Lack of Evaluation:** "[T]he Recommender Systems research community is facing a crisis where a significant number of papers present results that contribute little to collective knowledge [. . .] often because the research lacks the [. . .] evaluation to be properly judged and, hence, to provide meaningful contributions." [1]

2. **Dependency on Offline Evaluation Experiments:** "The research community's dependence on offline experiments [has] created a disconnect between algorithms that score well on accuracy metrics and algorithms that users will find useful." [1]

3. **Non-Reproducibility of Experiments and Results:** "I think that it is widely agreed in the community that this [non-reproducibility] is just the way things are – if you want a recommender system for a specific application, there is no better way than just test and optimize a number of alternatives. This probably cannot be avoided – there will never be a sufficient set of experiments that would allow "practitioners" to make decisions

without running through this optimization process for their specific app and dataset." [1]

2.2.2 Problems facing Recommender Systems Research Community

The problems opposing RS community research are listed along with proposed solutions.

Impossible to determine most Effective Recommendation Approaches (Problem 1): "First, it is currently not possible to determine the most effective recommendation approaches. If we were asked which recommendation approach to apply in practice or to use as baseline, there is no definite answer. We do not even have a clue as to which of the approaches might be most promising. This problem mainly relates to poor experimental design and lack of information, which includes inadequate evaluations and too little information given by the authors." [1]

Frameworks and Best Practice Guidelines for Evaluation of RPRS (Solution 1): "To solve this problem, we believe it is crucial that the community discusses and develops frameworks and best practice guidelines for the evaluation of research-paper recommender-systems. This should include an analysis and discussion of how suitable offline evaluations are; to what extent datasets should be pruned comparable to existing TREC datasets; the minimum number of participants in user studies; and which factors influence the results of evaluations (e.g., user demographics). Ideally, a set of reference approaches would be implemented that could be used as baselines. In addition, more details on implementations are needed, based on a discussion of the information needed in research articles. It is crucial to find out why seemingly minor differences in algorithms or evaluations lead to major variations in the evaluation results. As long as the reasons for these variations are not found, scholars cannot rely on existing research results because it is not clear whether the results can be reproduced in a new recommendation scenario." [1]

Unused Potential in RS Research (Problem 2): "Second, we identified unused potential in recommender systems research. This problem has two root causes [split into problem 2a and problem 2b]." [1]

Research Results often not Transferred into Practice or Considered by Peers (Problem 2a): "Research results are often not transferred into practice or considered by peers. Despite the large number of research articles, just a handful of active recommender systems exist, and most of them apply simple recommendation approaches that are not based on recent research results. As such, the extensive research conducted from 1998 to 2013 apparently had a rather minor impact on research-paper recommender systems in the real world. Additionally, developers of several of the active recommender systems do not engage in the research community or publish information about their systems. Some researchers also seem to be unaware of developments in related research domains, such as user modelling, scientometrics, and the reviewer-assignment problem. In addition, the major co-author groups in the domain of research-paper recommender systems do not cooperate much with each other. One reason for some of these problems might be a relatively short-lived interest in the research field. Most authors (73%) published only a single paper on research-paper recommender systems." [1]

Establish Researcher Collaboration Platform (Solution 2a): "One step towards using the full potential of research-paper recommender systems could be to establish a platform for researchers to collaborate, work on joint publications, communicate ideas, or to establish conferences or workshops focusing solely on research-paper recommender systems." [1]

User Satisfaction might not depend only on Accuracy (Problem 2b): "The majority of authors did not take into account that user satisfaction might depend not only on accuracy but also on factors such as privacy, data security, diversity, serendipity, labelling, and presentation. The operator perspective was widely neglected. Information about runtime was provided for 10% of the approaches. Complexity was covered by very few authors and the costs of running a recommender system were reported by a single article. We also observed that many authors neglect the user-modelling process: 81% of the approaches made their users provide some keyword, text snippets, or a single input paper to represent their information need. Few approaches automatically inferred information needs from the users' authored, tagged, or otherwise connected papers." [1]

Open-Source Recommender Framework (Solution 2b): "An open-source recommender framework containing the most promising approaches could help transfer the research results into practice. Such a framework would also help new researchers in the field access a number of baselines they could compare their own approaches with. A framework could either be built from scratch, or be based on existing frameworks such as MyMediaLite, LensKit, Mahout, Duine, RecLab Core, easyrec, or Recommender101. Finally, the community could benefit from considering research results from related disciplines. In particular, research in the area of user modelling and scientometrics appears promising, as well as research from the general recommender-systems community about aspects beyond accuracy." [1]

2.2.3 Open Challenges in the Research Paper Recommender Systems Community

These are the current open challenges in the RPRS community. Challenges 1-6 are taken from Joeran Beel et al [1] while challenges 7-16 are taken from Christin Katharina Kreutz and Ralf Schenkel [3].

1. **Neglect of User Modelling:** "A fundamental part of generating recommendations is the user modelling process that identifies a user's information needs [263]. Ideally, a recommender system identifies the needs automatically by inferring the needs from the user's item interactions. Alternatively, the recommender system asks users to specify their needs by providing a list of key words or through some other method. However, in this case a recommender system becomes very much like a search engine and loses one of its main features, namely the capability to recommend items even if users do not know exactly what they need."

   (a) **User Model Size:** "Another important question in user modelling is the user model size. While in search, user models (i.e., search queries) typically consist of a few words; user models in recommender systems may consist of hundreds or even thousands of words."

2. **Focus on Accuracy:** "The research-paper recommender-system community places a strong focus on accuracy and seems to assume that an accurate recommender system will lead to high user satisfaction. However, outside the research-paper recommender-system community, it is agreed that many aspects beyond accuracy affect user satisfaction."

(a) **Users' Tasks:** "Torres et al. from TechLens' considered a user's current task in the recommendation process. The authors distinguished between users who wanted to receive authoritative recommendations and novel recommendations [117]. Uchiyama et al found that students are not typically interested in papers similar to their input paper."

(b) **Diversity:** "Vellino et al. measured diversity as the number of journals from which articles were recommended [119]. If recommendations were all from the same journals, diversity was zero."

(c) **Layout:** "Farooq et al. from CiteSeer analyzed which information users wanted to see when receiving recommendations in RSS feeds [138]. They found that the information to display varies for the type of recommendation."

(d) **User Characteristics:** "For our own recommender system Docear, we found that researchers who registered tended to have higher click through rates than unregistered users (6.95% vs. 4.97%) [8]. In addition, older users seem to have higher average click-through rates (40–44 years: 8.46%) than younger users (20–24 years: 2.73%) [8]."

(e) **User Duration:** "Middleton et al. reported that the longer someone used the recommender system; the lower click-through rates became [94]. Jack reports the opposite, namely that precision increased over time ($p = 0.025$ in the beginning, $p = 0.4$ after 6 months) and depended on a user's library size ($p = 0.08$ for 20 articles and $p = 0.40$ for 140 articles) [58]. We showed that it might make sense to be "persistent" and show the same recommendations to the same users multiple times—even recommendations that users had clicked before were often clicked again [6]."

(f) **Recommendation Medium:** "User satisfaction also depends on the medium through which recommendations are made. Middleton et al. report that recommendations via email received half the click-through rate as the same recommendations delivered via a website."

(g) **Relevance and Profile Feedback:** "Relevance feedback is a common technique to improve recommendations [263] but it is widely ignored in the research-paper recommender-system community. Middleton et al. showed that profile feedback is better than relevance feedback: allowing users to edit their user models is more effective than just learning from relevance feedback."

3. **Translating Research into Practice:** "Translating research into practice is a current challenge in the research paper recommender system community. Out of the large number of proposed approaches in the field, 24 research-paper recommender systems could be used by users in practice (Table 9). Of these 24 recommender systems, eight (33%) never left the prototyping stage—and today only one of the prototypes is still publicly available. Of the remaining recommender systems, four are offline (25%), five are no longer actively maintained (31%), while seven are running and actively maintained (44%). Of the seven active recommender systems, four operators are involved with the recommender-system research community (see footnote 29) and publish information about their systems."

4. **Persistence and Authorities:** "One reason the research is not transferred directly into practice might be a lack of persistence and authorities in the field. Of the 276 authors who authored the 185 articles, 201 (73%) published just one article (Fig. 5). 15 authors (5%) published five or more articles, but of these authors, several were co-authors publishing the same articles. This means that there are only a few groups that consistently publish research

in the field of research-paper recommender systems."

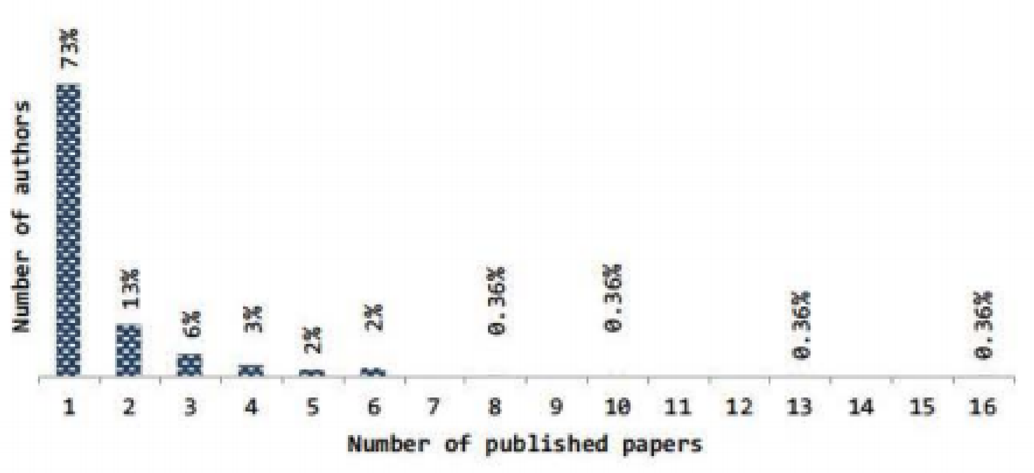

Figure 2.5: Number of papers published per author in the RPRS field from 1998-2013 [1].

5. **Co-operation:** "Most articles were authored by multiple authors: the majority of articles had two (26%), three (26%) or four authors (18%) (Fig. 6). 17% of articles were authored by a single researcher. On first glance, these numbers indicate a high degree of collaboration. However, we noticed that between different co-author groups little cooperation took place. The closest cooperation we could identify was that Giles was part of a committee for a thesis that Cohen supervised [74]. **Leading authors of different research groups did not co-author articles together. Co-author groups frequently seemed to work alone and did not always build on the results of the work done by their peers.** Among the reviewed articles, it rarely happened that authors reported to have built their novel approach based upon an existing approach. This lack of cooperation also becomes apparent when looking at the citations. Although some of the reviewed articles gained many citations, these citations usually resulted from articles outside the research-paper recommender domain. For instance, the

paper "Learning multiple graphs for document recommendations" attracted 63 citations since 2008 [127]. From these citations, only three were made by the reviewed articles. Another article, from the BibTiP developers, gained 24 citations since 2002 [32]. From the 24 citations, ten were self-citations and none was from the reviewed articles. Both examples are typical for most of the reviewed articles. One of the few articles that is constantly cited in the research-paper recommender community is an article about TechLens, which accumulated more than 100 citations [117]. However, many authors cited the article for authoritative reasons. In the citing papers, TechLens is mentioned but, with few exceptions, its approaches are neither adopted nor used as a baseline."

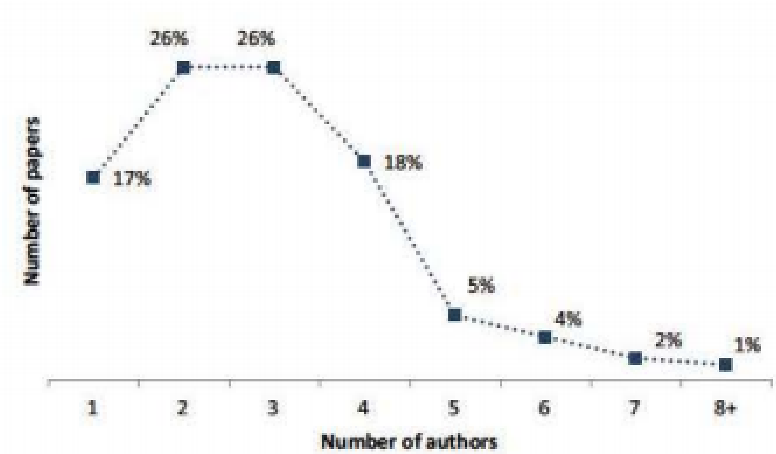

Figure 2.6: Number of authors per reviewed paper in the RPRS field from 1998-2013 [1].

6. **Information Scarcity:** "Most authors provided little detail about their approaches, which makes a re-implementation difficult, if not impossible. For instance, for 24 of the 34 content-based filtering approaches (71%), the authors did not report the weighting scheme they used (e.g., TF-IDF). The feature representation model (e.g., vector space model) was not re-

ported for 20 approaches (59%). Whether stop words were removed was not reported for 23 approaches (68%) . For 16 approaches (47%), no information was given on the fields the terms were extracted from (e.g., title or abstract). This information scarcity means, when an evaluation reports promising results for an approach, that other researchers would not know how to re-implement the approach in detail. If they tried, and guessed the specifics of an approach, the outcome would probably differ significantly from the original. This might cause problems in replicating evaluations, and reproducing research results and hinders the re-implementation and application of promising approaches in real-word recommender systems."

7. **Cold Start:** "Pure collaborative filtering systems encounter the cold start problem."

8. **Sparsity or Reduce Coverage:** ""Bai et al. [9] state the user-paper-matrix being sparse for collaborative filtering based approaches. Shahid et al. [92] also mention this problem as the reduce coverage problem. This trait makes it hard for approaches to learn relevancy of infrequently rated papers."

9. **Scalability:** "The problem of scalability was described by Bai et al. [9]. They state paper recommendation systems should be able to work in huge, ever-expanding environments where new users and papers are added regularly."

10. **Privacy:** "The problem of privacy in personalised paper recommendation is described by Bai et al. [9]. Shahid et al. [92] also mention this as a problem occurring in collaborative filtering approaches. An issue is encountered when sensitive information such as habits or weaknesses that users might

not want to disclose is used by a system. This leads to users' having negative impressions of systems. Keeping sensitive information private should therefore be a main goal."

11. **Serendipity:** "Serendipity is described by Bai et al. [9] as an attribute often encountered in collaborative filtering [16]. Usually, paper recommender systems focus on identification of relevant papers even though also including not obviously relevant ones might enhance the overall recommendation. Junior researchers could profit from stray recommendations to broaden their horizon, senior researchers might be able to gain knowledge to enhance their research. The ratio between clearly relevant and serendipitous papers is crucial to prevent users from losing trust in the recommender system."

12. **Unified Scholarly Data Standards:** "Different data formats of data collections is mentioned as a problem by Bai et al. [9]. They mention digital libraries containing relevant information which needs to be unified in order to use the data in a paper recommendation system. Additionally, the combination of datasets could also lead to problems."

13. **Synonymy:** "Shahid et al. [92] describe the problem of synonymy encountered in collaborative filtering approaches. They define this problem as different words having the same meaning."

14. **Gray Sheep:** "Gray sheep is a problem described by Shahid et al. [92] as an issue encountered in collaborative filtering approaches. They describe it as some users not consistently (dis)agreeing with any reference group."

15. **Black Sheep:** "Black sheep is a problem described by Shahid et al. [92] as an issue encountered in collaborative filtering approaches. They describe it as some users not (dis)agreeing with any reference group."

16. **Shilling Attack:** “Shilling attacks are described by Shahid et al. [92] as a problem encountered in collaborative filtering approaches. They define this problem as users being able to manually enhance visibility of their own research by rating authored papers as relevant while negatively rating any other recommendations.”

2.2.4 Emerging Open Challenges in the Research Paper Recommender Systems Community

These are the emerging open challenges in the RPRS community according to Christin Katharina Kreutz and Ralf Schenkel [3].

1. **User Evaluation:** “Paper recommendation systems targeted at users should always contain a user evaluation with a description of the composition of participants. Paper recommendation is always targeted at human users. But oftentimes an evaluation with real users to quantify users’ satisfaction with recommended publications is simply not conducted [84]. Conducting huge user studies is not feasible [38]. So sometimes user data to evaluate with is fetched from the presented datasets [39, 88] or user behaviour is artificially emulated [1, 19, 57].”

2. **Target Audience:** “Current works mostly fail to clearly characterise the intended users of a system altogether and the varying interests of different types of users are not examined in their evaluations. Definition and consideration of a specific target audience for an approach and evaluation with members of this audience. If there is no specific person group a system should suit best, this should be discussed, executed and evaluated accordingly.”

3. **Recommendation Scenario:** “Suggested papers from an approach should

either be ones to read [109, 44], to cite or fulfil another specified information need. The clear description of the recommendation scenario is important for comparability of approaches as well as the validity of the evaluation."

4. **Fairness/Diversity:** ""Anand et al [8] define fairness as the balance between relevance and diversity of recommendation results. Only focusing on fit between the user or input paper and suggestions would lead to highly similar results which might not be vastly different from each other. Having diverse recommendation results can help cover multiple aspects of a user query instead of only satisfying the most prominent feature of the query [8]. In general more diverse recommendations provide greater utility for users [76]. Ekstrand et al. [31] give a detailed overview of current constructs for measuring algorithmic fairness in information access and describe possibly arising problems in this context."

5. **Complexity:** "Paper recommendation systems tend to become more complex, convoluted or composed of multiple parts. We observed this trend by regarding the classification of current systems compared to previous literature reviews (see Section 3.3.1). While systems' complexity increases, users' interaction with the systems should not become more complex. If an approach requires user interaction at all, it should be as simple as possible. Users should not be required to construct sophisticated knowledge graphs [109] or enter multiple rounds of key words for an approach to learn their user profile."

6. **Explainability:** "The conceptualisation of recommendation systems which comprehensibly explain their users why a specific paper is suggested."

7. **Public Dataset:** "Utilisation of publicly available datasets in the evaluation of paper recommendation systems."

8. **Comparability:** "Evaluation of paper recommendation approaches, even those which are applicable in a wider context, should always be against at least one pa per recommendation system to clearly report relevance of the proposed method in the claimed context."

2.2.5 Open Challenges in Graph Learning Recommender Systems

This is a list of relevant open challenges from the GLRS community that could shed light on novelty in RPRS taken from Shoujin Wang et al [4].

1. **Self-Evolutionary Recommender System with Dynamic Graph Learning:** "In real-world RS, users, items and the interactions between them, keep evolving over time [Wang et al., 2019d]. This originates graphs with dynamic topology, and such dynamics could have direct impacts on the user and requirement modeling, causing even a clear change of recommendation results over time. However, this issue is still underestimated in existing GLRS. Therefore, it is a promising future research direction to design self-evolutionary RS over dynamic graphs."

2. **Explainable Recommender System with Causal Graph Learning:** "Causal inference is a major technique used to discover the causal relations between objects or actions. Although some progress has been achieved in explainable RS, we are is still far away from achieving a complete understanding of the reasons and intents behind user choice behaviours, which is a critical step to make reliable and explainable recommendations [Zhang and Chen, 2018]. To this end, it is another promising direction to construct explainable RS with causal graph learning."

3. **Deep Reinforcement Learning:** Employing DRL techniques on graphs is another promising research direction that could yield novel results.

2.2.6 Related Research Fields

This is a summary of research fields related to RPRS that the project can draw inspiration from in the form of novelty as stated by Joeran Beel et al [1].

Academic Search Engines: "Research on academic search engines deals with calculating relevance between research papers and search queries [227- 229]. The techniques are often similar to those used by research-paper recommender systems. In some cases, recommender systems and academic search engines are even identical. As described later, some recommender systems require their users to provide keywords that represent their interests. In these cases, research-paper recommender systems do not differ from academic search engines where users provide keywords to retrieve relevant papers. Consequently, these fields are highly related and most approaches for academic search engines are relevant for research-paper recommender systems."

Reviewer Assignment Problem: "The reviewer assignment problem targets using information-retrieval and information-filtering techniques to automate the assignment of conference papers to reviewers [230]. The differences from research-paper recommendations are minimal: in the reviewer assignment problem, a relatively small number of paper submissions must be assigned to a small number of users, i.e., reviewers; research-paper recommender systems recommend a few papers out of a large corpus to a relatively large number of users. However, the techniques are usually identical. The reviewer assignment problem was first addressed by Dumais and Nielson in 1992 [230]; 6 years before Giles et al. introduced the first research-paper recommender system. A good survey on the reviewer assignment problem was published by Wang et al. [231]."

Scientometrics: "Scientometrics deals with analyzing the impact of researchers, research articles and the links between them. Scientometrics researchers use several techniques to calculate document relatedness or to rank a collection of articles. Some of the measures- h-index [232], co-citation strength [233] and bibliographic coupling strength [234]- have also been applied by research-paper recommender systems [13, 123, 126]. However, there are many more metrics in scientometrics that might be relevant for research-paper recommender systems [235]."

User Modelling: "User modeling evolved from the field of Human Computer Interaction. One thing user modeling focuses on is reducing users' information overload making use of users' current tasks and backgrounds [236]. User modeling shares this goal with recommender systems, and papers published at the major conferences in both fields (UMAP14 and RecSys 15) often overlap. User modeling is a central component of recommender systems because modeling the users' information needs is crucial for providing useful recommendations. For some comprehensive surveys about user modeling in the context of web personalization, refer to [237,238]."

Other: "Other related research fields include book recommender systems [239], educational recommender systems [240], academic alerting services [241], expert search [242], automatic summarization of academic articles [243-245], academic news feed recommenders [246,247], academic event recommenders [248], venue recommendations [249], citation recommenders for patents [250], recommenders for academic datasets [251], and plagiarism detection. Plagiarism detection, like many research-paper recommenders, uses text and citation analysis to identify similar documents [252-254]. Additionally, research relating to crawling the web and analyzing academic articles can be useful for building research-paper recommender systems, for instance, author name extraction and disambiguation [255], title extraction [256-260], or citation extraction and matching [261]. Finally, most of the research on content-based [262] or collaborative filtering [263,264] from other domains, such as movies or news, can also be relevant for research-paper recommender systems."

Chapter 3

Data

In this chapter, the metrics, algorithms, datasets, direct competitors, indirect competitors and APIs being used in this project are gathered, enumerated, described and justified. In addition to this, any pre-processing steps that are made to the data are explained for easy reproduction of data.

3.1 APIs

This section contains complete analyses of the APIs involved in this project covering description, justification and limitations of each API.

3.1.1 OpenAlex

OpenAlex is a fully open index of scholarly works, authors, venues, institutions, and concepts as stated in the title of the paper by Jason Priem et al. It is a relatively new API which launched in 2022 as a continuation of Microsoft Academic which closed around the same time. As a result, there are not many researchers who are aware of its existence or that have used it in their research. So, it is of paramount importance to describe it in detail. Thankfully, Jason Priem et al. have already described OpenAlex in detail [5]. Therefore, an overview of the API's capabilities will suffice. The following is a synopsis of information available on the OpenAlex API web page.

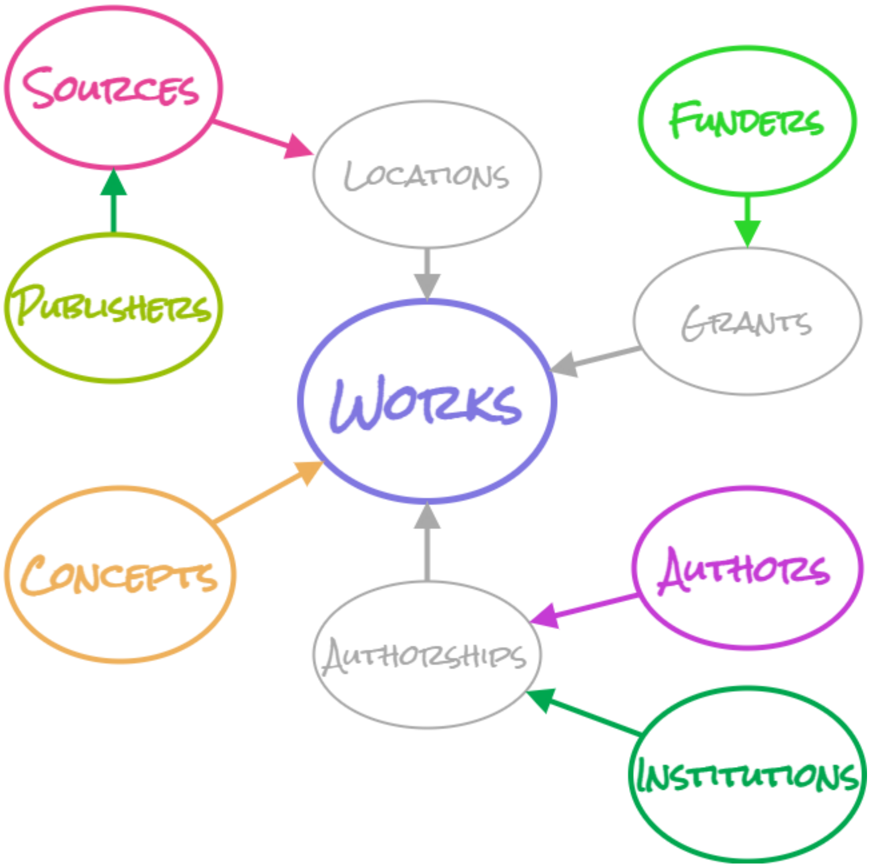

Figure 3.1: OpenAlex entities overview diagram [5].

Description: OpenAlex is a fully open catalog of the global research system. It is named after the ancient Library of Alexandria and made by the nonprofit OurResearch. The OpenAlex API describes scholarly entities and how these entities are connected to each other. Types of entities include works, authors, sources, institutions, concepts, publishers and funders. Together, these make a huge web or more technically, heterogeneous directed graph, of hundreds of millions of entities and billions of connections between them all.

- **Works:** Scholarly documents like journal articles, books, datasets, and theses.
- **Authors:** People who create works.
- **Sources:** Where works are hosted (such as journals, conferences, and repositories).
- **Institutions:** Universities and other organizations to which authors claim affiliations.
- **Concepts:** Topics assigned to works.
- **Publishers:** Companies and organizations that distribute works.
- **Funders:** Organizations that fund research.
- **Geo:** Where things are in the world.

Limitations and Access: The API is the primary way to get OpenAlex data. It's free and requires no authentication. The daily limit for API calls is 100,000 requests per user per day. There is also a complete database snapshot available to download. If there is need for a higher API limit, or more frequent updates, it is advisable to look into OpenAlex Premium. There is also a web interface called OpenAlex Web which will be available in July 2023.

Justification: OpenAlex is central to the project as it forms the back-end of GAPRS. It is the ever-growing database from which GAPRS will source its large pool of recommendations.

3.2 Datasets

Due to the lack of available datasets in the RPRS space, it was necessary to create a dataset to act as a ground truth for which offline evaluation metrics could be calculated and performance compared across different recommender systems. This section describes the created dataset to be used in the "Experiments and Results" chapter 5.

3.2.1 User Queries

The "User Queries" dataset consists of eleven handcrafted user queries. Each user query is centered around either a Bachelor's thesis or a Master's thesis and contains a number of elements which provide context or options for the actual user query. If available, each example in the dataset consists of a thesis topic, author, keywords, research questions or a research problem. This context is there to help the user craft their actual user query string which will be entered into the recommender system. If the thesis topic produces no results then the author is used, if author produces no results then the keywords are used, and so on until all context or options are exhausted. However, in most cases, the thesis topic or research questions should be available to the user and should suffice. To see the dataset in full, please refer to the GitHub link containing the PDF.

3.3 Lists of Metrics and Algorithms

In this section, the different lists of data points which are relevant to the project are compiled and justified.

3.3.1 List of Recommender System Evaluation Metrics

To evaluate the performance of GAPRS, we will need to make reference to and employ a range of different kinds of evaluation metrics for RS to obtain a multi-faceted picture of its capabilities. These evaluation metrics will be made reference to in the "Methodology" chapter 4 and used in the "Experiments and Results" chapter 5. Below, is a list of the evaluation metrics in consideration for this project whose definitions have been paraphrased from these papers [1, 2, 6].

1. **User Studies:** User studies are used to assess the user behavior and experience towards the developed system. A number of participants are asked to take part in user study and perform a set of actions during their interaction with the system. This typically includes question answering about their choice, experience etc. While interacting with the system, user behavior is observed subjected to statistical methods for identifying latent characteristics. By using some quantitative measures, knowledge about hidden features which are difficult to observe, can be drawn. Based on the inferred user information, personalized recommendations are generated. Further, various explicit and implicit methods are employed to validate the correctness of algorithm. Questionnaires are frequently used in user studies to collect user opinion, expressed in terms of reviews, ratings, etc. Following the recommendations, a user is asked to answer various questions about the recommended articles. Questionnaire should be designed in such a way that it considers user opinion on different aspects such as accuracy, novelty, diversity, usefulness, serendipity etc. Implicit methods such as click-through rate may also be used to measure relevance, usefulness, satisfaction and so on. User studies are of two types: lab and real-world user studies.

   (a) **Lab:** In lab studies, users are well informed of the experiments conducted. Participants have prior understanding of the system and know that they are taking part in a user study. They are aware that their interaction with the system and actions performed are recorded, which knowingly or unknowingly affect their response. The user behavior observed may not be genuine or include some biases, therefore the evaluation results are not as expected.

(b) **Real-World:** In real-world studies, real users participate in the evaluation process. They do not have any prior knowledge about the system and a genuine interaction takes place during the user session. They are unaware of the experiments conducted and explore the system features for their own benefit. In these studies, users provide what they actually feel about the system in terms of ratings, reviews, tags and so on. Their feedback helps in fine-tuning the system for real users and improves the efficiency of recommendation system. With true user opinion, the system generates useful recommendations by adapting to the new circumstances and user's current requirements.

2. **Online Evaluation:** Online evaluations, also known as real-life testing, were first used by the online advertising and e-commerce fields. They measure the acceptance rates of recommendations in real-world recommender systems. Acceptance rates are typically measured by click-through rates (CTR), i.e., the ratio of clicked recommendations to displayed recommendations. Online evaluation is the most reliable measure which is used to evaluate many real world systems, it tests all the functional and non-functional aspects of a system. It evaluates a recommender system under real-world situations covering real facets such as user satisfaction, robustness, scalability, latency, performance in terms of stress and load testing.

3. **Offline Evaluation:** Offline evaluations typically measure the accuracy of a recommender system based on a ground truth dataset. This means that approaches that are effective in offline evaluations are not necessarily effective in real-world recommender systems.

3.3.2 List of Network Science Metrics

Assessing the effectiveness of the graph/network on which GAPRS is based will require comparing the graph/network against a baseline which will be defined in a later chapter. These network science concepts have been taken from Mark Newman's book on networks [7].

1. **Order:** Number of nodes in network.

2. **Size:** Number of edges in network.

3. **Co-Citation Count (CCC):** This is the number of academic papers which cite academic papers A and B together in their list of references.

4. **Bibliographic Coupling Count (BCC):** This is the number of academic papers that papers A and B cited in common in their respective lists of references.

5. **Jaccard Coefficient:** This is the cardinality of set intersection over the cardinality of set union.

6. **Normalized Co-Citation Count (NCCC):** This is the jaccard coefficent applied to the CCC. The number of academic papers that co-cite A and B together divided by the number of academic papers that only cite A or B without those co-citing both.

7. **Normalized Bibliographic Coupling Count (NBCC):** This is the jaccard coefficient applied to BCC. The number of academic papers that A and B cite in common in their list of references divided by the number of academic papers that only A cites in references plus the number of papers only B cites in its references without the papers cited by A and B in common.

3.3.3 List of Recommendation Algorithms

Comparing the algorithm eventually used by GAPRS against SOTA recommendation algorithms is a must to obtain a relative understanding of GAPRS' back-end throughput and computational complexity. Below, is a list of recommendation algorithms, categorised by recommendation approach, which will be implemented and compared against GAPRS. Explanation of the comparison procedure will be provided in the next chapter 4 and the results for each comparison will be shown in the "Experiments" chapter 5. Please note that the definitions in the list below have been taken from Joeran Beel et al. [1].

1. **Stereotyping:** Stereotyping is one of the earliest user modeling and recommendation classes. It was introduced by Rich in the recommender system Grundy, which recommended novels to its users. Rich was inspired by stereotypes from psychology that allowed psychologists to quickly judge people based on a few characteristics. Rich defined stereotypes, which she called "facets", as collections of characteristics. For instance, Grundy assumed that male users have "a fairly high tolerance for violence and suffering, as well as a preference for thrill, suspense, fast plots, and a negative interest in romance. . Consequently, Grundy recommended books that had been manually classified to match the facets.

2. **Content-based Filtering:** Content-based filtering (CBF) is one of the

most widely used and researched recommendation class. One central component of CBF is the user modeling process, in which the interests of users are inferred from the items that users interacted with. *Items* are usually textual, for instance emails or webpages. *Interaction* is typically established through actions, such as downloading, buying, authoring, or tagging an item. Items are represented by a content model containing the items' features. Features are typically word-based, i.e., single words, phrases, or n-grams. Some recommender systems also use non-textual features, such as writing style, layout information and XML tags. Typically, only the most descriptive features are used to model an item and users and these features are commonly weighted. Once the most discriminative features are identified, they are stored, often as a vector that contains the features and their weights. The user model typically consists of the features of a user's items. To generate recommendations, the user model and recommendation candidates are compared, for example using the vector space model and the cosine similarity coefficient. In the RPRS community, CBF is the predominant recommendation class.

3. **Collaborative Filtering:** The term "collaborative filtering" (CF) was coined in 1992 by Goldberg et al., who proposed that "information filtering can be more effective when humans are involved in the filtering process". The concept of collaborative filtering as it is understood today was introduced 2 years later by Resnick et al. Their theory was that users like what like-minded users like, where two users were considered like-minded when they rated items alike. When like-minded users were identified, items that one user rated positively were recommended to the other user, and vice versa. **Compared to CBF, CF offers three advantages**. First, CF is content independent, i.e., no error-prone item processing is required. Sec-

ond, because humans do the ratings, CF takes into account real quality assessments. Finally, CF is supposed to provide serendipitous recommendations because recommendations are not based on item similarity but on user similarity.

4. **Co-Occurrence:** To give co-occurrence recommendations, those items are recommended that frequently co-occur with some source items. One of the first applications of co-occurrence was co-citation analysis introduced by Small. Small proposed that two papers are more related to each other, the more often they are co-cited. Many others adopted this concept, the most popular example being Amazon's "Customers Who Bought This Item Also Bought." Amazon analyzes which items are frequently bought together, and when a customer browses a product, items frequently bought with that item are recommended.

5. **Graph-based:** This approach combines a variety of different graph concepts such as utilising graph metrics to find recommendation candidates when enough source data is gathered. Typically, one or several input papers are given and from this input random walks with restarts are conducted to find the most popular items in the graph.

6. **Global Relevance:** In its simplest form, a recommender system adopts a one-fits-all approach and recommends items that have the highest global relevance. In this case, the relevance is not calculated specific to a user. Instead, some global measures are used, such as overall popularity. For instance, a movie-rental system could recommend those movies that were most often rented or that had the highest average rating over all users. In this case, the basic assumption would be that users like what most other users like.

7. **Hybrid:** Approaches of the previously introduced recommendation classes may be combined in hybrid approaches that are either weak or true. In the former, one of the combined approaches is still dominant while in the latter both are similarly important.

3.4 Direct Competitors

There are a handful of organisations working on RPRSs that are directly competing with GAPRS. These organisations are mainly small start-ups targeting the same niche and using a similar technique. They aggregate their data from multiple open academic repositories and databases then model the collected academic papers in a citation network and calculate the most highly-cited and recent academic papers to finally recommend these to the user. This section presents a non-exhaustive list of organisations directly competing with GAPRS. They are largely unknown and not as popular as the indirect competitors listed in the next section.

3.4.1 List of Research Paper Recommender Systems

To be convinced of this project's performance, we must compare GAPRS against its ilk. The list below contains the direct competitors currently known to the author.

- LitMaps
- ResearchRabbit
- Inciteful
- Connected Papers

- SciSpace
- Elicit.org
- Iris.ai
- System Pro

3.5 Indirect Competitors

There exists a larger number of indirect competitors with GAPRS which are being run by non-profit organisations for the most part. In this section, a non-exhaustive list of organisations indirectly competing with GAPRS is presented.

3.5.1 List of Academic Databases and Search Engines

To be convinced of this project's performance, we must compare GAPRS against the most commonly used methods for finding academic papers. For more entries, please reference Wikipedia's article on the subject [8] as only the most commonly used software are being considered in this project for comparison.

- Academia.edu
- Arxiv
- Biorxiv
- Techrxiv
- Arxiv Sanity
- Papers with Code
- JSTOR

- Research Gate
- Elsevier (Scopus)
- ScienceOpen
- Google Scholar
- eLife
- PubMed
- Microsoft Academic
- OpenAlex
- OurResearch

Chapter 4

Methodology

This chapter explains how GAPRS makes recommendations using a graph-based approach as represented by the "G" in the acronym. The first subsection describes the graph/network algorithm including its inputs, outputs and hyperparameters. The second subsection covers the steps taken by the algorithm from its input to output and everything in-between. The third subsection describes the type of networks created by the graph/network algorithm to allow the reader to interpret the "Experiments and Results" in the following chapter 5.

4.1 Graph-based Recommendation Algorithm

This section describes the algorithm used by GAPRS and explains how it works step-by-step.

4.1.1 Description of Algorithm

The graph algorithm used by GAPRS to make recommendations can be described completely in terms of its inputs, outputs and hyperparameters.

4.1.1.1 Algorithm Inputs

Below are the user inputs required by the algorithm organised into a numbered list.

1. **User Query String (String):** The first input required from the user by GAPRS is a user query string. This user query string is then used to query the OpenAlex API for a ranked list of academic papers sorted in descending order of relevancy score; a metric created by OpenAlex that consists of text similarity to the user query string and citation count.

2. **Rank (Integer):** Secondly, the algorithm requires the user to select the ranks of the recommendations they wish to select from the initial list of academic papers presented to the user.

3. **Choice (Character):** Thirdly, the algorithm asks the user to enter the character "Y" or "N" to indicate whether or not they would like to see another iteration of the algorithm. The program terminates when the user enters the character "N".

4.1.1.2 Algorithm Outputs

Below are the outputs produced by the algorithm organised into a numbered list.

1. **List of Initial Recommendations:** The algorithm takes the first input and uses it to query the OpenAlex API for a ranked list of academic papers sorted in descending order of relevancy score. This list is then printed to the console for the user to see.

2. **List of Selected Recommendations with References:** The second output produced by the algorithm is a list of the recommendations selected

by the user from the initial list of recommendations. This new list is printed to the console for the user to easily see their selections.

3. **Images of Network:** On each iteration of the algorithm an image of the network is produced for the user visually interpret the recommendations. The red nodes are recommendations while the blue nodes are references. The images also provide some sense of trustworthiness as the user can see how the algorithm is working from one image to another.

4. **Text File of Final Image of Network in Edge List Graph Format:** The algorithm outputs a text file containing an edge list representation of the network in the final image where each line in the file is a comma-separated triple consisting of the source node which is a space-separated first name, last name and publication year; the destination node which is the same format as the source node; finally, the edge weight which is a numeric value representing the strength or weakness of the citation relationship between the source and destination nodes.

5. **List of Final Recommendations:** Finally, the algorithm outputs a list of the final recommendations after expanding select nodes in the network. This list is then printed to the console for the user to see all recommendations. Both the initial recommendations selected by the user and the additional recommendations calculated by the algorithm.

4.1.1.3 Algorithm Hyperparameters

The associated hyperparameters of the algorithm are listed and defined in this subsection.

Number of Selected Recommendations: This is the number of recommendations selected by the user from the initial list of recommendations and it is represented by the variable, K.

Number of Times User Asks to see another Iteration: The user can ask to see another iteration an indefinite number of times. This number will be multiplied by two when used as it represents a double pass where each pass counts as one iteration and is represented by the variable, M.

Total Number of Iterations: This is the formula for the total number of iterations of the algorithm that the user will see and is represented by the variable, N.

$$N = 1 + 2M + 1$$

- **First Iteration:** The first iteration always occurs so that is where the first 1 comes from in the above equation. The first iteration adds K red nodes to the total.
- **Intermediate Iterations:** The next $2M$ iterations are optional and will always be even. Every second iteration of these $2M$ iterations adds K red nodes to the total.
- **Final Iteration:** The final iteration always occurs so that is where the second 1 comes from. The final iteration does not add any red nodes to the total.
- **Parity:** N is always even so there is no need to worry about fractions on division by two.

Iteration Index: This is the index tracking which iteration the algorithm is on. The index is represented by the variable, i, and it falls in the range $1 \leq i \leq N$.

Total Number of Red Nodes in Final Image of Network: This is the total number of red nodes or egos or recommendations which will be presented to the user and is represented by the following formula

$$(N/2) * K$$

.

4.1.2 Steps of Algorithm

The graph algorithm used by GAPRS to make recommendations will be explained logically step-by-step in this subsection.

1. **Querying the OpenAlex API and Generating Lists of Recommendations:** The first step of the algorithm involves asking the user to enter their query string and then using this user query string to query the OpenAlex API for it to generate a ranked list of academic papers sorted by decreasing order of relevancy score and finally display these lists to the user. This process only happens once but lists are displayed twice. Once when only the initial list of recommendations is shown in the console and again when the user's selection of recommendations from the initial list is shown in the console.

2. **Assembling the Hybrid Weighted Undirected 1-Degree Multi-Egocentric Citation Network:** The next step involves building the base citation net-

work that will be used to create sub-networks for later calculations.

(a) **User's Selected Recommendations become Egos:** All of the academic papers selected by the user from the initial list of recommendations become egos of the network.

(b) **References of Selected Recommendations become Alters:** Each academic paper selected from the initial list of recommendations is an ego of the network. Each of the references in the bibliography associated with each of these selected academic papers become alters of their respective egos.

(c) **Edge Weights Calculated between Alters and their Respective Egos:** The edge weights are calculated between the alters and their respective egos. The edge weight is given by the following formula $\frac{NCCC+NBCC}{2}$ or written equivalently as seen in the images, $avg(NCCC, NBCC)$.

(d) **Save Image of Network:** An image of the network is saved as a PNG file.

3. **Polling User for Input:** Keep asking the user whether they want to see another iteration of the algorithm. This step is referring to the optional $2M$ iterations. The first pass consists of steps a-c and the second pass consists of steps d-f.

(a) **Create Hybrid Weighted Undirected 1.5-Degree Egocentric Citation Sub-Networks excluding Egos:** A 1.5-degree egocentric citation sub-network with the ego excluded is created for each ego.

(b) **Combine Sub-Networks:** Each of the sub-networks created in the previous step are combined into one large network.

(c) **Save Image of Network:** An image of the network is saved as a PNG file.

(d) **Expand Alter Nodes with Highest Centrality Measure:** For each sub-network in the large network, the alter node with highest degree taking edge weight into account is expanded. Meaning that it is turned from an alter into an ego with its own alters.

(e) **Add New Egos to Original Network:** These new egos are then added to the original network which adds to the total count of egos.

(f) **Save Image of Network:** An image of the network is saved as a PNG file.

4. **Save Text File of Final Image of Network with Egos Only:** A text file representing the network in the final image with egos only which are denoted by red nodes is saved in edge list graph format.

5. **Save Final Image of Network with Egos Only:** An image of the network with egos only which are denoted by red nodes is saved as a PNG file.

4.1.3 Description of Networks

This subsection explains the terminology behind the two types of networks created by the algorithm. The first type of network created by the algorithm is a *Hybrid Weighted Undirected 1-Degree Multi-Egocentric Citation Network* and the second type of network created by the algorithm is a *Hybrid Weighted Undirected 1.5-Degree Egocentric Citation Sub-Network excluding Ego*. Each of the terms present in both networks are defined below to build familiarity and better understanding.

Hybrid: The term "hybrid" is used to emphasise the fact that these citation networks are composed of a combination of CCC and BCC, namely $avg(NCCC, NBCC)$. Usually citation networks use one or the other not both and in that case it would not be hybrid.

Weighted: This is to say that the graph/network has numeric values called edge weights associated with each of the edges.

Undirected: The edges between nodes are directionless or mutual.

1-Degree (or 1.5 Degree) Multi-Egocentric Network: This definition can be broken down as follows.

- **Ego:** Centre of an egocentric network.
- **Alter:** Immediate neighbour of ego node.
- **Multi-Egocentric:** There are multiple egos in the network or multiple egocentric networks.
- **1-Degree Egocentric Network:** An egocentric network with edges between ego and its alters only.
- **1.5 Degree Egocentric Network:** An egocentric network with edges between ego and its alters as well as edges between alters with each other.
- **Sub-Network:** Subset of nodes and edges within a network.
- **Citation Network:** Network where nodes are academic papers and edges represent citation relationship between academic papers A and B.

Chapter 5

Experiments and Results

This chapter aims to specify, enumerate and set the experiments deemed relevant to the project in such a way that the experimental results are reproducible.

5.1 Baselines

These experiments are **baselines** against which other experiments will be compared to see if they are up to standard and do better than simple approaches for recommendation without the use of any graph features.

5.1.1 Data Retrieval Recommender

This experiment is a **baseline** against which other experiments will be compared to see if they are up to standard and do better than basic data retrieval in a recommender system without the use of any graph features.

1. **Independent Variable:** User Information Need, User Query String
2. **Dependent Variable:** Number of Academic Papers Retrieved
3. **Control Variable:** Recommendation Approach

4. **Hyperparameter Tuning:** Not Applicable
5. **Evaluation Metrics:** Novelty defined as number of new recommendations generated by the algorithm excluding the ones selected by the user.

5.2 Experimental Setup

In this section, the hyperparameters are set and the experimental details to be referenced elsewhere are stated clearly for consistency.

5.2.1 Hyperparameter Settings

These are the hyperparameters settings which are fixed for each experiment to establish a control for comparison: $K = 3$, $M = 2$, $N = 6$, $1 \leq i \leq 6$, $(N/2) * K \leq 9$.

5.2.2 Experimental Details Excluded

Visualisations of the results in the form of images for each experiment and the full context surrounding each user query will be made available at the GitHub repository.

5.3 Experimental Results

This section presents the results of each experiment in numbered list format.

1. **User Query:** Graph-based Academic Paper Recommender System | **Novelty:** 6
2. **User Query:** Minimum Sudoku Clue Problem | **Novelty:** 6

3. **User Query:** Neuroevolution Approach to Robotic Arm Control | **Novelty:** 5

4. **User Query:** Anomaly Detection using Internet of Things Sensors | **Novelty:** 6

5. **User Query:** Dynamic Economic Emissions Dispatch with Thresholded Lexicographic Ordering | **Novelty:** 6

Chapter 6

Summary and Conclusions

In summary, this masters thesis proved to be a very practical opportunity to dive deeply into the topics of my choice, namely, RSs and RPRSs more specifically. It was an enlightening experience with major moments where there was need to exhibit decisiveness, professionalism, ingenuity, novelty, perseverance, communication, organisation, initiative, drive, swiftness, learning ability, working with ambiguity, patience and understanding. To dedicate a concentrated portion of the year towards something you can call your own and be able to grow from all your acquired knowledge through struggling with concepts and finding one's way in the dark is what the masters thesis is all about. There is a feeling of getting the full masters experience from start to end that brings a sense of fulfillment and relief.

References

[1] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Research paper recommender systems: A literature survey," in *International Journal on Digital Libraries*. Springer International Publishing, 2016. 10, 11, 12, 16, 17, 18, 19, 20, 23, 24, 30, 37, 40

[2] R. Sharma, D. Gopalani, and Y. Meena, "An anatomization of research paper recommender systems: Overview, approaches and challenges," *Engineering Applications of Artificial Intelligence*, vol. 118, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0952197622006315 13, 14, 15, 37

[3] C. K. Kreutz and R. Schenkel, "Scientific paper recommendation systems: A literature review of recent publications," *Computing Research Repository (CoRR)*, 2022. [Online]. Available: https://arxiv.org/abs/2201.00682 20, 27

[4] S. Wang, L. Hu, Y. Wang, X. He, Q. Z. Sheng, M. A. Orgun, L. Cao, F. Ricci, and P. S. Yu, "Graph learning based recommender systems: A review," *Computing Research Repository (CoRR)*, 2021. [Online]. Available: https://arxiv.org/abs/2105.06339 29

[5] J. Priem, H. Piwowar, and R. Orr, "Openalex: A fully-open index of scholarly works, authors, venues, institutions and concepts," *International*

Conference on Science and Technology Indicators, 2022. [Online]. Available: https://arxiv.org/abs/2205.01833 34

[6] C. Wang, "Evaluating recommender systems," Bachelor's Thesis, University College Dublin, Dublin, Republic of Ireland, October 2018. [Online]. Available: https://wangcongcong123.github.io/files/bechelor-thesis/report.pdf 37

[7] M. Newman, *Networks (2nd Edition).* Oxford University Press, 2018. 39

[8] Wikipedia, "List of academic databases and search engines," accessed on 21-June-2023. [Online]. Available: https://en.wikipedia.org/wiki/List_of_academic_databases_and_search_engines 44