INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

A Thesis on

A Data-Centric Lens for Representation Learning in Graphs via Semantic and Structural Enhancement

submitted
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

by
Karan Goyal
PhD21007

Advisors: Prof. Vikram Goyal, Prof. Mukesh Mohania

Department of Computer Science and Engineering
Indraprastha Institute of Information Technology - Delhi
New Delhi-110020 (India)

July 17, 2026

Certificate

This is to certify that the thesis titled " **A Data-Centric Lens for Representation Learning in Graphs via Semantic and Structural Enhancement** " being submitted by **Karan Goyal** to the Indraprastha Institute of Information Technology Delhi, for the award of the degree of **Doctor of Philosophy**, is an original research work carried out by him under our supervision. In our opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

17th July, 2026

Dr. Vikram Goyal
Thesis Supervisor
Professor, Dept. of CSE

Dr. Mukesh Mohania
Thesis Supervisor
Professor, Dept. of CSE

Indraprastha Institute of Information Technology Delhi
New Delhi 110020

Declaration

"I acknowledge that I am fully responsible for the entire content of my thesis, including any sections assisted by online tools, including Artificial Intelligence-based tools. I accept full accountability for any violations of ethical standards in publications arising from the use of such tools."

July 17, 2026

Karan Goyal, PhD21007

Dept. of CSE

Advised by Prof. Vikram Goyal and Prof. Mukesh Mohania

Indraprastha Institute of Information Technology Delhi
New Delhi 110020

Acknowledgements

I fall extremely short of words to express my utmost gratitude towards my advisors, Prof. Mukesh Mohania and Prof. Vikram Goyal. This journey was made immeasurably better by the camaraderie of Yash, Aseem, Shivani and Sarah, who were always there. Had I acted sooner upon their collective wisdom, this journey would have been completed atleast an year earlier. But then, life is the best teacher. I am also deeply grateful to Pankaj and Akshit for their immense help when I needed it the most. I thank Pragya, Shyam, Aradhya, Sourav, Shivam and Sanyam for being helpful in my journey. I sincerely thank Bhawani for his IT help. Lastly, I again deeply thank Sarah for voluntarily reviewing my work and providing really good suggestions.

Karan Goyal

Abstract

While progress in graph representation learning has often been model-centric: focusing on the design of more powerful architectures, this thesis presents a data-centric lens: a framework viewing the graph not as a static input, but as an object to be actively and intelligently enhanced. We argue and then show that superior performance can be unlocked through a deliberate progression, beginning with specialised node-level semantic enhancement and culminating in generic graph-level structural enhancement.

Our investigation begins with the specialised task of citation recommendation in scholarly networks, where we first explore semantic enhancement. Our formative framework, **SymTax**, enriches node data by modelling human exploratory behaviour via symbiotic relationship and by fusing taxonomies into hyperbolic space. Building upon this, we address critical issues of scalability and real-world applicability by introducing Profiler, a highly efficient module that creates a rich **Public Profile** for each paper by statically aggregating signals from its inward citations, which is then carefully integrated with our designed DAVINCI reranker. This advancement is validated under our proposed Inductive evaluation protocol, establishing a more rigorous, robust and realistic standard for the field.

Having established the power of semantic enrichment, we then widen the horizon of our data-centric lens to structural enhancement for the generalised challenge of learning on heterophilic graphs. We propose **PathLens**, a model-agnostic technique that fundamentally augments the graph's topology. By strategically adding supernodes derived from spectral clustering, PathLens creates new information pathways, effectively reducing the average shortest path length between similar nodes thereby increasing the graph's homophily. This directly addresses the core impediment of heterophily, significantly boosting the performance of a wide range of standard GNNs as well as heterophilic GNNs by providing them with a more learnable graph structure.

Collectively, this body of work validates the efficacy of the data-centric lens. By first enriching the semantic content of nodes and then augmenting the structural fabric of the graph, we demonstrate a powerful and generalisable paradigm for building high-performance systems for learning on graphs. This thesis contributes not only a suite of novel techniques but also a guiding philosophy that prioritises the enhancement of the data itself as a primary driver of success for representation learning in graphs.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The research documented within this thesis follows a principled trajectory, moving from a specific, application-driven challenge to a general, methodologically-focused solution for learning on complex networks. This journey is built upon a central argument: that significant advancements in graph machine learning can be unlocked through a deliberate, data-centric progression, beginning with the enrichment of data at the node level and culminating in the augmentation of the graph's fundamental structure. We motivate and develop this progression through two distinct but connected domains: first, by tackling the intricate problem of citation recommendation in scholarly networks, and second, by generalising our data-centric philosophy to enhance the performance of Graph Neural Networks on the broad and challenging class of heterophilic graphs. This introductory chapter serves to establish the context for this investigation, articulate the details of our core research narrative, and provide a clear roadmap for the specific contributions and analyses presented in the chapters that follow.

1.1 The Ubiquity and Challenge of Graph-Structured Data

From the intricate web of scientific citations to the vast expanse of social networks, graph structures are a fundamental data modality for representing complex relationships in the real world. They provide a powerful toolkit for learning from this relational data, enabling groundbreaking advances in tasks ranging from recommendation to classification. However, the raw performance of several learning models is often constrained by the inherent properties of the input graph. Real-world networks are rarely simple or clean; they are complex, noisy, and often possess challenging structural characteristics that can impede learning. This thesis is founded on the premise that to unlock the next level of performance in graph machine learning, we must move beyond model-centric innovations and adopt a data-centric perspective, asking: How can we strategically enrich and augment the graph data itself to make it more amenable to effective representation learning?

1.2 From Data Enrichment to Structural Augmentation

This thesis charts a deliberate research journey that explores this question through a progressively generalisable framework. Our central argument is that sophisticated, domain-aware data enhancement is a key driver of success in complex graph learning tasks. We demonstrate this through a clear, two-part trajectory:

- Node-level data enrichment in a specialised domain: We begin our investigation within the specialised, high-impact context of citation recommendation in scholarly networks. Here, we develop techniques that enrich the information available at the node level. We explore two complementary forms of enrichment: first, by modelling latent human behaviours and fusing external hierarchical knowledge (SymTax), and second, by developing a highly scalable method to fuse semantic signals from a node's local network neighbourhood (Public Profile Matters).

- Graph-level structural augmentation in a general domain: Having established the power of data enrichment, we then abstract this principle to a more fundamental level. We shift our focus from enriching the features of existing nodes to augmenting the very structure of the graph. We introduce a novel technique, PathLens, that operates on generic graphs, particularly those exhibiting challenging heterophily. PathLens creates new information pathways, fundamentally altering the graph's topology to facilitate more effective information flow for downstream GNNs.

This progression from specialised node enrichment to general structural augmentation forms the narrative backbone of this thesis, demonstrating a cohesive and deepening exploration of data-centric AI for graphs.

1.3 Contributions

The core contributions of this thesis are organised along this trajectory:

- Formative techniques for data enrichment (SymTax): We introduce a novel citation recommendation framework that enriches the learning process by i) modelling the human behaviour of citation discovery through a Symbiosis mechanism that leverages a paper's outgoing neighbourhood, and ii) enriching node features by fusing them with hierarchical Taxonomic knowledge learned in hyperbolic space.

- Scalable and realistic framework for data enrichment (Public Profile Matters): We address the critical limitations of prior work by i) proposing a rigorous inductive evaluation protocol that better simulates real-world deployment, and ii) developing the Profiler module, a scalable, non-learnable method for enriching a paper's representation by fusing semantic signals from its inward citation neighbourhood. This work establishes a new state-of-the-art in both performance and practical applicability.

- Generalisable framework for structural augmentation (PathLens): We transcend node-level enrichment to introduce a graph-level augmentation technique. PathLens is a structure-driven method that improves GNN performance on heterophilic

graphs by strategically adding a new layer of supernodes and pathways, enabling better information exchange between both local and distant regions of the graph.

1.4 Thesis Roadmap

This thesis is structured as follows. Chapter 2 provides the necessary background on citation recommendation, graph neural networks, and data-centric approaches. Part I via Chapter 3 then provides a prologue to our work on data enrichment for citation recommendation, with Chapter 4 detailing our formative SymTax framework in a transductive setting, and Chapter 5 presenting our improved, scalable, and inductive framework. Part II via Chapter 6 generalises our focus, with Chapter 7 detailing our PathLens design for structural augmentation of heterophilic graphs. Finally, Chapter 8 concludes by summarising our journey, pointing out the limitations of our work, and outlining avenues for future research. At last, Chapter 9 presents the research output and coursework conducted throughout this PhD journey.

Chapter 2

Background

The contributions of this thesis are situated at the intersection of a specific, high-impact application domain and a general, fundamental machine learning methodology. To provide a clear context for our work, this chapter is organised into two primary sections. We first provide a overview of literature on Citation Recommendation in Scholarly Networks, establishing this as the complex, real-world testbed where the initial challenges of data enrichment were confronted. We then broaden our lens to the underlying principles of Representation Learning on Graphs, focusing on the evolution of Graph Neural Networks and the critical difficulties they encounter in heterophilic settings. This dual-focus review provides the necessary technical and conceptual foundation, clarifying the state-of-the-art from which our novel solutions depart.

2.1 Citation Recommendation in Scholarly Networks

The exponential growth of scientific literature has made the manual discovery of relevant prior work a significant bottleneck in the research lifecycle. To mitigate this, the field has developed citation recommendation systems, which can be broadly categorised into two methodologies: global and local recommendation. Global recommendation aims to suggest a list of relevant papers for an entire document, typically based on its title and abstract. In contrast, this thesis focuses on the more fine-grained and practical task of local citation recommendation. As defined in foundational works like He et al. (2010), the objective of local recommendation is to suggest the most relevant citation for a specific textual segment under consideration within a document.

Early investigations into this task ranged from vector similarity methods based on TF-IDF to more advanced neural approaches, such as the neural probabilistic model of Huang et al. (2015) and the encoder-decoder architectures of Ebesu and Fang (2017). A significant advancement came with the integration of graph-based signals. The BERT-GCN model (Jeong et al., 2020), for instance, combined powerful contextualised embeddings from BERT with structural information from a Graph Convolutional Network (GCN). However, as noted by Gu, Gao, and Hahnloser (2022), the computational intensity of GCNs created a critical scalability bottleneck, constraining evaluation to small datasets.

This scalability challenge motivated the development of the now-dominant two-stage recommendation architecture, which decouples the task into a rapid retrieval (or "prefetching") stage and a more computationally intensive, precise reranking stage. This

paradigm, notably established by the HAtten model (Gu, Gao, and Hahnloser, 2022), optimises for both speed and accuracy and serves as a primary architectural baseline for the work in this thesis. The work in this thesis in Chapter 4, particularly SymTax (Goyal et al., 2024), further evolved this into a three-stage architecture by introducing an intermediate "Enricher" stage, demonstrating the continued innovation within this retrieval-and-reranking framework.

A central contribution of this thesis is addressing a fundamental, yet often overlooked, limitation in the standard evaluation protocol for citation recommendation. Traditionally, models are evaluated in a transductive setting. In this setup, the corpus of candidate documents is constructed from the union of the training, validation, and test sets. While this does not lead to direct label leakage, it creates an artificial evaluation landscape where the ground-truth citation for a test query may itself be another document within the test set. This means the system is evaluated on its ability to find connections within a static collection where query documents are pre-indexed and searchable – a condition that never holds in a real-world application.

To faithfully address this shortcoming, our work in "Public Profile Matters" proposes and adopts a rigorous inductive evaluation setting. The core principle of this setting is to enforce a strict temporal separation between the evaluation query and the candidate corpus. Formally, for any query document, the candidate corpus must only contain documents published strictly before it. This setup ensures that, at evaluation time, a model is tasked with recommending citations for a newly authored paper using only the body of existing literature. This protocol eliminates any artificial advantage and provides a more realistic and reliable assessment of a model's true generalisation capabilities, a standard adopted for the advanced work presented in Chapter 5.

2.2 Representation Learning with Graph Neural Networks

Graph Neural Networks have become the premier tool for learning tasks like node classification on graph-structured data. The dominant paradigm for GNNs is message passing (Gilmer et al., 2017), where a node's feature representation is iteratively updated by aggregating features from its local neighbourhood. Foundational models like Graph Convolutional Networks (GCN) (Kipf & Welling, 2017), GraphSAGE (Hamilton et al., 2017), and Graph Attention Networks (GAT) (Vel et al., 2018) each propose different mechanisms for this aggregation and update process.

The success of these standard GNNs is largely predicated on the homophily assumption, or the principle that connected nodes in a graph tend to be similar (e.g., belong to the same class). However, many real-world graphs exhibit a low degree of homophily, a property known as heterophily, where connections between dissimilar nodes are frequent. In this setting, the message-passing mechanism becomes problematic, as aggregating features from dissimilar neighbours can corrupt a node's representation and degrade performance. This challenge has given rise to two major research directions aimed at improving GNNs in heterophilic settings.

One significant line of research focuses on designing novel and more complex GNN architectures specifically for heterophilic graphs. These models often modify the core

message-passing scheme to better distinguish between similar and dissimilar neighbours. While often effective, these model-centric approaches typically increase architectural complexity. The work in this thesis pursues an alternative and complementary direction. Instead of designing new models, the data-centric paradigm seeks to improve performance by modifying the input data or graph structure itself to make it more amenable to learning by standard GNNs. This aligns directly with the central theme of this thesis. Prior work in this area has explored several strategies for graph augmentation or rewiring. These methods confirm the viability of improving performance through structural modification. The PathLens framework, presented in Chapter 7, contributes a novel and distinct approach to this paradigm creating new global information pathways and establishing state-of-the-art in heterophilic graph representation learning.

Chapter 3

Prologue to Part I: Citation Recommendation via Data Enrichment in Scholarly Networks

This first part of the thesis focuses on the specialised domain of citation recommendation, presenting a two-chapter arc that demonstrates a clear progression in design choices, scalability, and real-world applicability. Chapter 4, based on our first article, presents our initial exploration into enhancing citation recommendation by enriching the data at the node level. This work establishes a strong performance baseline and, critically, its inherent limitations in scalability and evaluation realism serve as the primary motivation for the more advanced system developed in the subsequent chapter.

Building directly on the insights and challenges identified in Chapter 4, our second article documented in Chapter 5 introduces a more robust, scalable, and practically-grounded system for citation recommendation establishing a new state-of-the-art on a more realistic and challenging task formulation. We provide the details of the articles in their published form as follows.

3.1 Details of Article 1

SymTax: Symbiotic Relationship and Taxonomy Fusion for Effective Citation Recommendation Karan Goyal, Mayank Goel, Vikram Goyal, and Mukesh Mohania. (Accepted at ACL 2024 conference)

Citing pertinent literature is pivotal to writing and reviewing a scientific document. Existing techniques mainly focus on the local context or the global context for recommending citations but fail to consider the actual human citation behaviour. We propose `SymTax`, a three-stage recommendation architecture that considers both the local and the global context, and additionally the taxonomical representations of query-candidate tuples and the *Symbiosis* prevailing amongst them. `SymTax` learns to embed the infused taxonomies in the hyperbolic space and uses hyperbolic separation as a latent feature to compute query-candidate similarity. We build a novel and large dataset `ArSyTa` containing 8.27 million citation contexts and describe the creation process in detail. We conduct extensive experiments and ablation studies to demonstrate the effectiveness and design choice of each module in our framework. Also, combinatorial analysis from our

experiments shed light on the choice of language models (LMs) and fusion embedding, and the inclusion of section heading as a signal. Our proposed module that captures the symbiotic relationship solely leads to performance gains of 26.66% and 39.25% in Recall@5 w.r.t. SOTA on `ACL-200` and `RefSeer` datasets, respectively. The complete framework yields a gain of 22.56% in Recall@5 wrt SOTA on our proposed dataset. The code and dataset are available at https://github.com/goyalkaraniit/SymTax.

3.2 Details of Article 2

Public Profile Matters: A Scalable Integrated Approach to Recommend Citations in the Wild Karan Goyal, Dikshant Kukreja, Vikram Goyal, and Mukesh Mohania. (Under review)

Proper citation of relevant literature is essential for contextualising and validating scientific contributions. While current citation recommendation systems leverage local and global textual information, they often overlook the nuances of the human citation behaviour. Recent methods that incorporate such patterns improve performance but incur high computational costs and introduce systematic biases into downstream rerankers. To address this, we propose *Profiler*, a lightweight, non-learnable module that captures human citation patterns efficiently and without bias, significantly enhancing candidate retrieval. Furthermore, we identify a critical limitation in current evaluation protocol: the systems are assessed in a transductive setting, which fails to reflect real-world scenarios. We introduce a rigorous *Inductive* evaluation setting that enforces strict temporal constraints, simulating the recommendation of citations for newly authored papers in the wild. Finally, we present *DAVINCI*, a novel reranking model that integrates profiler-derived confidence priors with semantic information via an adaptive vector-gating mechanism. Our system achieves new state-of-the-art results across multiple benchmark datasets, demonstrating superior efficiency and generalisability.

Chapter 4

A Formative Approach for Data Enrichment in a Transductive Setting

4.1 Introduction

Citing serves as the foundation of scientific research, fostering trust and reinforcing assertions within scholarly works. It is through meticulous citation that researchers acknowledge prior contributions, substantiate their own claims, and draft their work into the existing fabric of knowledge (Johnson et al., 2018). It enables trust in the scientific document thus curated. However, the very success of scientific endeavour has led to an exponential surge in published literature (Drozdz and Ladomery, 2024; Rousseau et al., 2023; Nane et al., 2023; Bornmann et al., 2021). The sheer volume of publications makes manual literature survey increasingly time-consuming and prone to overlooking crucial studies. This information deluge presents a significant impediment to researchers seeking to locate and refer pertinent papers (Datta et al., 2024; Bornmann et al., 2021; Nane et al., 2023; Bhagavatula et al., 2018; Johnson et al., 2018). Hence, it becomes increasingly crucial to formally streamline the process for authors to efficiently identify and integrate relevant prior work (Gu et al., 2022).

To address this ever-growing challenge, citation recommendation techniques are designed to intelligently assist researchers in navigating the vast landscape of scientific literature. Broadly, current citation recommendation approaches can be categorised into two distinct methodologies: "local" (Gu et al., 2022; Ostendorff et al., 2022; Robertson et al., 2009) and "global" recommendation (Ni et al., 2024; Ali et al., 2021; Xie et al., 2021). Local citation recommendation (LCR) operates with a fine-grained, context-sensitive approach, and is the theme of our research work. This methodology focuses on specific textual segments within a document – often termed as "citation context/excerpt". The core objective of local recommendation is to identify and suggest the most relevant citations that align semantically and conceptually with the immediate meaning and purpose of that particular passage. On the other hand, global citation recommendation recommends a list of suitable prior art for the entire document, mainly given the title and abstract or the whole document. However, the global recommendation does not specify the broad position of citation. To emphasise the importance of local citation

recommendation, consider the below citation excerpt:[1]

This can have extreme consequences in real-life scenarios such as autonomous cars `CitX`.

Examining the above context in isolation makes it challenging to predict the specific article cited at `CitX`. Henceforth, prior research (Medić and Šnajder, 2020; Gu et al., 2022) has integrated supplementary details like title and abstract to augment the accuracy of local citation suggestions. We posit that incorporating additional semantic information, such as taxonomical concepts, into the input could further enhance recommendation performance. Accordingly, we advocate for the utilisation of an article's taxonomy as a significant feature and its seamless integration into the recommendation framework. By harnessing information such as titles, abstracts, and taxonomies, we can narrow down the search scope. Additionally, we observe a deficiency in the candidate sets selected by current methodologies, with many cited articles being overlooked. To address this, we propose the inclusion of **Symbiosis**, which emulates human citation patterns, within our framework. Leveraging symbiotic relationships (Symbiosis) furnishes the model with an expanded pool of the most pertinent candidates for recommendation.

In this work, we propose the **SymTax** method (please see Figure 4.1) that operates through three distinct stages to recommend local citations effectively. Given inputs such as a citation excerpt, title, abstract, flat taxonomy concepts, an existing tree-structured taxonomy, and a citation context graph, it identifies relevant documents for citation. The SymTax method's innovation lies in several key aspects: first, it utilises a small language model to encode textual signals; second, it integrates flat taxonomy concepts with the tree-structured taxonomy thereby using hyperbolic separation for representation; third, it employs a symbiotic process to enrich the candidate set; and finally, it combines all signals for enhanced representation in the Reranker.

To ensure accurate citation recommendations, a citation recommendation system must operate with a dataset comprising a wide array of documents and comprehensive linkages between citing and cited documents across various contexts. However, existing datasets often lack the inclusion of all context linkages. Thus, employing a more robust approach, we have developed an expansive and diverse domain dataset **ArSyTa** that encompasses a broader range of relevant features and offers improved linkages between documents across a wider spectrum of contexts. This dataset holds significant promise for advancing research in the field of citation recommendation. Unlike the ACL-200 and RefSeer datasets, which feature curated contexts of fixed sizes, we curate richer contexts by incorporating complete information from adjacent sentences relative to the citation sentence. By leveraging symbiotic relationships, our method demonstrates substantial performance improvements, with Recall@5 gains of 26.66% and 39.25% on the ACL-200 and RefSeer datasets, respectively, compared to SOTA.

To summarise, we make the following contributions and discuss them in detail as the chapter follows:

- *Dataset*: We have constructed a dataset ArSyTa comprising 8.27 million comprehensive citation contexts across diverse domains, featuring richer density and

[1]Excerpt is borrowed from *Towards Consistency in Adversarial Classification* of (Meunier et al., 2022). The cited article is *An analysis of adversarial attacks and defenses on autonomous driving models* of (Deng et al., 2020b).

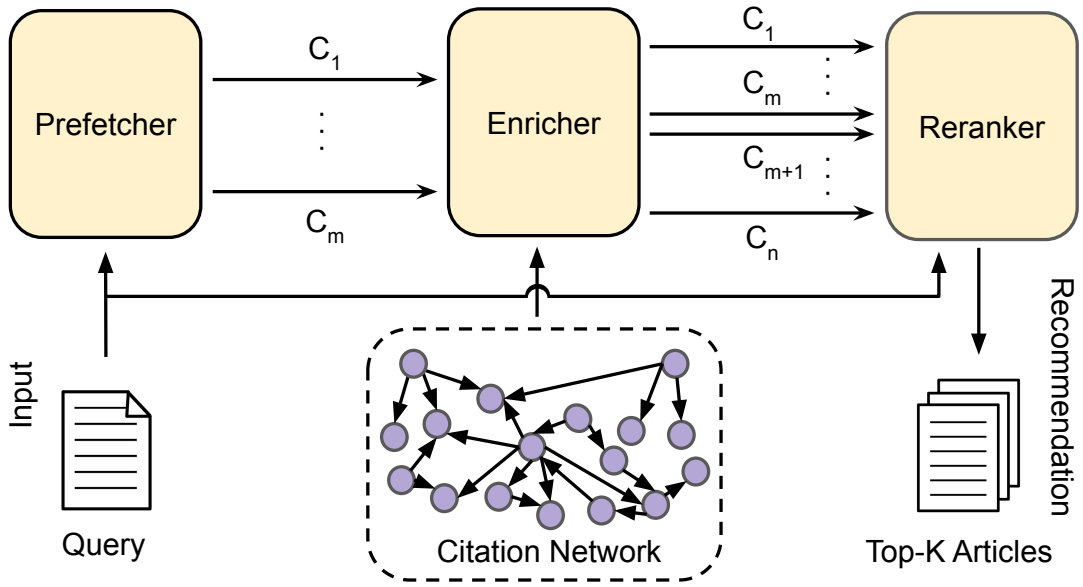


Figure 4.1: Proposed method *SymTax* consists of three essential modules. *Prefetcher* takes query consisting of citation context, title and abstract of the citing paper as input. For each candidate paper (C_i), *Enricher* encompasses knowledge from citation network and *Reranker* generates the final top-K ranked recommendations considering taxonomy as an additional signal. It presents an overview of our methodology showing top-K citation recommendations for a given query. C_i represents a candidate paper.

relevant features, including taxonomy concepts, to facilitate the task of citation recommendation.

- *Conceptual*: We explore the concept of Symbiosis from Biology and draw its analogy with human citation behaviour in the scientific research ecosystem and select a better pool of candidates.
- *Methodological*: We propose a novel taxonomy fused reranker that subsequently learns projections of fused taxonomies in hyperbolic space and utilises hyperbolic separation as a latent feature.
- *Empirical*: We perform extensive experiments, ablations, and analysis on five datasets and six metrics, demonstrating that SymTax consistently outperforms SOTA by huge margins.

4.2 Related Work

Local citation recommendation has lately drawn comparatively less interest than its global counterpart. He et al. (2010) introduced the task of local citation recommendation by using tf-idf based vector similarity between context and cited articles. Livne et al. (2014) extracted hand-crafted features from the citation excerpt and the remaining document text, and developed a system to recommend citations while the document is being drafted. The neural probabilistic model of Huang et al. (2015) determines the

citation probability for a given context by jointly embedding context and all articles in a shared embedding space. Ebesu and Fang (2017) proposed neural citation network based on encoder-decoder architecture. The encoder obtains a robust representation of citation context and further augments it via author networks and attention mechanism, which the decoder uses to generate the title of the cited paper. Dai et al. (2019) utilised stacked denoising autoencoders for representing cited articles, bidirectional LSTMs for citation context representation and attention principle over citation context to enhance the learning ability of their framework.

Jeong et al. (2020) proposed a BERT-GCN model which uses BERT (Kenton and Toutanova, 2019) to obtain embeddings for context sentences, and Graph Convolutional Network (Kipf and Welling, 2017) to derive embeddings from citation graph nodes. The two embeddings are then concatenated and passed through a feedforward neural network to obtain relevance between them. However, due to the high cost of computing GCN, as mentioned in Gu et al. (2022), BERT-GCN model was evaluated on tiny datasets containing merely a few thousand citation contexts. It highlights the limitation of scaling such GNN models for recommending citations on large datasets.

Medić and Šnajder (2020) suggested the use of global information of articles along with citation context to recommend citations. It computes semantic matching score between citation context and cited article text, and bibliographic score from the article's popularity in the community to generate a final recommendation score. Ostendorff et al. (2022) perform neighbourhood contrastive learning over the full citation graph to yield citation embeddings and then uses k-nearest neighbourhood based indexing to retrieve the top recommendations. The most recent work in local citation recommendation by Gu et al. (2022) proposed a two-stage recommendation architecture comprising a fast prefetching module and a slow reranking module. We build on the work of Gu et al. (2022) by borrowing their prefetching module and designing a novel reranking module and another novel module named **Enricher** that fits between Prefetcher and Reranker. We name our model as SymTax (Symbiotic Relationship and Taxonomy Fusion).

4.3 Proposed Dataset

In this section, we discuss the motivation, curation process and merits of our proposed dataset.

4.3.1 Motivation

Citation recommendation algorithms depend on the availability of the labelled data for training. However, curating such a dataset is challenging as full pdf papers must be parsed to extract citation excerpts and map the respective cited articles. Further, the constraint that cited articles should be present in the corpus eliminates a large proportion of it, thus reducing the dataset size considerably. e.g. FullTextPeerRead (Jeong et al., 2020) and ACL-200 (Medić and Šnajder, 2020) datasets contain only a few thousand papers and contexts. RefSeer (Medić and Šnajder, 2020) contains 0.6 million papers published till 2014 and hence is not up-to-date. Gu et al. (2022) released a large and recent arXiv-based dataset (we refer to it as arXiv(HAtten)) by following the same strategy adopted by

Dataset	# Contexts				# Papers	LCC	Deg	Pub Years
	Train	Val	Test	Total				
ACL-200	30,390	9,381	9,585	49,356	19,776	0.035	3.42	2009-2015
FTPR	9,363	492	6,814	16,669	4,837	0.036	2.84	2007-2017
RefSeer	3,521,582	124,911	126,593	3,773,086	624,957	0.033	4.46	-2014
arXiv	2,988,030	112,779	104,401	3,205,210	1,661,201	0.027	3.30	1991-2020
ArSyTa	8,030,837	124,188	124,189	**8,279,214**	474,341	**0.051**	**9.98**	**2007-2023**

Table 4.1: Statistics across various datasets indicate the largest, densest and most recent nature of our dataset, ArSyTa. FTPR is FullTextPeerRead, arXiv is arXiv(HAtten), and LCC and Deg are the average local clustering coefficient and average degree of the citation context network, respectively.

ACL-200 and FullTextPeerRead for extracting contexts. They consider 200 characters around the citation marker as the citation context. The above mentioned datasets have limited features, which may restrict the design of new algorithms for local citation recommendation. Thus, we propose a novel dataset ArSyTa[2] which is latest, largest and contains rich citation contexts with additional features.

4.3.2 Dataset Creation

We selected 475, 170 papers contained in an arXiv dump belonging to Computer Science (CS) categories from over 1.7 million scholarly papers spanning STEM disciplines available on arXiv. The papers are selected from April 2007-January 2023 publication dates to ensure current relevance. arXiv contains an extensive collection of scientific papers that offer innate diversity in different formatting styles, templates and written characterisation, posing a significant challenge in parsing pdfs. We comprehensively evaluate established frameworks, namely, arXiv Vanity[3], CERMINE[4], and GROBID[5], for data extraction. arXiv Vanity converts pdfs to HTML format for data extraction but produces inconsistent results, thus turning extraction infeasible in this scenario. CERMINE uses JAVA binaries to generate BibTeX format from pdf but fails to extract many references, thereby not providing the required level of information. GROBID is a state-of-the-art tool that accurately and efficiently produces easy-to-parse results in XML format with a standard syntax. We conduct extensive manual testing to assess parsing efficacy and finally choose GROBID as it adeptly parses more than 99.99% (i.e., 474, 341) of the documents.

We organise the constructed dataset into a directed graph. Nodes within the graph encapsulate a rich array of attributes, encompassing abstracts, titles, authors, submitters, publication dates, topics, categories within CS, and comments associated with each paper. Edges within the graph symbolise citations, carrying citation contexts and section

[2]ArSyTa: Arxiv Symbiotic Relationship Taxonomy Fusion

[3]https://github.com/arxiv-vanity/arxiv-vanity

[4]https://github.com/CeON/CERMINE

[5]https://github.com/kermitt2/grobid_client_python

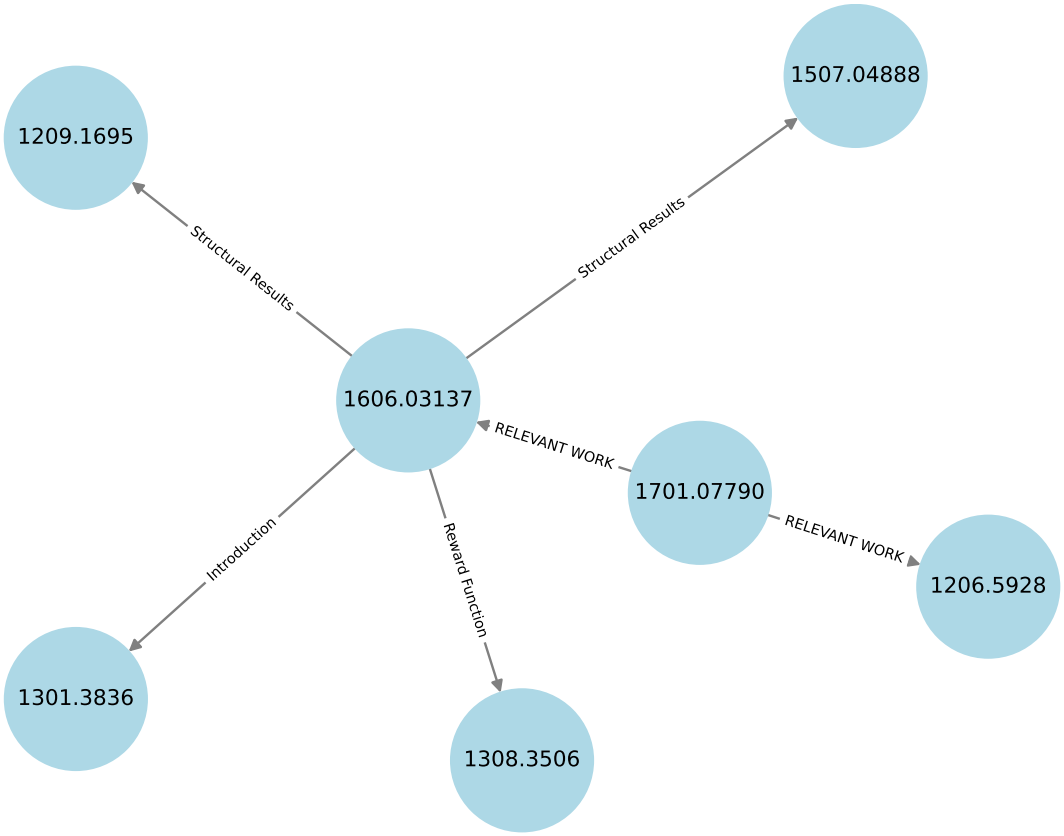


Figure 4.2: Sample NetworkX graph representation of ArSyTa. Nodes encapsulate paper details such as title, abstract, and arXiv category class. At the same time, edges denote citation contexts pertaining to the section headings, forming a structured network that enables visual exploration of scholarly relationships and information flow.

headings in which they appear. This provides a format that offers better visualisation and utilisation of data, as depicted concisely in Figure 4.2.

Unlike previously available datasets, which use a 200-character length window to extract citation context, we consider one sentence before and after the citation sentence as a complete citation context. We create a robust mapping function for efficient data retrieval. Since every citation does not contain a Digital Object Identifier, mapping citations to corresponding papers is challenging. The use of several citation formats and the grammatical errors adds a further challenge to the task. To expedite title-based searches that associate titles with unique paper IDs, we devise an approximate mapping function based on LCS (Longest Common Substring), but the sheer size of the number of papers makes it infeasible to run directly, as each query requires around 10 seconds. Finally, to identify potential matches, we employ an approximate hash function called MinHash LSH (Locality Sensitivity Hashing), which provides the top 100 candidates with a high probability for a citation existing in our raw database to be present in the candidate list. We then utilise LCS matching with a 0.9 similarity score threshold to give a final candidate, thus reducing the time to a few microseconds.

Finally, our dataset consists of 8.27 million citation contexts whereas the largest existing dataset, RefSeer, consists of only 3.7 million contexts. The dataset is essentially

comprised of contexts and the corresponding metadata only and not the research papers, as is the case with other datasets. Even after considering a relatively lesser number of papers as a raw source, we curated significantly more citation contexts (i.e., final data), thus showing the effectiveness of our data extraction technique. This is further supported empirically by the fact that our dataset has significantly higher values of average local clustering coefficient and average degree with respect to the other datasets (as shown in Table 4.1). Each citing paper and cited paper that corresponds to a citation context respectively belongs to a CS concept in the flat-level arXiv taxonomy that contains 40 classes. The distribution of category classes in arXiv taxonomy for ArSyTa is shown in Figure 4.3.

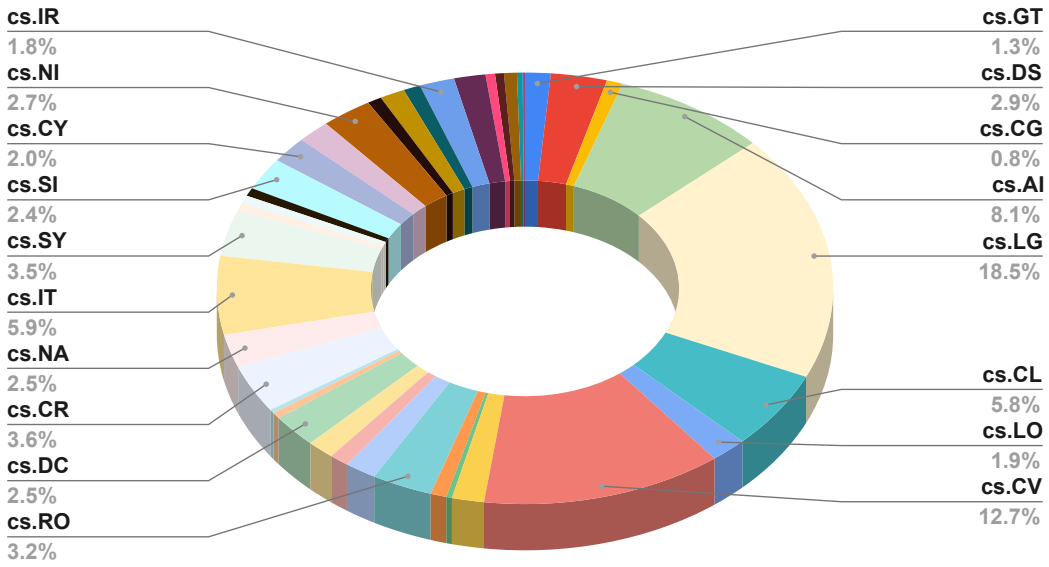


Figure 4.3: Statistics show the distribution of major category classes of flat-level arXiv taxonomy corresponding to ArSyTa. The highest number of research papers belong to Machine Learning (cs.LG) followed by Computer Vision (cs.CV), and Artificial Intelligence (cs.AI), respectively.

4.3.3 Technical Merits

ArSyTa offers the following merits over the existing datasets: (i) As shown in Table 4.1, ArSyTa is 2.2x and 2.6x larger than RefSeer and arXiv(HAtten), respectively. Also, our citation context network is more dense than all other datasets, clearly showing that our dataset creation strategy is better. (ii) It is the most recent dataset that contains papers till January 2023. (iii) It contains longer citation contexts and additional signals such as section heading and document category. (iv) ArSyTa is suitable for additional scientific document processing tasks that can leverage section heading as a feature or a label. (v) ArSyTa is more challenging than others as it contains papers from different publication venues with varied formats and styles submitted to arXiv.

4.4 SymTax Model

We discuss the detailed architecture of our proposed model, SymTax, as shown in Figure 4.4. It comprises a fast prefetching module, an enriching module and a slow and precise reranking module. We borrow an existing prefetching module from Gu et al. (2022) whereas an enriching module and a reranking module are our novel contributions in the overall recommendation technique. The subsequent subsections elaborate on the architectures of these three modules.

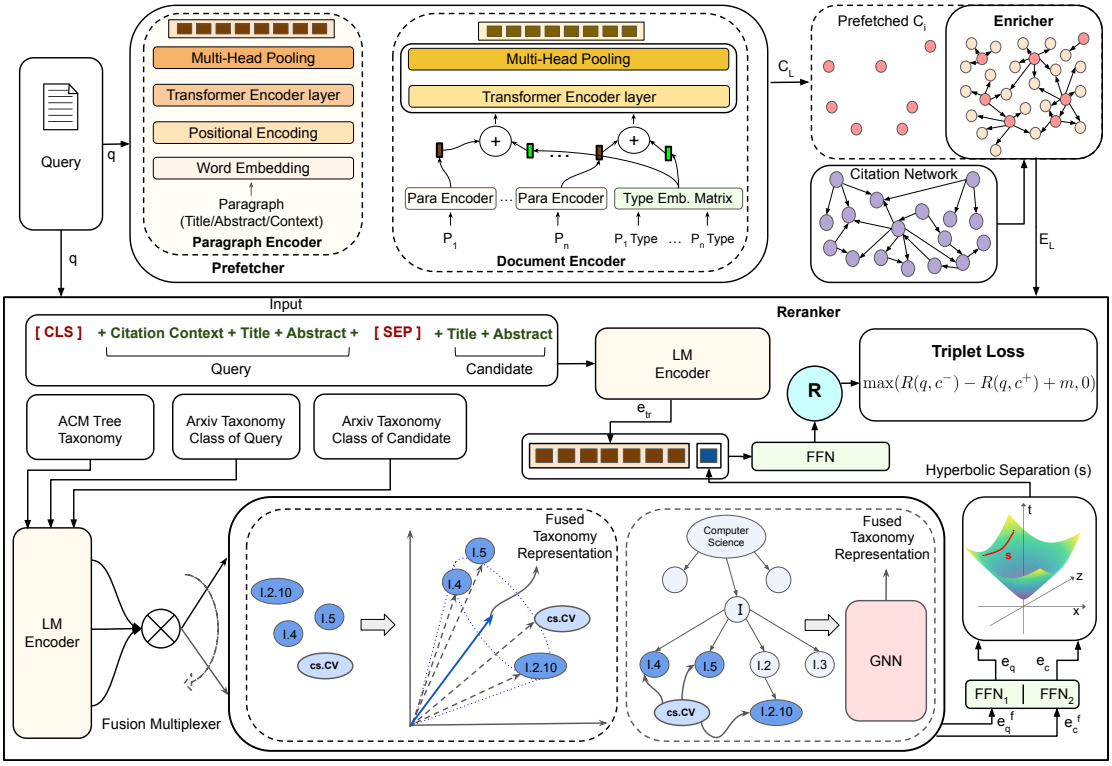


Figure 4.4: Architecture of SymTax. It consists of three essential modules – (a) Prefetcher, (b) Enricher, and (c) Reranker. The task of Enricher is to enrich the candidate list generated by Prefetcher and provide it as an input to Reranker. Reranker utilises taxonomy fusion and hyperbolic separation to yield final recommendation score (R). Mapping:- I.4: Image Processing and Computer Vision, I.5: Pattern Recognition, I.2.10: Vision and Scene Understanding, cs.CV: Computer Vision. Fusion Multiplexer enables switching between vector-based and graph-based taxonomy fusion.

4.4.1 Prefetcher

The task of the prefetching module is to provide an initial set of high-ranking candidates by scoring all the papers in the database with respect to the query context. It uses cosine similarity between query embedding and document embedding to estimate the relevance between query context and the candidate document. Prefetcher comprises two submodules, namely, Paragraph Encoder and Document Encoder. Paragraph Encoder computes the embedding of a given paragraph, i.e. title, abstract or citation context,

using a transformer layer followed by multi-head pooling. Document Encoder takes paragraph encodings as input along with paragraph types and passes them through a multi-head pooled transformer layer to obtain the final document embedding. The prefetched document candidates are found by identifying the top-K nearest document embeddings to the query embedding in terms of cosine similarity. The ranking is performed using a brute-force nearest neighbour search among all document embeddings. We adopt the prefetching module from Gu et al. (2022) and use it as a plugin in our overall recommendation technique.

4.4.2 Enricher

In general, prefetching is often followed by reranking in recommendation systems. Let $C_L = \{c_1, c_2, ..., c_m\}$ be the candidate list generated by prefetcher, and $G = (V, E)$ be the given citation network $s.t.\ c_i \in V \ \forall\ i \in \{1, 2, ..., m\}$.

The nodes in citation network represent papers and the directed edges represent citation relationship. Here, we propose a new module Enricher that augments the candidate list generated by the prefetcher and outputs an enriched candidate list. It generates an ego network for every article in the candidate list using the citation graph and excludes the incoming neighbours and the duplicates from the expanded network list. For every node $u \in V$, we define $N_o(u) \subset V$ as the ego network of u containing only the outgoing neighbours. Let E_L denotes the enriched list, then

$$\begin{aligned} E_L &= \{c_1, c_2, ..., c_m, N_o(c_1), N_o(c_2), ..., N_o(c_m)\} \\ E_L &= \{c_1, c_2, ..., c_m, c_{m+1}, c_{m+2}, ..., c_n\} \end{aligned} \tag{4.1}$$

where $\{\}$ represents a set operator. We then feed this enriched list as input to the reranker. The design notion of Enricher is inspired by Symbiosis (a.k.a. Symbiotic Relationship), a concept in Biology.

Symbiosis. The idea of including cited papers of identified candidates has been pursued in the literature (Cohan et al., 2020) but from the perspective of hard negatives. To the best of our knowledge, the concept of Enrichment has never been discussed earlier for citation recommendation to model the human citation behaviour. We identify two different types of citation behaviours that prevail in the citation ecosystem and draw a corresponding analogy with *mutualism* and *parasitism* that falls under the concept of Symbiosis. Symbiosis is a long-term relationship or interaction between two dissimilar organisms in a habitat. In our work, the habitat is the citation ecosystem, and the two dissimilar organisms are the candidate article and its neighbourhood.

We try to explain the citation phenomena through Symbiosis wherein the candidate and its neighbourhood either play the role of mutualism or parasitism. In mutualism, the query paper recommends either only the candidate paper under consideration or both the considered candidate paper and from its 1-hop outdegree neighbour network. On the other hand, in parasitism, the neighbour organism feeds upon the candidate to get itself cited, i.e., the query paper, rather than citing the candidate article, in turn, recommends from its outgoing edge neighbours. This whole idea, in practice, is analogous to human citation behaviour. When writing a research article, researchers often gather a few highly

relevant prior art and cite highly from their references. We can interpret this tendency as a slight human bias or highly as utilising the research crowd's wisdom. Owing to this, Enricher is only required at the inference stage. Nevertheless, it is a significantly important signal, as evident from the results in Table 4.2 and Table 4.3.

4.4.3 Reranker

The purpose of the reranker is to rerank the candidates fetched from previous modules with higher accuracy. Therefore, the reranker is generally slower than the prefetcher. It makes a fine-grain comparison between the query and each candidate, calculates a recommendation score and returns it as the final output. Our proposed reranker considers the relevance between query and candidate at two separate levels, namely (i) text relevance and (ii) taxonomy relevance. In text relevance, we concatenate the query text and candidate text with a [SEP] token and pass it through a Language Model (LM) to obtain the joint embedding $e_{tr} \in \mathbb{R}^d$. Query text is the concatenation of citation context, title and abstract of the query article, whereas the candidate text is the concatenation of title and abstract of the candidate paper. In taxonomy relevance, we perform taxonomy fusion and hyperbolic projections to calculate the separation between query and candidate.

Taxonomy Fusion. The inclusion of taxonomy fusion is an important and careful design choice. Intuitively, a flat-level taxonomy (arXiv concepts) does not have a rich semantic structure in comparison to a hierarchically structured taxonomy like ACM. In a hierarchical taxonomy, we have a semantic relationship in terms of generalisation, specialisation and containment. Mapping the flat concepts into hierarchical taxonomy infuses a structure into the flat taxonomy. It also enriches the hierarchical taxonomy as we get equivalent concepts from the flat taxonomy. Each article in our proposed dataset ArSyTa consists of a feature category that represents the arXiv taxonomy[6] class it belongs to. Since ArSyTa contains papers from the CS domain, so we have a flat arXiv taxonomy. e.g. cs.LG and cs.CV represents Machine Learning and Computer Vision classes, respectively.

We now propose the fusion of flat-level arXiv taxonomy with ACM tree taxonomy[7] to obtain rich feature representations for the category classes. We mainly utilise the subject class mapping information mentioned in the arXiv taxonomy and domain knowledge to create a class taxonomy mapping from arXiv to ACM. e.g. cs.CV is mapped to ACM classes I.2.10, I.4 and I.5 (as shown in Figure 4.4). We released this mapping config file along with our data. We employ two fusion strategies, namely vector-based and graph-based. In vector-based fusion, the classes are passed through LM and their aggregated vector is obtained by averaging out class vectors in feature space. In graph-based fusion, we first form a directed acyclic graph by injecting arXiv classes into the ACM tree and creating directed edges between them. We initialise node embeddings using LM and run Graph Neural Network (GNN) algorithm to learn fused representations. We consider GAT(Veličković et al., 2018) and APPNP(Gasteiger et al., 2019) as GNN algorithms and observe their performance as the same. The final representations of cs.{} nodes

[6]https://arxiv.org/category_taxonomy

[7]https://tinyurl.com/22t2b43v

represent the fused representations learnt. Empirically, we can clearly observe that the fusion of concepts helps to attain significant performance gains as shown in Table 4.3.

Hyperbolic Separation. Hyperbolic spaces have recently gained momentum in deep learning due to their high capacity and tree-likeliness properties, thus making them suitable for learning better representations for hierarchical data (Ganea et al., 2018). Motivated by the works of Sawhney et al. (2022) and Ganea et al. (2018), and the notion that Euclidean space may not fully capture the complex characteristics of hierarchical data, we use hyperbolic distance as our metric to compute taxonomy relevance. Let q and c denote query and candidate, respectively, where $c \in E_L$. Let ${e_q}^f, {e_c}^f \in \mathbb{R}^D$ represents the fusion embeddings for q and c respectively. We pass fusion embeddings through a feed forward network $h_\theta(.)$ and then compute hyperbolic separation between query and candidate to project the embeddings into hyperbolic space. So, we obtain the following representations

$$e_q = h_\theta({e_q}^f) \in \mathbb{R}^d; e_c = h_\theta({e_c}^f) \in \mathbb{R}^d \tag{4.2}$$

and then compute the hyperbolic separation s between e_q and e_c as follows

$$s(e_q, e_c) = 2\tan^{-1}\left(\left\|(-e_c) \oplus e_q\right\|\right) \tag{4.3}$$

where $\oplus$ represents Möbius addition and for a pair of points $a, b \in \mathcal{B}$, is defined as,

$$a \oplus b := \frac{(1 + 2\langle a, b\rangle + \|b\|^2)a + (1 - \|a\|^2)b}{1 + 2\langle a, b\rangle + \|a\|^2\|b\|^2}$$

where $\langle\cdot,\cdot\rangle$, $\|\cdot\|$ are Euclidean inner product and norm.

Final Recommendation. We use this hyperbolic separation representing the taxonomy relevance as a latent feature and concatenate ($\odot$) it with the text relevance embedding e_{tr}. This step ensures that category classes with similar concepts in the taxonomy learn to embed themselves closely in the manifold. We then pass this relevance embedding through a feed forward network ($g_\theta(.)$) and apply the sigmoid activation function ($\sigma(.)$) to get the final relevance score, defined as,

$$R = \sigma(g_\theta(e_{tr} \odot s)) \in (0, 1) \tag{4.4}$$

To interpret R as the final recommendation score, we employ and minimise the Triplet loss, L:

$$L = \max(R(q, c^-) - R(q, c^+) + m, 0) \tag{4.5}$$

where m is margin, and c^+ and c^- are positive and negative candidates respectively. We adopt a simple technique for mining triplets for a query. We choose cited paper as positive candidate and randomly select papers from the candidate list as negative candidates.

4.5 Experiments and Results

This section illustrates the various baselines, evaluation metrics and datasets used to benchmark our proposed method followed by the performance comparison.

4.5.1 Baselines

We consider evaluating various available systems for comparison. BM25 (Robertson et al., 2009): It is a prominent ranking algorithm, and we consider its several available implementations and choose Elastic Search implementation[8] as it gives the best performance with the highest speed. SciNCL (Ostendorff et al., 2022): We use its official implementation available on GitHub[9]. HAtten (Gu et al., 2022): We use its official implementation available on GitHub[10]. NCN (Ebesu and Fang, 2017) could have been a potential baseline; however, as reported by Medić and Šnajder (2020), the results mentioned could not be replicated. DualLCR (Medić and Šnajder, 2020): It is essentially a ranking method that requires a small and already existing list of candidates containing the ground truth, which turns it into an artificial setup that, in reality, does not exist. This unfair setup is also reported by Gu et al. (2022), which is state-of-the-art in our task. Thus for a fair comparison, we could not consider it in comparing our final results.

4.5.2 Evaluation Metrics

To stay consistent with the literature that uses Recall@10 and Mean Reciprocal Rank (MRR) as the evaluation metrics, we additionally use Normalised Discounted Cumulative Gain (NDCG@10) and Recall@K for different values of K to obtain more insights from the recommendation performance. Recall@K measures the presence of a cited paper in top-K recommendations. To assess the quality of the ranking order, we use MRR, which rewards systems for placing the correct item higher in the list by returning the average of the reciprocal ranks of the correct recommendations, and NDCG@K, which similarly provides a greater reward for correct items ranked at the very top by applying a logarithmic discount to the relevance of items based on their position. For all metrics, we report the average over all test queries in percentage, where higher values indicate better performance.

4.5.3 Datasets

We consider the following five datasets for all the experiments:

- **ACL-200** contains papers published at ACL venues. It is a processed version of the ACL-ARC dataset created using ParsCit[11], a string parsing package based on conditional random field. It contains citation contexts by considering ±200 characters around the citation placeholder.

[8] https://github.com/kwang2049/easy-elasticsearch
[9] https://github.com/malteos/scincl
[10] https://github.com/nianlonggu/Local-Citation-Recommendation
[11] https://github.com/knmnyn/ParsCit

- **FullTextPeerRead** is an expansion of PeerRead dataset that contains the peer reviews of papers submitted to top venues in the Artificial Intelligence domain. So, FullTextPeerRead contains the citation contexts from the papers present in the PeerRead dataset.

- **RefSeer** dataset is curated by extracting scientific articles belonging to various engineering domains. A citation excerpt is taken as the text of ±200 characters around the citation marker. It is a large dataset that contains 3.7 million citation contexts.

- **arXiv (HAtten)** is created using arXiv papers from a large and diverse corpus of scientific articles contained in S2ORC[12]. For every paper having its full text available, a citation excerpt is considered if the cited paper is also present in the arXiv database. Following the similar trend setup by ACL-200 and RefSeer, this dataset is also curated by considering the words in the ±200 character window around the citation marker.

- **ArSyTa** is discussed in detail in Section 4.3 above.

4.5.4 Implementation Details

We run all experiments on an NVIDIA DGX A100 GPU cluster. We use Adam optimiser with $\beta 1 = 0.9$ and $\beta 2 = 0.999$. We set learning rate to $1e^{-4}$ and weight decay to $1e^{-5}$ in Prefetcher, whereas for Reranker these were set to $1e^{-5}$ and to $1e^{-2}$ respectively for fine-tuning LM. We choose the top 100 candidates returned by Prefetcher as input to Enricher. We choose random seed = 12 for sampling 10k citation contexts from ArSyTa for conducting Quantitative Analysis as discussed in Table 4 and Table 5. We set margin $m = 0.1$ in the triplet loss function. The maximum sequence length for LM is 512. The values of D and d in the Reranker are 768 and 512 respectively. Since ArSyTa is a highly dense network, we sort the enriched candidate list by frequency count and take the top 300 candidates with the highest frequency count to run the Reranker further. All the results are reported as an average of 3 runs.

4.5.5 Performance Comparison

As evident from Table 4.2, our evaluation shows the superior performance of SymTax on all metrics across all the datasets. We consider two different variants of SymTax in our main results comparison (i) SpecG: with SPECTER (Cohan et al., 2020) as LM and graph-based taxonomy fusion, and (ii) SciV: with SciBERT (Beltagy et al., 2019) as LM and vector-based taxonomy fusion. SPECTER and SciBERT are two state-of-the-art LMs trained on scientific text. SciV performs as the best model on ACL-200, FullTextPeerRead, RefSeer and ArSyTa on all metrics. SpecG performs best on arXiv(HAtten) on all metrics and results in a marginally less R@20 score than SciV. We observe the highest scores on FullTextPeerRead followed by ACL-200. It is due to the fact that these datasets lack diversity to a large extent. e.g. FullTextPeerRead is

[12]https://github.com/allenai/s2orc

extracted from papers belonging to Artificial Intelligence field, and ACL-200 contains papers published at ACL venues.

In contrast, we observe the lowest scores on ArSyTa followed by arXiv(HAtten). The common reason driving these performance trends is that both of these arXiv-based datasets contain articles from different publication venues with various formats, styles and domain areas, making the learning difficult and recommendation challenging. Our reasoning is further supported by the fact that ArSyTa is the latest dataset, and thus contains the maximum amount of diverse samples and is shown to be the toughest dataset for recommending citations. To summarise, we obtain performance gains in Recall@5 of 26.66%, 23.65%, 39.25%, 19.74%, 22.56% with respect to SOTA on ACL-200, FullTextPeerRead, RefSeer, arXiv(HAtten) and ArSyTa respectively. The results show that NDCG is a tough metric compared to the commonly used Recall, as it accounts for the relative order of recommendations. Since the taxonomy class attribute is only available for our proposed dataset, we intentionally designed SymTax to be highly modular for better generalisation, as evident in Table 4.2.

4.6 Analysis

We conduct extensive analysis to assess further the modularity of SymTax, the importance of different modules, combinatorial choice of LM and taxonomy fusion, and the usage of hyperbolic space over Euclidean space. Furthermore, we analysed the effect of using section heading as an additional signal.

4.6.1 Ablation Study

We perform an ablation study to highlight the importance of Symbiosis, taxonomy fusion and hyperbolic space. We consider two variants of SymTax, namely SciBERT_vector and SPECTER_graph. For each of these two variants, we further conduct three experiments by (i) removing the Enricher module that works on the principle of Symbiosis, (ii) not considering the taxonomy attribute associated with the citation context and (iii) using Euclidean space to calculate the separation score.

As evident from Table 4.3, Symbiosis exclusion results in a drop of 21.40% and 24.45% in Recall@5 and NDCG respectively for SciBERT_vector whereas for SPECTER_graph, it leads to a drop of 17.84% and 20.32% in Recall@5 and NDCG respectively. Similarly, taxonomy exclusion results in a drop of 34.94% and 27.88% in Recall@5 and NDCG respectively for SciBERT_vector whereas for SPECTER_graph, it leads to a drop of 14.81% and 12.51% in Recall@5 and NDCG respectively. It is clear from Table 4.3 that the use of Euclidean space instead of hyperbolic space leads to performance drop across all metrics in both variants. Exclusion of Symbiosis impacts higher recall metrics more in comparison to excluding taxonomy fusion and hyperbolic space.

Model	R@5	R@10	R@20	R@50	NDCG	MRR
ACL-200						
BM25	13.74	19.39	25.31	34.86	8.08	10.74
SciNCL	15.17	22.50	31.76	44.67	10.44	6.69
HAtten	41.86	49.97	55.79	59.62	30.02	23.62
SymTax (SpecG)	45.29	58.97	70.38	83.96	30.34	21.26
SymTax (SciV)	**53.02**	**65.29**	**76.40**	**88.03**	**38.18**	**29.55**
FullTextPeerRead						
BM25	26.88	33.71	40.76	51.68	17.50	21.36
SciNCL	21.73	31.04	42.17	57.22	14.52	9.35
HAtten	50.27	57.88	62.63	65.14	35.66	28.47
SymTax (SpecG)	46.11	62.66	76.19	88.99	30.87	20.90
SymTax (SciV)	**62.16**	**75.05**	**83.98**	**92.94**	**44.72**	**35.00**
RefSeer						
BM25	17.37	21.92	26.77	33.65	11.85	14.24
SciNCL	9.67	14.86	21.14	30.89	7.00	4.50
HAtten	26.72	33.74	39.85	46.37	19.25	14.66
SymTax (SpecG)	27.24	38.31	49.60	65.12	19.42	13.53
SymTax (SciV)	**37.21**	**48.45**	**59.16**	**72.64**	**26.76**	**19.93**
arXiv(HAtten)						
BM25	15.29	19.73	24.55	31.60	10.19	12.45
SciNCL	10.76	16.04	2227	31.92	7.37	4.68
HAtten	24.26	32.92	40.97	49.49	16.51	11.36
SymTax (SpecG)	**29.05**	**40.95**	53.08	**69.92**	**19.83**	**13.23**
SymTax (SciV)	28.17	39.97	**53.17**	69.87	19.28	12.84
ArSyTa						
BM25	17.77	22.03	26.40	32.69	11.55	10.06
SciNCL	16.12	21.55	27.57	36.24	10.88	7.51
HAtten	15.67	20.46	25.22	30.70	10.74	7.66
SymTax (SpecG)	20.61	27.47	34.99	46.68	14.21	10.03
SymTax (SciV)	**21.78**	**29.76**	**38.08**	**50.29**	**14.86**	**10.18**

Table 4.2: Results clearly show that SymTax consistently outperforms SOTA (HAtten) across datasets on all metrics. Best results are highlighted in bold. Abbreviation: SpecG:- SPECTER_Graph; SciV:- SciBERT_Vector; R:- Recall.

SymTax Variant	R@5	R@10	R@20	R@50	NDCG	MRR
SciBERT_vector	21.78	29.76	38.08	50.29	14.86	10.18
– Symbiosis	17.94	22.34	26.51	31.05	11.94	8.60
– Taxonomy	16.14	23.77	32.15	46.11	11.62	7.87
– Hyperbolic	19.05	26.78	35.07	47.19	13.16	8.91
SPECTER_graph	20.61	27.47	34.99	46.68	14.21	10.03
– Symbiosis	17.49	21.78	25.98	30.79	11.81	8.62
– Taxonomy	17.95	25.07	33.84	47.33	12.63	8.74
– Hyperbolic	20.28	26.69	34.44	46.41	13.86	9.81

Table 4.3: Ablation shows importance of Symbiosis, taxonomy fusion and hyperbolic space on ArSyTa. Excluding Symbiosis reduces the metrics more as compared to the exclusion of taxonomy and hyperbolic space.

SymTax Variant	R@5	R@10	R@20	R@50	NDCG	MRR
SciBERT_graph	15.89	24.21	33.10	46.07	11.16	7.15
SPECTER_vector	15.52	23.24	32.52	46.07	10.92	7.12
SPECTER_graph	*20.25*	*26.67*	*34.56*	*46.54*	*13.85*	**9.81**
SciBERT_vector	**20.97**	**28.76**	**37.54**	**49.39**	**14.32**	*9.78*

Table 4.4: Analysis on choice of LM and taxonomy fusion on 10k random samples from ArSyTa. Best results are highlighted in bold and second best are italicised.

4.6.2 Choice of LM and Taxonomy Fusion

We consider two available LMs, i.e. SciBERT and SPECTER, and the two types of taxonomy fusion, i.e. graph-based and vector-based. This results in four variants, as shown in Table 4.4. As evident from the results, SciBERT_vector and SPECTER_graph are the best-performing variants. So, the combinatorial choice of LM and taxonomy fusion plays a vital role in model performance. The above observations can be attributed to SciBERT being a LM trained on plain scientific text. In contrast, SPECTER is a LM trained with Triplet loss using 1-hop neighbours of the positive sample from the citation graph as hard negative samples. So, SPECTER embodies graph information inside itself, whereas SciBERT does not.

4.6.3 Use of Section Heading

We conduct another quantitative analysis using the section heading as an additional signal in our reranking module. We concatenate the section heading with query context in reranker and run our two SymTax variants. From Table 4.5, we can observe that using section heading leads to a significant performance drop in SciBERT_vector for all the metrics. However, for SPECTER_graph, the overall performance remains nearly the same

SymTax Variant	R@5	R@10	R@20	R@50	NDCG	MRR
SciBERT_vector	20.97	28.76	37.54	49.39	14.32	9.78
+ Section	15.56	22.89	31.93	47.07	11.23	7.63
SPECTER_graph	20.25	26.67	34.56	46.54	13.86	9.81
+ Section	20.05	27.72	36.32	50.01	13.85	9.50

Table 4.5: Analysis on the inclusion of section heading as a feature on 10k random samples from ArSyTa data. The results indicate that using section heading as a feature acts as a noise as the citation contexts are already rich.

with some improvement for higher Recall values. Both of these patterns clearly indicate that using section heading as a feature acts as a noise, and thus the citation contexts are already rich. Since our proposed dataset contains this additional feature, it is suitable for two additional tasks: context-specific citation generation (Wang et al., 2022b), and section heading prediction for a given citation context.

4.7 Summary

In this work, we presented a model for local citation recommendation that leverages the notion of Symbiosis from Biology, and we drew its analogy with human citation behaviour. We proposed the notion of taxonomy fusion for learning rich concept representations and project them into hyperbolic space to derive a latent feature. We introduced a novel dataset that is comparatively large, dense, recent and more challenging than other existing datasets. Through several experiments and analyses, we proved our model as highly modular, which can run on datasets with comparatively few signals and accommodate additional signals as well. Our model consistently outperformed state-of-the-art by huge margins for all evaluation metrics across all datasets.

Chapter 5

A Scalable and Inductive Approach through Inward-Signal Enrichment

5.1 Navigating the Pitfalls of Modern Local Citation Recommendation

The state-of-the-art local citation recommendation (LCR) systems also use global information, primarily, the title and abstract as metadata along with the citation context. This context-awareness allows for targeted recommendations, directly relevant to the specific claims or discussions being made at a given point in the text. For instance, if a sequence of sentences describes a novel experimental method, a local citation recommendation system would aim to suggest papers that previously introduced, utilised, or evaluated similar methodologies.

The latest work in local citation recommendation, i.e., SymTax, (Goyal et al., 2024), additionally uses symbiotic relationship and taxonomy fusion for recommending better citations locally for a given excerpt. SymTax, which is a three stage recommendation architechture uses a state-of-the-art Prefetcher to retrieve suitable candidates and employs an enricher to create an enriched candidate pool for further reranking. Enricher captures symbiotic relationship between candidate and its neighbourhood by mimicing the human citation behaviour. When constructing a research document, it is a common phenomenon that researchers frequently concentrate their citations on a few selected seminal or highly relevant prior works, drawing disproportionately from the references within these key papers. From a pragmatic perspective, it represents an efficient approach to literature review, allowing researchers to quickly access a concentrated body of foundational knowledge and influential contributions. Paradoxically, it reflects a degree of inherent human bias, most likely a form of anchoring or confirmation bias, which keeps on perpetuating in the citation ecosystem.

This incorporation and simple modelling of public wisdom has shown impressive gains in the citation recommendation. However, this leads to a tradeoff with the introduction of bias. Moreover, the entire candidate retrieval significantly increases computational overhead for the end-to-end citation recommendation system composed of a downstream reranker. A further limitation of current state-of-the-art work is its reliance on paper taxonomy to boost reranking performance. This dependency curtails

the model's generalisability, as this specific meta-information is present in only one of the five benchmark datasets. Moreover, we will now shed light on the current training and evaluation practice of local citation recommendation systems operating in a setting that deviates from real-world scenarios.

More recently, (Çelik and Tekir, 2025) proposed CiteBART by performing continual pre-training of BART-base to generate parenthetical author-year strings directly for an input citation context. We identify two critical flaws in this setup: (i) the generative nature leads to hallucinations of non-existent citations, and (ii) the framework is semantically decoupled from the research content. By focusing on "author-year" strings, the model treats research as a function of primary authors' names (e.g., "Celik" or "Goyal") rather than the substantive scientific content, which is fundamentally independent of such identifiers. Crucially, this generative approach relies heavily on author-year surface forms rather than the underlying research contributions, thus creating an artificial recommendation landscape. This is concerning, given that LLM policy of premier conferences like NeurIPS considers hallucinated citations to be grounds for a paper's rejection or revocation.

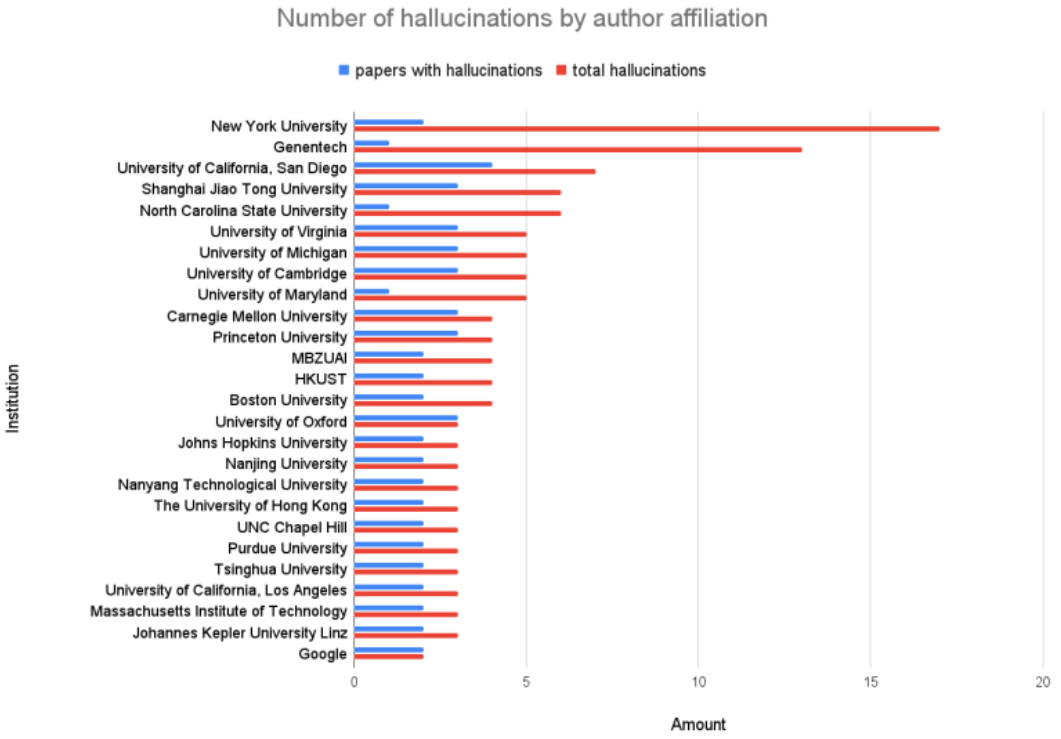


Figure 5.1: Courtesy (Shmatko et al., 2026): Distribution of hallucinations by author's institution. A paper with 2 hallucinations with any authors from University A and University B will count as 2 hallucinations and 1 paper with hallucinations for both universities, independent of the number of authors from either university.

Moreover, a very recent analysis (Shmatko et al., 2026) of 4, 841 papers accepted in NeurIPS 2025 by GPTZero, uncovered 100s of hallucinated citations missed by the 3+ reviewers who evaluated each paper. Since NeurIPS 2025 had an acceptance rate for

Published Paper	GPTZero Scan	Example of Verified Hallucination	Comment
SimWorld: An Open-ended Simulator for Agents in Physical and Social Worlds	Sources AI	John Doe and Jane Smith. Webvoyager: Building an end-to-end web agent with large multimodal models. arXiv preprint arXiv:2401.00001, 2024.	Article with a matching title exists here. Authors are obviously fabricated. arXiv ID links to a different article.
Unmasking Puppeteers: Leveraging Biometric Leakage to Expose Impersonation in AI-Based Videoconferencing	Sources AI*	John Smith and Jane Doe. Deep learning techniques for avatar-based interaction in virtual environments. IEEE Transactions on Neural Networks and Learning Systems, 32(12):5600-5612, 2021. doi: 10.1109/TNNLS.2021.3071234. URL https://ieeexplore.ieee.org/document/307123	No author or title match. Doesn't exist in publication. URL and DOI are fake.
Unmasking Puppeteers: Leveraging Biometric Leakage to Expose Impersonation in AI-Based Videoconferencing	Sources AI*	Min-Jun Lee and Soo-Young Kim. Generative adversarial networks for hyper-realistic avatar creation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 1234-1243, 2022. doi: 10.1109/CVPR.2022.001234. URL https://ieeexplore.ieee.org/ document/00123	No author or title match. Doesn't exist in publication. URL and DOI are fake.
SimWorld-Robotics: Synthesizing Photorealistic and Dynamic Urban Environments for Multimodal Robot Navigation and Collaboration	Sources AI*	Firstname Lastname and Others. Drivlme: A large-scale multi-agent driving benchmark, 2023. URL or arXiv ID to be updated.	No title or author match. Potentially referring to this article, but year is off (2024)

Figure 5.2: Courtesy (Shmatko et al., 2026): A few examples of verified hallucinated citations from NeurIPS 2025 accepted papers.

main track papers of 24.52%, each of these papers beat out 15,000 other papers despite containing one or more hallucinations. Figure 5.1 depicts 100 confirmed hallucinations spanning over 51 NeurIPS papers, which were not previously reported. Similarly, GPTZero (Esau et al., 2025) scanned 300 out of 20,000 under-review submissions in ICLR 2026 and discovered 50 submissions included atleast one hallucinated citation, which were not previously reported. Worryingly, each of these submissions has already been reviewed by 3-5 peer experts, most of whom missed the fake citation(s). This failure suggests that some of these papers might have been accepted by ICLR without any intervention. Some had average ratings of 8/10, meaning they would almost certainly have been published. According to the ICLR's editorial policy, even a single, clear

hallucitation is an ethics violation that could lead to the paper's rejection. Figure 5.2 and Figure 5.3 show a few hallucinated citation examples each from NeurIPS and ICLR as reported in Shmatko et al. (2026) and Esau et al. (2025).

Title	Average Review Rating	Paper Link	Citation Check Scan Link	Example of Verified Hallucination	Comment
TamperTok: Forensics-Driven Tokenized Autoregressive Framework for Image Tampering Localization	8.0	TamperTok: Forensics-Driven Tokenized Autoregressive Framework for Image Tampering Localization | OpenReview	https://app.gptzero.me/documents/4645494f-70eb-40bb-aea7-0007e13f7179/share	Chong Zou, Zhipeng Wang, Ziyu Li, Nan Wu, Yuling Cai, Shan Shi, Jiawei Wei, Xia Sun, Jian Wang, and Yizhou Wang. Segment everything everywhere all at once. In Advances in Neural Information Processing Systems (NeurIPS), volume 36, 2023.	This paper exists, but all authors are wrong.
MixtureVitae: Open Web-Scale Pretraining Dataset With High Quality Instruction and Reasoning Data Built from Permissive Text Sources	8.0	MixtureVitae: Open Web-Scale Pretraining Dataset With High Quality Instruction and Reasoning Data Built from Permissive Text Sources | OpenReview	https://app.gptzero.me/documents/bfd10666-ea2d-454c-9ab2-75faa8b84281/share	Dan Hendrycks, Collin Burns, Steven Basart, Andy Critch, Jerry Li, Dawn Ippolito, Aina Lapedriza, Florian Tramer, Rylan Macfarlane, Eric Jiang, et al. Measuring massive multitask language understanding. In Proceedings of the International Conference on Learning Representations (ICLR), 2021.	The paper and first 3 authors match. The last 7 authors are not on the paper, and some of them do not exist
Catch-Only-One: Non-Transferable Examples for Model-Specific Authorization	6.0	Catch-Only-One: Non-Transferable Examples for Model-Specific Authorization | OpenReview	https://app.gptzero.me/documents/9afb1d51-c5c8-48f2-9b75-250d95062521/share	Dinghuai Zhang, Yang Song, Inderjit Dhillon, and Eric Xing. Defense against adversarial attacks using spectral regularization. In International Conference on Learning Representations (ICLR), 2020.	No Match
OrtSAE: Orthogonal Sparse Autoencoders Uncover Atomic Features	6.0	OrtSAE: Orthogonal Sparse Autoencoders Uncover Atomic Features | OpenReview	https://app.gptzero.me/documents/e3f155d7-067a-4720-adf8-65dc9dc714b9/share	Robert Huben, Logan Riggs, Aidan Ewart, Hoagy Cunningham, and Lee Sharkey. Sparse autoencoders can interpret randomly initialized transformers, 2025. URL https://arxiv.org/abs/2501.17727.	This paper exists, but all authors are wrong.
Principled Policy Optimization for LLMs via Self-Normalized Importance Sampling	5.0	Principled Policy Optimization for LLMs via Self-Normalized Importance Sampling | OpenReview	https://app.gptzero.me/documents/54c8aa45-c97d-48fc-b9d0-d491d54df8d3/share	David Rein, Stas Gaskin, Lajanugen Logeswaran, Adva Wolf, Oded teht sun, Jackson H. He, Divyansh Kaushik, Chitta Baral, Yair Carmon, Vered Shwartz, Sang-Woo Lee, Yoav Goldberg, C. J. H. un, Swaroop Mishra, and Daniel Khashabi. Gpqa: A graduate-level google-proof q\&a benchmark, 2023	All authors except the first are fabricated.
PDMBench: A Standardized Platform for Predictive Maintenance Research	4.5	PDMBench: A Standardized Platform for Predictive Maintenance Research | OpenReview	https://app.gptzero.me/documents/5c55afe7-1689-480d-ac44-9502dc0f9229/share	Andrew Chen, Andy Chow, Aaron Davidson, Arjun DCunha, Ali Ghodsi, Sue Ann Hong, Andy Konwinski, Clemens Mewald, Siddharth Murching, Tomas Nykodym, et al. Mlflow: A platform for managing the machine learning lifecycle. In Proceedings of the Fourth International Workshop on Data Management for End-to-End Machine Learning, pp. 1-4. ACM, 2018.	Authors and conference match this paper, but title is somewhat different and the year is wrong.
IMPQ: Interaction-Aware Layerwise Mixed Precision Quantization for LLMs	4.5	IMPQ: Interaction-Aware Layerwise Mixed Precision Quantization for LLMs | OpenReview	https://app.gptzero.me/documents/5461eefd-891e-4100-ba1c-e5419af520c0/share	Chen Zhu et al. A survey on efficient deployment of large language models. arXiv preprint arXiv:2307.03744, 2023.	The arXiv ID is real, but the paper has different authors and a different title.
C3-OWD: A Curriculum Cross-modal Contrastive Learning Framework for Open-World Detection	4.5	C3-OWD: A Curriculum Cross-modal Contrastive Learning Framework for Open-World Detection | OpenReview	https://app.gptzero.me/documents/c07521cd-2757-40a2-8dc1-41382d7eb11b/share	K. Marino, R. Salakhutdinov, and A. Gupta. Fine-grained image classification with learnable semantic parts. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4500-4509, 2019.	Authors and subject match this paper

Figure 5.3: Courtesy (Esau et al., 2025): A few examples of verified hallucinated citations from ICLR 2026 under-review papers.

With enough insights and motivation from the above discussion on state-of-the-art, we make the following major contributions and discuss them in detail as the chapter follows:

- We introduce the Profiler, a lightweight, non-learnable module for candidate retrieval. It is remarkably efficient and free from confirmational bias, yet it outperforms the sequential combination of Prefetcher and Enricher.
- We demonstrate the importance of a paper's "public profile", i.e., how the research ecosystem perceives a paper, as a remarkably vital signal for recommendation.
- We develop the DAVINCI reranker, which discriminatively integrates confidence priors with textual semantics via an adaptive vector-gating mechanism. Unlike previous state-of-the-art, it is architecturally generalisable across diverse datasets without requiring special metadata like taxonomies.

- We establish a new state-of-the-art, demonstrating that DAVINCI surpasses both specialised LCR systems and massive-scale open-source rerankers adapted for this task.
- Finally, we introduce and benchmark LCR in an inductive setting, providing a more realistic evaluation framework for citations "in the wild."

5.2 Proposed Work

5.2.1 Problem Formulation

We formulate the task of local citation recommendation as a two-stage retrieval and reranking problem, designed to handle the immense scale of modern scholarly corpora. Given a query instance $q = (S_q, M_q)$ — comprising a snippet of citation context S_q and the source document's meta information M_q characterised by its title T_q and abstract A_q — and a large corpus of scientific documents $C = \{D_i\}$, the process is as follows. First, in the **retrieval stage**, our novel **Profiler** module efficiently retrieves an initial candidate set $C_q \subset C$, where $|C_q| \ll |C|$. For each candidate document $c_i \in C_q$, Profiler also yields a confidence score, s_i, which serves as an initial estimate of its relevance. Second, in the **reranking stage**, our proposed **DAVINCI** module ingests this candidate set and their associated confidence scores. It computes a final, fine-grained relevance score, $f_{\text{DAVINCI}}(q, c_i, s_i)$, by fusing a deep semantic analysis of the content with the discriminative priors obtained by refining the confidence signal from the Profiler. The final output is a ranked list L_q of the documents in C_q, sorted in descending order based on their DAVINCI scores, representing the most suitable citations for a given context.

5.2.2 Rethinking the Evaluation Protocol: An Inductive Setting

A central contribution of our work is to address a fundamental yet often overlooked limitation in the standard evaluation protocol for citation recommendation. Traditionally, models are evaluated in a transductive setting. In this setup, the corpus of candidate documents is often constructed from the union of training, validation and test sets, and also the unparsable documents. While this does not lead to direct data leakage (i.e., using test labels for training), it creates an artificial evaluation landscape. Specifically, the ground truth citation for a given test query itself may be another document within the test set. This means the system is evaluated on its ability to find connections within a static collection where the query documents themselves are pre-indexed and searchable which is a condition that never holds in a real-world application. To faithfully address this shortcoming, we define and adopt a rigorous inductive evaluation setting. The core principle of the inductive setting is to enforce a strict temporal separation between the evaluation query and the candidate corpus, mirroring the natural arrow of time in research.

Formally, let D_{eval} be an evaluation set (either the validation set, D_{val}, or the test set, D_{test}), and let C be the candidate corpus available for recommendation. The inductive setting imposes two critical constraints:

Dataset	Transductive				Inductive			
	# Contexts			# Papers	# Contexts			# Corpus
	Train	Val	Test		Train	Val	Test	
ACL-200	30,390	9,381	9,585	19,776	30,390	8,512	7,072	7,108
FTPR	9,363	492	6,814	4,837	9,363	472	5,918	3,313
RefSeer	3,521,582	124,911	126,593	624,957	3,521,582	117,724	105,411	580,059
arXiv	2,988,030	112,779	104,401	1,661,201	2,988,030	103,125	95,247	700,403
ArSyTa	8,030,837	124,188	124,189	474,341	8,030,837	123,515	122,989	412,127

Table 5.1: The impact of our rigorous inductive setting. Enforcing temporal consistency corrects the inflation in corpus and evaluation sets seen in standard benchmarks, resulting in a markedly smaller and more realistic set of documents for training and inference. 'FTPR': FullTextPeerRead.

1. **Disjoint Sets:** The set of evaluation documents and the candidate corpus must be strictly disjoint, as defined by:

$$D_{\text{eval}} \cap C = \emptyset \tag{5.1}$$

2. **Temporal Consistency:** For any query document $D_q \in D_{\text{eval}}$, the candidate corpus C must only contain documents published strictly before D_q, formalised as:

$$\forall D_q \in D_{\text{eval}}, \forall D_c \in C : \text{date}(D_c) < \text{date}(D_q) \tag{5.2}$$

This setup ensures that, at evaluation time, a model is tasked with recommending citations for a "newly authored" paper (D_q) using only the body of "existing" literature (C). By adopting this inductive protocol, we eliminate any artificial advantage gained from a pre-known test set and obtain a more realistic and reliable assessment of a model's true generalisation capabilities. All experiments and benchmarks presented in this paper are conducted under this stringent inductive setting to ensure a fair and meaningful comparison. We show the statistics for benchmark datasets in Table 5.1.

5.2.3 Profiler: A Non-Learnable First-Stage Retrieval

The first stage of our system is the Profiler, a novel retrieval module designed to overcome the computational bottlenecks inherent in current state-of-the-art citation recommendation systems. Its design philosophy is rooted in decoupling the expensive process of representation enrichment of documents from the online query task. A key technical merit of Profiler is that it is entirely a non-learnable module. It operates as a principled, static transformation of the citation network, making it exceptionally fast and scalable. The name 'Profiler' reflects its core function: to compute a rich *public profile* for every document. We posit that a paper's relevance is a function of both its intrinsic content and its perceived identity within the scholarly network, i.e., an identity shaped

by its citing papers and the contexts of those citations.

Profiled Document Representations: A Static Enrichment Process. The Profiler's first task is a one-off, offline pre-processing step: transforming the entire corpus into a *profiled citation network*. For every document $D_i \in C$, we begin by initialising its base vector representation, $v_i \in \mathbb{R}^{d_{\text{ENC}_1}}$, using a small pre-trained language model encoder, $\text{ENC}_1(\cdot)$. We use specter2_base (Singh et al., 2022) as encoder due to its better performance observed with citation networks Goyal et al. (2024). To construct the profile of D_i, we augment this base representation with signals from its inward ego network, $\mathcal{N}_{in}(D_i)$, which is the set of documents that cite D_i. For each citing document $D_j \in \mathcal{N}_{in}(D_i)$, we extract two distinct signals: the representation of the citing paper's content, v_j, and the representation of the specific citation context snippet, $v_{s_{ji}}$, in which the citation is made.

The final profiled representation, $\hat{v}_i$, for document D_i is a static fusion of these signals. It is formally defined as the sum of its base representation and the aggregated vectors from its citing neighbours:

$$\hat{v}_i = v_i + \frac{1}{|\mathcal{N}_{in}(D_i)|} \sum_{D_j \in \mathcal{N}_{in}(D_i)} (\alpha \cdot v_{s_{ji}} + \beta \cdot v_j) \tag{5.3}$$

Here, $\alpha \in [0, 1]$ and $\beta \in [0, 1]$ are non-learnable hyperparameters, where $\alpha + \beta = 1$. This inherently robust design formulation provides a crucial regularising effect. For a very recent paper with no citations ($|\mathcal{N}_{in}(D_i)| = 0$), the profiled representation naturally defaults to its base semantic vector, v_i, directly tackling the cold start problem in a principled and graceful manner. Thus, the retrieval stage becomes purely content-based for such papers. Importantly, this is not a special fallback mechanism but a natural consequence of our designed formulation. As inbound citations accumulate over time, the representation is progressively enriched through degree-normalised aggregation. Therefore, Profiler provides monotonic refinement: it never degrades below semantic retrieval performance and strictly improves representation quality as structural evidence becomes available. This design ensures robustness under the inductive protocol, where newly published papers may initially lack citation signals.

Concurrently, the averaging mechanism ensures that the profiles of highly cited papers are not unduly skewed, while effectively modelling papers from emerging fields with sparse citations and interdisciplinary work with diverse citation patterns. Therefore, highly cited papers are not inherently up-weighted; only the semantic content of their citing contexts contributes to enrichment. While Profiler leverages citation structure, it deliberately excludes citation counts, venue prestige, and temporal weighting, thus mitigating the popularity bias.

Profiler can be interpreted as a deterministic one-step ratio-composite feature propagation operator over the citation graph. Specifically, Eq. 5.3 performs degree-normalised aggregation over the inward ego network, analogous to a fixed-weight graph convolution without learned parameters. Unlike GCN-style propagation, it introduces no learned filters, no degree amplification, and no spectral assumptions, thereby avoiding over-smoothing and hub amplification. This interpretation situates Profiler within graph representation learning as a parameter-free, content-aware smoothing operator that

captures how a paper is semantically contextualised by its citing literature.

Query Formulation and Efficient Cosine Similarity Search. For an incoming query $q = (S_q, M_q)$, we formulate a composite query representation, v_q, using a similar curation strategy:

$$\begin{aligned} v_q &= \gamma \cdot \text{ENC}_1(S_q) + \delta \cdot \text{ENC}_1(M_q) \\ &= \gamma \cdot \text{ENC}_1(S_q) + \delta \cdot \text{ENC}_1(T_q \oplus A_q) \end{aligned} \tag{5.4}$$

where $\oplus$ denotes textual concatenation, and $\gamma \in [0, 1]$ and $\delta \in [0, 1]$ are non-learnable hyperparameters constrained by $\gamma + \delta = 1$. With the entire corpus of profiled vectors ($\hat{v}_i$) pre-computed and indexed, the retrieval stage is reduced to a remarkably efficient similarity search. We employ cosine similarity to score the relevance of each candidate document against the query:

$$\text{Score}(q, D_i) = \text{cosine}(v_q, \hat{v}_i) \tag{5.5}$$

The resulting similarity scores are not only used to rank the initial candidate list C_q but are also passed directly to DAVINCI as a valuable set of confidence scores, $\{s_i\}$.

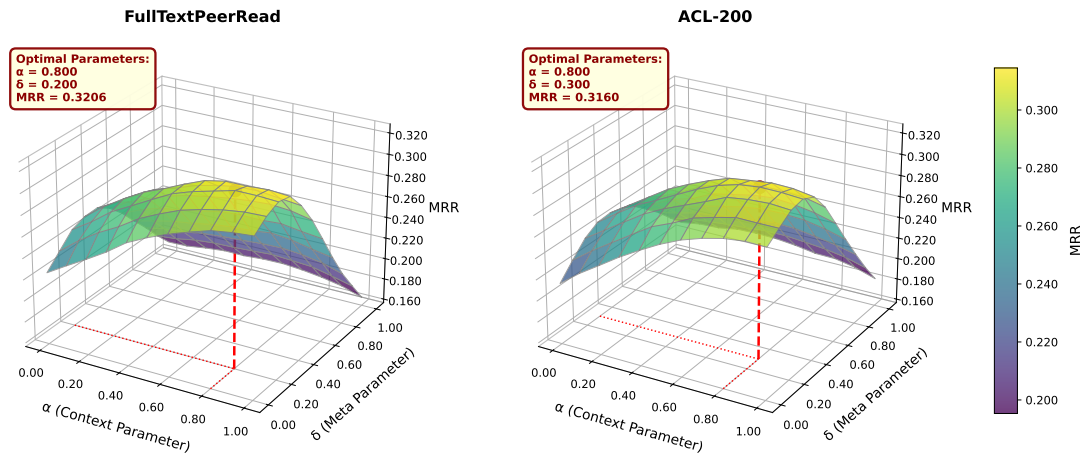
Hyperparameter Selection. The values for non-learnable hyperparameters ($\alpha, \beta, \gamma, \delta$) are determined empirically via a systematic sweep analysis on the validation sets of two of our smaller datasets. Crucially, as shown in Figure 5.4, the optimal set of values identified from this constrained analysis ($\alpha = 0.8$, $\delta = 0.3$) is then applied universally across all larger datasets without further tuning to ensure generalisation. Our analysis revealed that a specific ratio, i.e., the one that moderately prioritises the local context signal ($\alpha = 0.8$, $\gamma = 0.7$) over the global document topic ($\beta = 0.2$, $\delta = 0.3$) yields consistently strong performance. This finding underscores that the effectiveness of Profiler doesn't lie in dataset-specific tuning, but in its ability to capture a fundamental and generalisable structural property of scholarly networks.

5.2.4 The DAVINCI Reranking Architecture

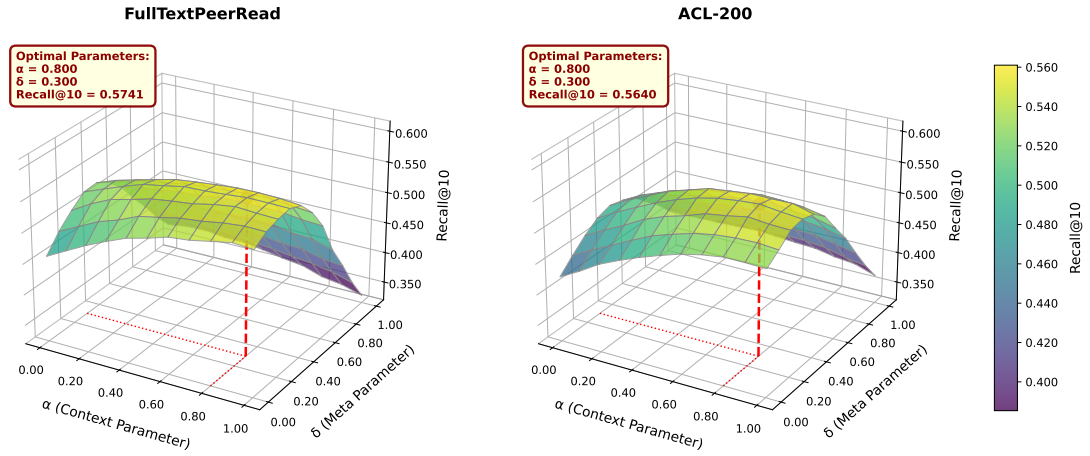
The efficacy of second-stage reranker is fundamentally constrained by its ability to enrich the semantic information of the query and the candidates obtained from the first-stage retrieval. We posit that state-of-the-art performance hinges not merely on the power of semantic encoding, but also on the confidence priors. Moreover, it depends on the sophistication of fusion mechanism that reconciles these modalities.

To this end, we introduce **DAVINCI (Discriminative & Adaptive Vector-gated Integration of Network Confidence & Information)**. It is founded on two core concepts: (i) a principled, non-linear transformation to refine the low information signal from the retrieval stage, and (ii) a novel fashion that creates a soft masking mechanism to achieve a dynamic and fine-grained fusion of signals. Finally, the reranker is optimised end-to-end using contrastive learning.

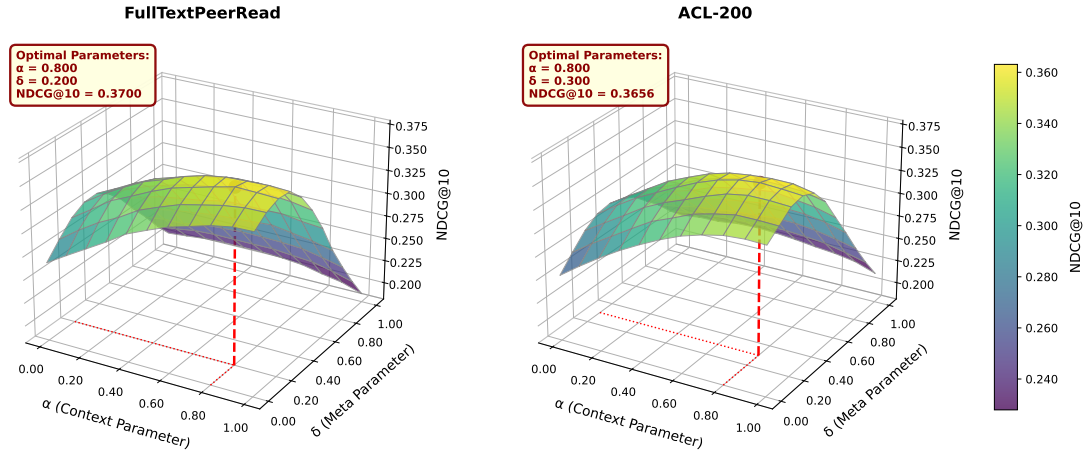
From Degenerate Scores to Discriminative Priors. A prerequisite for effective fusion is the availability of well-informed input signals. Raw cardinal scores from dense



(a) MRR Landscape



(b) Recall@10 Landscape



(c) NDCG@10 Landscape

Figure 5.4: Navigating the performance landscape of the public profile enrichment on the FullTextPeerRead and ACL-200 validation sets. Each plot shows a different evaluation metric: (a) MRR, (b) Recall@10, and (c) NDCG@10.

retrievers often exhibit severe score compression, providing a low-information signal with poor discriminative capacity. We therefore introduce a deterministic preprocessing block to transform this signal into a robust retrieval prior. (i) **Ordinal Abstraction**: We obtain a 1-indexed rank list $\{r_i\}$ from the list of cardinal scores $\{s_i\}$. For any ground-truth candidate not found in the profiler's output (e.g., an oracle-provided positive injected for training), we assign a default rank of $k+1$, where $k = |C_q|$. (ii) **Non-Linear Remapping**: The resulting integer ranks, while robust, are both linearly spaced and numerically large, and thus fails to capture the power-law distribution of relevance in ranked lists. These large integer values can be problematic for gradient-based optimisation, potentially leading to unstable training or exploding gradients. To address both issues simultaneously, we apply a non-linear exponential decay function to map the rank r_i to final transformed prior p_i:

$$p_i = \lambda^{r_i} \tag{5.6}$$

where $\lambda \in (0, 1)$ is a decay-rate hyperparameter, empirically set to 0.95. This transformation yields a geometrically spaced, continuous prior that more accurately models the steep non-linear decay of relevance probability. This transformed prior p_i serves as the definitive retrieval signal for all subsequent model components.

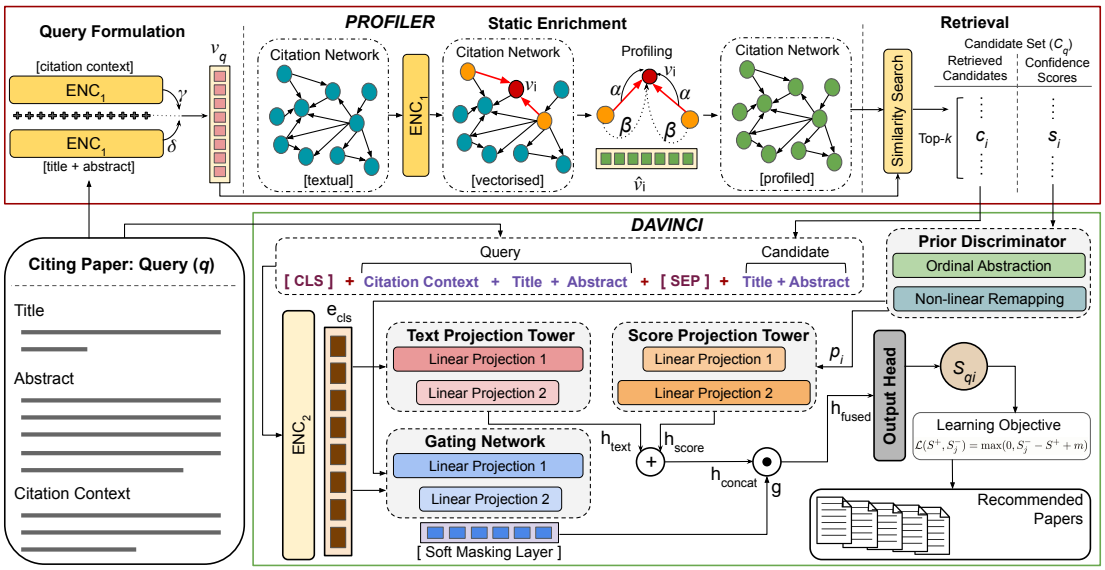


Figure 5.5: The architecture of our two-stage citation recommendation system. (1) The non-learnable **Profiler** performs a scalable retrieval by matching the query against a corpus of documents enriched with their *public profile*. (2) **DAVINCI** reranks the retrieved candidates using a vector-gated mechanism to integrate the discriminative retrieval priors with deep semantic features to produce a final ranked list of recommended papers for citation.

Adaptive Gated Fusion. The DAVINCI design is engineered to leverage the discriminative confidence prior p_i and fuse it intelligently with the raw semantic information. The semantic information is obtained by textually concatenating query text with the candidate text using a [SEP] token and encoding it using a small pretrained language model, $\text{ENC}_2(\cdot)$. We use SciBERT (Beltagy et al., 2019) as encoder due to its better

performance observed with non-graph fusion techniques (Goyal et al., 2024). We extract the [CLS] token's final hidden state, $e_{\text{cls}} \in \mathbb{R}^{d_{\text{ENC}_2}}$, as the raw semantic representation. To enable fusion, the heterogeneous inputs are first mapped into a common d_h-dimensional latent space via two independent Multi-Layer Perceptron (MLP) towers representing modality specific projection networks:

- **Text Projection Tower ($\mathbf{MLP_{text}}$):** Learns a non-linear mapping $f_{\text{text}} : \mathbb{R}^{d_{\text{ENC}_2}} \to \mathbb{R}^{d_h}$, yielding a task-specific text representation h_{text}.
- **Score Projection Tower ($\mathbf{MLP_{score}}$):** Learns a mapping $f_{\text{score}} : \mathbb{R} \to \mathbb{R}^{d_h}$, vectorising the scalar prior p_i into a dense score representation h_{score}.

To obtain the final processed semantic information, projected representations are concatenated as:

$$h_{\text{concat}} = [h_{\text{text}}; h_{\text{score}}] \in \mathbb{R}^{2d_h} \tag{5.7}$$

A separate **Gating Network**, $\mathbf{MLP_{gate}}$, then computes a vector-valued gate g. This network is conditioned on the original input signals (e_{cls} and p_i) to form an unbiased assessment of the raw evidence:

$$g = \sigma(\text{MLP}_{\text{gate}}([e_{\text{cls}}; p_i])) \in \mathbb{R}^{2d_h} \tag{5.8}$$

Here, σ is the element-wise sigmoid function, which constrains each element of the gating vector g to the range $(0, 1)$. Each element g_j can be interpreted as a learned throughput coefficient for the j-th feature. The final fusion is executed via the Hadamard product ($\odot$), which applies the gate g as a **per-dimension soft mask**:

$$h_{\text{fused}} = g \odot h_{\text{concat}} \tag{5.9}$$

This operation constitutes a form of *element-wise feature modulation*, providing a degree of representational flexibility unattainable with scalar fusion methods. The adaptively fused vector, h_{fused}, is passed to a dedicated **Output Head**, a final MLP ($\mathbf{MLP_{out}}$), which maps the $2d_h$-dimensional representation to a single logit. A final sigmoid activation produces the final reranked DAVINCI score $S_{qi} = f_{\text{DAVINCI}}(q, c_i, s_i)$ as shown below

$$S_{qi} = \sigma(\text{MLP}_{\text{out}}(h_{\text{fused}})) \in (0, 1) \tag{5.10}$$

This score S_{qi} represents the system's final confidence that candidate document D_i is a relevant citation for query q.

Learning Objective: Direct Optimisation of Ranking. To align model's training with its downstream evaluation, we use a loss function that directly optimises the relative ordering of candidates. The training process is structured around queries and their associated sets of k retrieved document candidates, which are labeled as positive (c^+) or negative (c^-) based on ground-truth relevance. To construct robust training instances and expose the model to a diverse set of negative signals, we adopt a negative sampling strategy. For a positive candidate c^+ associated with a query q, we compare it with randomly sampled n negative candidates, denoted as $\{c_1^-, c_2^-, \ldots, c_n^-\}$, from the pool of k retrieved candidates for the query. This process yields n distinct training triplets for a

positive example. For each triplet (q, c^+, c_j^-), the model computes the respective scores, S^+ and S_j^-. We then optimise the model using the margin-based triplet loss, applied individually to each pair:

$$\mathcal{L}(S^+, S_j^-) = \max(0, S_j^- - S^+ + m) \tag{5.11}$$

where $m \in (0, 1)$ is a margin hyperparameter. The total loss for a positive sample c^+ is the average sum of losses computed over these n sampled negatives: $\frac{1}{n} \sum_{j=1}^{n} \mathcal{L}(S^+, S_j^-)$. This objective function directly penalises incorrect rank-ordering across a varied subset of competitors, forcing the model to learn a scoring function that produces a well-separated ranking of candidates (please see Figure 5.5).

5.3 Experiments and Results

5.3.1 Implementation Details

This section details the experimental protocol designed to rigorously evaluate our proposed work. We conduct all experiments and benchmark under the realistic inductive setting. Our experimental pipeline is designed to reflect the distinct computational profiles of retrieval and reranking. The coarse-grained retrieval stages for all systems are executed on NVIDIA A100 DGX clusters. The more computationally intensive, fine-grained reranking stages utilise NVIDIA H200 DGX systems to ensure efficient processing. Given the substantial scale of the corpora and datasets, conducting multiple full training runs is computationally prohibitive. To ensure the robustness of our findings, we first perform a stability analysis. We conduct three training trials on representative subsets of the training data and observed minimal variance in performance, confirming the numerical stability of our training procedure. Consequently, the final results reported in all tables are from a single, comprehensive run on the full-scale datasets. The hyperparameters used for training DAVINVI are: Learning_rate = 1e-5; L2_weight = 0.01; Dropout = 0.3; Optimiser = AdamW; Adam_epsilon = 1e-6; Loss_margin = 0.1 and Batch_size = 16.

5.3.2 Results: First Stage Retrieval

We compare the results of our Profiler with the current state-of-the-art Prefetcher (Gu et al., 2022) and the sequential combination of Prefetcher followed by Enricher (Goyal et al., 2024). Prefetcher operates on a hierarchical attention based text encoding to obtain a list of first stage retrieved candidates for citation recommendation. Enricher ingests top 100 candidates from this prefetched list and models their symbiotic relationship embedded in the citation network to curate an enriched list of retrieved candidates, thus yielding a significantly higher Recall@300. In Table 5.2, results show that the non-learnable and scalable nature of Profiler makes it highly computationally efficient in reducing the retrieval time by 32.52x and 43.92x on the largest dataset (ArSyTa) and the smallest dataset (FullTextPeerRead), respectively. Results also show Profiler's merit to retrieve better candidates by increasing the MRR by 63.85% and 45.3% on ArSyTa and FullTextPeerRead, respectively.

Model	Comp. Time	MRR	Recall@K			NDCG@K		
			10	50	300	10	50	300
ACL-200								
Prefetcher	56.22m	21.14	40.33	65.37	86.98	24.57	30.11	33.30
Pref+Enr	64.43m	21.16	40.33	65.37	88.93	24.57	30.11	33.48
Profiler	**2.52m**	**30.17**	**53.79**	**74.63**	**89.58**	**34.88**	**39.57**	**41.78**
FullTextPeerRead								
Prefetcher	45.61m	21.73	39.17	63.43	87.16	24.78	30.15	33.63
Pref+Enr	49.20m	21.76	39.17	63.43	88.40	24.78	30.15	33.97
Profiler	**1.12m**	**31.62**	**57.23**	**82.05**	**96.27**	**36.62**	**42.23**	**44.35**
Refseer								
Prefetcher	99.17h	11.88	22.72	41.88	66.76	13.56	17.77	21.39
Pref+Enr	101.43h	11.92	22.72	41.88	69.91	13.56	17.77	21.88
Profiler	**3.10h**	**16.65**	**32.18**	**52.46**	**72.17**	**19.40**	**23.91**	**26.80**
arXiv								
Prefetcher	84.31h	13.78	27.09	48.83	74.16	15.94	20.73	24.43
Pref+Enr	85.94h	13.80	27.09	48.83	76.24	15.94	20.73	24.96
Profiler	**2.72h**	**16.61**	**33.41**	**55.95**	**76.61**	**19.56**	**24.57**	**27.61**
ArSyTa								
Prefetcher	225.88h	7.89	15.52	31.08	56.00	8.96	12.36	15.95
Pref+Enr	236.14h	7.94	15.52	31.08	66.59	8.96	12.36	17.31
Profiler	**7.26h**	**13.01**	**26.36**	**47.46**	**69.35**	**15.17**	**19.84**	**23.04**

Table 5.2: Performance comparison across all benchmark datasets for candidate retrieval. Our retrieval module (Profiler) consistently outperforms the SOTA baselines on all datasets across metrics and also with respect to the computational timing. Pref+Enr refers to sequential combination of Prefetcher followed by Enricher, leading to higher Recall@300 and NDCG@300 while keeping the same metric values for K=10 and K=50 as per its enrichment principle. Best values are highlighted in bold.

5.3.3 Results: End-to-End System

We evaluate our complete system with other standard baselines in Table 5.3 as detailed in our experimental setup. We outperform the SOTA citation recommendation systems and establish a new state-of-the-art on all datasets across all metrics. All the results are consistent with the inference reported in Goyal et al. (2024) with ArSyTa and arXiv

Model	MRR	Recall@K			NDCG@K		
		5	10	20	5	10	20
ACL-200							
BM25	10.53	15.45	20.82	26.71	10.71	12.44	13.92
SciNCL	15.41	21.39	30.04	39.76	15.01	17.79	20.24
HAtten	45.53	58.93	68.24	75.78	47.32	50.34	52.25
SymTax	46.98	60.20	69.47	76.83	48.73	51.75	53.62
Ours	**50.31**	**64.10**	**73.08**	**80.20**	**52.30**	**55.22**	**57.03**
FullTextPeerRead							
BM25	16.60	24.50	31.15	38.23	17.27	19.42	21.23
SciNCL	17.80	25.31	35.43	46.48	17.53	20.77	23.57
HAtten	55.03	68.60	75.58	80.62	57.33	59.60	60.88
SymTax	56.63	69.94	76.92	82.29	58.84	61.11	62.47
Ours	**59.68**	**74.41**	**82.17**	**87.42**	**62.16**	**64.68**	**66.02**
Refseer							
BM25	10.85	15.31	19.71	24.50	11.11	12.52	13.73
SciNCL	7.17	10.02	14.68	20.46	6.74	8.23	9.69
HAtten	30.64	39.41	45.78	51.41	32.01	33.72	34.98
SymTax	31.80	40.61	47.24	53.25	32.79	34.94	36.46
Ours	**32.57**	**42.19**	**49.52**	**56.37**	**33.62**	**36.00**	**37.73**
arXiv							
BM25	10.28	14.64	19.04	23.89	10.50	11.93	13.15
SciNCL	9.22	13.06	18.37	24.89	8.91	10.61	12.25
HAtten	28.13	37.01	45.06	52.32	28.86	31.36	32.37
SymTax	29.02	38.46	46.78	54.97	29.80	32.49	34.56
Ours	**30.46**	**40.86**	**49.89**	**58.50**	**31.38**	**34.31**	**36.49**
ArSyTa							
BM25	9.24	13.39	17.52	22.14	9.46	10.79	11.96
SciNCL	8.16	11.25	15.71	21.08	7.85	9.28	10.64
HAtten	19.92	27.70	34.90	42.25	20.50	22.83	24.69
SymTax	22.00	30.16	38.06	46.03	22.49	25.05	27.07
Ours	**24.01**	**33.74**	**42.83**	**51.56**	**24.73**	**27.67**	**29.89**

Table 5.3: Our end-to-end citation recommendation system (Ours) consistently outperforming all baselines. Best values are highlighted in bold.

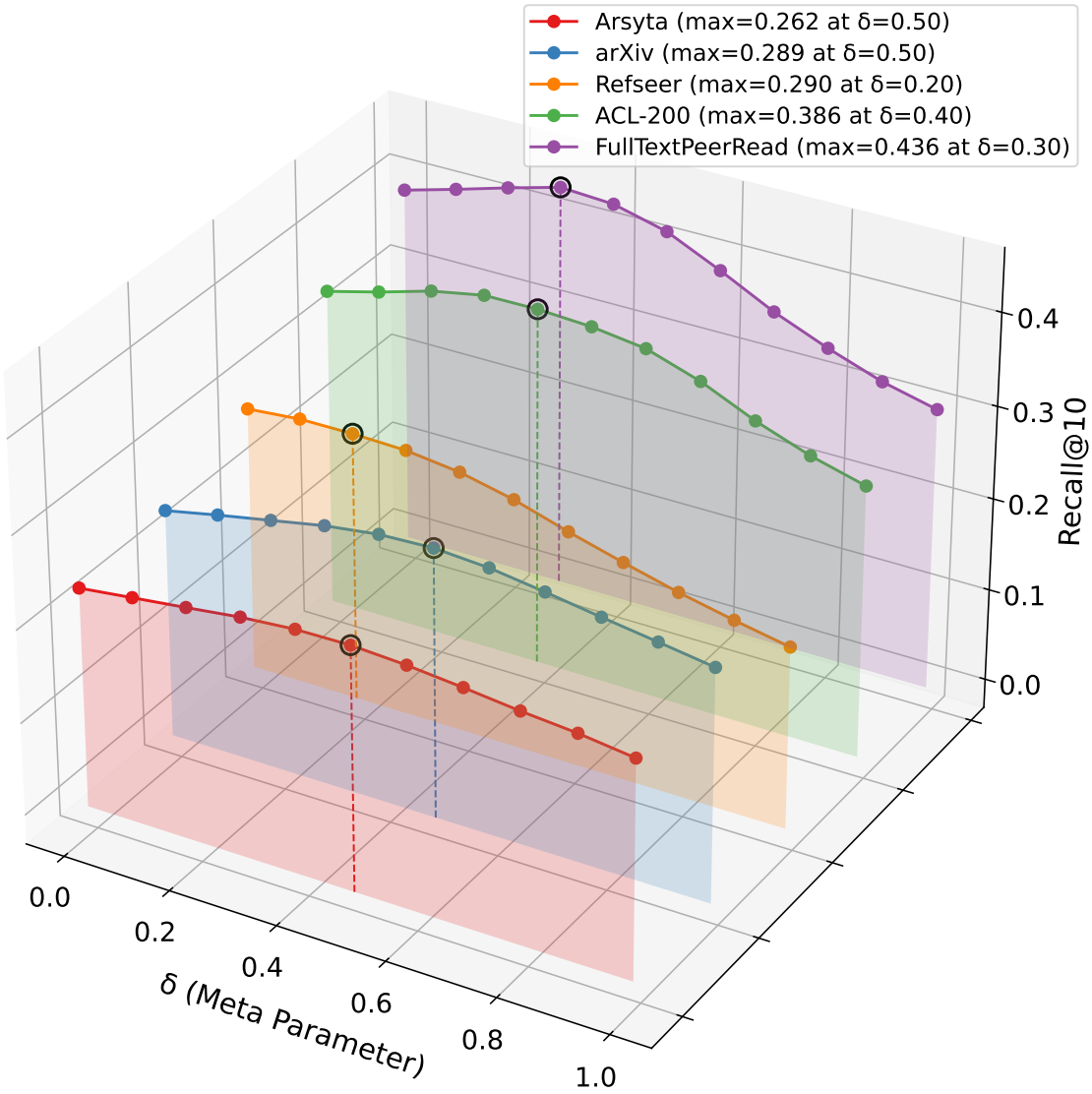


Figure 5.6: The performance variation for varied query composition when the profiling is disabled.

being the toughest benchmarks owing to their broad spectrum of scientific papers.

5.4 Analysis

To dissect the contributions of our core design choices, we conduct a series of targeted **ablation** studies on both the Profiler and the DAVINCI reranker. These analyses are designed to validate our architectural hypotheses and quantify the impact of each novel component. Additionally, we present both the **quantitative analysis** and the **qualitative analysis**.

	ACL-200	FTPR	Refseer	arXiv	ArSyTa
w/ Profiling	53.7	57.2	32.1	33.4	26.3
max attainable w/o Profiling	38.6	43.6	29.0	28.9	26.2

Table 5.4: Performance comparison w.r.t. Recall@10 on all datasets with profiling enabled versus the maximum attainable scores possible for any query composition when the profiling is disabled.

5.4.1 Ablation Analysis: Profiler

We perform two key analyses to validate the efficacy of the public profile concept and its implementation in the Profiler. In Figure 5.4, we visualise and navigate the landscape of public profile corresponding to FullTextPeerRead and ACL-200 datasets for MRR, Recall@10 and NDCG@10, clearly depicting the entire spectrum of public profile. To measure the performance gain enabled by profiling, we conduct an ablation where the profile enrichment is turned off (i.e., setting $\alpha = 0$ and $\beta = 0$ in Equation 5.3, so $\hat{v}_i = v_i$). As shown in Figure 5.6, we observe a significant degradation in retrieval performance for all datasets across varied query compositions (i.e., different γ, δ values). Moreover, we observe that large and tough datasets are relatively more robust to varied query compositions in this case. We also show the scores for Recall@10 on all datasets with profiling enabled versus the maximum attainable scores possible for any query composition when the profiling is disabled in Table 5.4. This directly confirms that profiling is not merely a hypothetical construct but a vital signal for effective first-stage retrieval.

5.4.2 Ablation Analysis: DAVINCI

To isolate the contribution of each component within DAVINCI, we conduct four ablation studies, systematically deconstructing the full model. The results for these ablations on the FullTextPeerRead and ACL-200 datasets are presented in Table 5.5, and are described as follows (1) **Semantics Only:** We discard the use of network confidence scores. This experiment is designed to quantify the value of integrating the Profiler's retrieval confidence into the reranking stage. (2) **Turned-off Discriminator:** We bypass our signal refining process (ordinal abstraction and exponential remapping) and instead feed the raw, untransformed confidence scores from the Profiler to testify the necessity of our proposed transformation for handling low-information retrieval signals. (3) **Softmax Normalisation:** We replace our discriminative transformation with a standard softmax function applied to the retrieval scores of the top-k candidates. This provides a direct comparison of our principled remapping scheme against a common baseline for score normalisation. (4) **Scalar Gating:** We replace the vector-gating mechanism with scalar gating of semantic information controlled by discriminative prior. This experiment directly measures the performance gain attributable to our fine-grained, per-dimension adaptive fusion policy.

Model	MRR	Recall@K			NDCG@K		
		5	10	20	5	10	20
ACL-200							
Ours	**50.31**	**64.10**	**73.08**	**80.20**	**52.30**	**55.22**	**57.03**
A1	48.42	62.42	71.32	78.38	50.44	53.34	55.13
A2	48.30	61.85	70.75	77.81	50.20	53.10	54.89
A3	49.46	62.67	71.83	78.49	51.27	54.26	55.96
A4	45.16	57.66	66.05	72.99	46.81	49.54	51.30
FullTextPeerRead							
Ours	**59.68**	**74.41**	**82.17**	**87.42**	**62.16**	**64.68**	**66.02**
A1	58.08	72.88	80.30	86.19	60.56	62.96	64.46
A2	58.20	72.78	80.20	86.26	60.61	63.03	64.56
A3	58.49	72.90	80.88	86.23	60.83	63.42	64.78
A4	53.58	68.04	75.90	82.22	55.90	58.46	60.08

Table 5.5: Ablation analysis showing the impact of our design choices w.r.t. **our** complete system, namely, **A1** (Semantics Only), **A2** (Turned-off Discriminator), **A3** (Softmax Normalisation), and **A4** (Scalar Gating). Best values are highlighted in bold.

5.4.3 Quantitative Analysis

To analyse the sensitivity of our reranker to the candidate pool size, we present a quantitative analysis varying the number of candidates (k) to be reranked. While the main experiments in this paper are conducted with k=300, Table 5.6 details the performance variation at different values of k. The results reveal two distinct trends: on smaller datasets, performance scales positively with k; however, on larger datasets, performance peaks around k=300 and subsequently degrades. This degradation suggests that processing too many low-quality candidates introduces noise that can harm the reranker's precision. Given that computational cost also grows linearly with k, this analysis confirms that k=300 represents an optimal trade-off, maximising performance while avoiding the dual penalties of increased noise and computational overhead.

5.4.4 Qualitative Analysis

To complement our quantitative results and provide deeper insight into the mechanisms driving our model's performance, we conduct a qualitative case study. By manually inspecting the recommendations for a representative query, we can better understand how our system compares with the state-of-the-art citation recommendation systems, as shown in Table 5.7. We select a query paper from our test set whose topic is nuanced and requires a deep understanding of the semantics. The SOTA models demonstrate a classic failure mode of relying on broad and superficial topic matching. They correctly

k	MRR	Recall@K			NDCG@K		
		5	10	20	5	10	20
ACL-200							
50	46.97	59.62	67.02	71.75	49.04	51.46	52.67
100	48.92	62.08	70.36	76.46	50.92	53.62	55.17
300	50.31	64.10	73.08	80.20	52.30	55.22	57.03
1000	**50.92**	**64.86**	**74.31**	**81.73**	**52.84**	**55.91**	**57.79**
FullTextPeerRead							
50	55.03	68.60	75.14	79.35	57.48	59.61	60.68
100	57.69	72.01	79.25	83.87	60.19	62.54	63.72
300	59.68	74.41	82.17	87.42	62.16	64.68	66.02
1000	**60.20**	**75.14**	**82.84**	**88.43**	**62.71**	**65.22**	**66.64**
Refseer							
50	28.78	37.47	43.61	48.73	29.96	31.95	33.25
100	30.60	39.77	46.55	52.66	31.70	33.90	35.45
300	32.57	**42.19**	**49.52**	**56.37**	33.62	**36.00**	**37.73**
1000	**32.74**	42.01	49.11	55.89	**33.68**	35.98	37.70
arXiv							
50	27.24	37.12	45.01	51.55	28.44	31.00	32.66
100	28.85	39.02	47.65	55.33	29.89	32.69	34.64
300	**30.46**	**40.86**	**49.89**	**58.50**	**31.38**	**34.31**	**36.49**
1000	30.06	39.82	48.62	56.97	30.81	33.66	35.78
ArSyTa							
50	21.51	30.52	37.73	43.67	22.60	24.94	26.45
100	22.96	32.47	40.60	47.84	23.93	26.57	28.40
300	**24.01**	**33.74**	**42.83**	**51.56**	**24.73**	**27.67**	**29.89**
1000	20.74	29.24	38.37	47.98	21.02	23.96	26.39

Table 5.6: Analysis showing the impact of number of candidates (**k**) on reranking performance. We found the value of 300 as an overall better choice for the final reranking performance with respect to the metrics and the computational overhead.

identify the general topic of 'Machine Translation' but completely misses the critical and specific usage of the term 'MERT'. Instead they focus on another term 'Moses' from both the citation context and the query abstract, and use these two signals to recommend

Citation Context:- "lation, phrases are extracted from this synthetic corpus and added as a separate phrase table to the combined system (CH1). The relative importance of this phrase table is estimated in standard MERT (TARGETCIT) . The final translation of the test set is produced by Moses (enriched with this additional phrase table) and additionally post-processed by Depfix. Note that all components of this combination have d"
Query Title:- What a Transfer-Based System Brings to the Combination with PBMT.
Query Abstract:-We present a thorough analysis of a combination of a statistical and a transferbased system for English→Czech translation, Moses and TectoMT. We describe several techniques for inspecting such a system combination which are based both on automatic and manual evaluation. While TectoMT often produces bad translations, Moses is still able to select the good parts of them. In many cases, TectoMT provides useful novel translations which are otherwise simply unavailable to the statistical component, despite the very large training data. Our analyses confirm the expected behaviour that TectoMT helps with preserving grammatical agreements and valency requirements, but that it also improves a very diverse set of other phenomena. Interestingly, including the outputs of the transfer-based system in the phrase-based search seems to have a positive effect on the search space. Overall, we find that the components of this combination are complementary and the final system produces significantly better translations than either component by itself.

#	HAtten recommendation	SymTax recommendation	Ours recommendation
1	Moses: Open Source Toolkit for Statistical Machine Translation	Moses: Open Source Toolkit for Statistical Machine Translation	**Minimum Error Rate Training in Statistical Machine Translation**
2	Combining Multi-Engine Translations with Moses	Findings of the 2012 Workshop on Statistical Machine Translation	Statistical Phrase-Based Translation
3	SMT and SPE Machine Translation Systems for WMT'09	A STATISTICAL APPROACH TO MACHINE TRANSLATION	Moses: Open Source Toolkit for Statistical Machine Translation
4	MANY: Open Source MT System Combination at WMT'10	Combining Multi-Engine Translations with Moses	Findings of the 2012 Workshop on Statistical Machine Translation
5	Edinburgh's Machine Translation Systems for European Language Pairs	Phrasetable Smoothing for Statistical Machine Translation	Improved Statistical Alignment Models
6	Toward Using Morphology in French-English Phrase-based SMT	**Minimum Error Rate Training in Statistical Machine Translation**	A STATISTICAL APPROACH TO MACHINE TRANSLATION
7	Parallel Implementations of Word Alignment Tool	Training Phrase Translation Models with Leaving-One-Out	Improvements in Phrase-Based Statistical Machine Translation
8	Improved Alignment Models for Statistical Machine Translation	SMT and SPE Machine Translation Systems for WMT'09	Combining Multi-Engine Translations with Moses
9	Investigations on Translation Model Adaptation Using Monolingual Data	Statistical Phrase-Based Translation	Hierarchical Phrase-Based Translation
10	A STATISTICAL APPROACH TO MACHINE TRANSLATION	MANY: Open Source MT System Combination at WMT'10	Phrasetable Smoothing for Statistical Machine Translation

Table 5.7: Case study of citation recommendations for a sample from the ACL-200 dataset. The table contrasts the top-10 predictions from SOTA baseline models against our system, with **ground-truth** citation highlighted in **bold** to illustrate our model's improved relevance. We can see that our model is successfully able to predict the correct citation by checking the abbrevation 'MERT' against the titles of the available candidates whereas the other systems just focus on the term ('Moses') in the abstract of the citing paper and the citation context, and use it for checking. # denotes the rank of the recommended citation.

from the candidate pool. On the other hand, our system also identify the same general topic of 'Machine Translation' but intelligently picking up the abbreviated term 'MERT' and using it effectively for recommending from the retrieved candidate set by comparing it with their titles.

5.5 Comparing Profiler with APPNP as a Static Propagation Model

Section 5.2.3 situates Profiler within the graph propagation literature as a parameter-free, single-hop, inward-only smoothing operator. To make this comparison concrete, we conduct an extensive empirical evaluation of APPNP (Gasteiger et al., 2019) as an alternative corpus-side propagation scheme. APPNP is run in a label-free setting using the same inward-only citation graph and the same edge filter as Profiler, isolating the difference to the propagation and its adjoining retrieval mechanism alone, and ensuring the inductive setting remains intact.

APPNP propagation. Starting from $Z^{(0)} = v_i$ (base SPECTER2 embeddings), APPNP propagates document representations across the citation graph through B iterative steps:

$$Z^{(k+1)} = (1-u)\,\hat{A}\,Z^{(k)} \;+\; u\,Z^{(0)}, \qquad k = 0, 1, \ldots, B-1 \tag{5.12}$$

where $\hat{A}$ is the row-normalised adjacency matrix built from inward-only citation edges, and $u \in (0, 1)$ is the teleport probability governing the balance between the graph-propagated signal and the original base embedding. The final propagated representations $Z^{(B)}$ serve as the corpus-side document vectors for retrieval via cosine similarity.

Query formulation. For the query, we use the simple average of the SPECTER2 encodings of the citation context and the concatenated title and abstract:

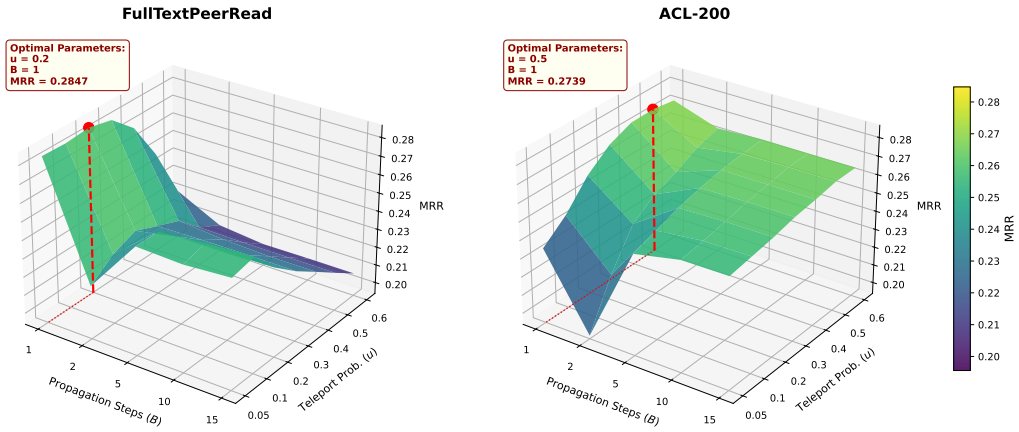
$$v_q = \frac{1}{2}\big(\,\mathrm{ENC}_1(S_q) + \mathrm{ENC}_1(T_q \oplus A_q)\,\big) \tag{5.13}$$

where S_q is the citation context snippet, T_q is the title, A_q is the abstract, and $\oplus$ denotes textual concatenation. This equal weighting is adopted deliberately to avoid introducing the additional query-side hyperparameters γ and δ of Eq. 5.4, which were designed specifically for Profiler.

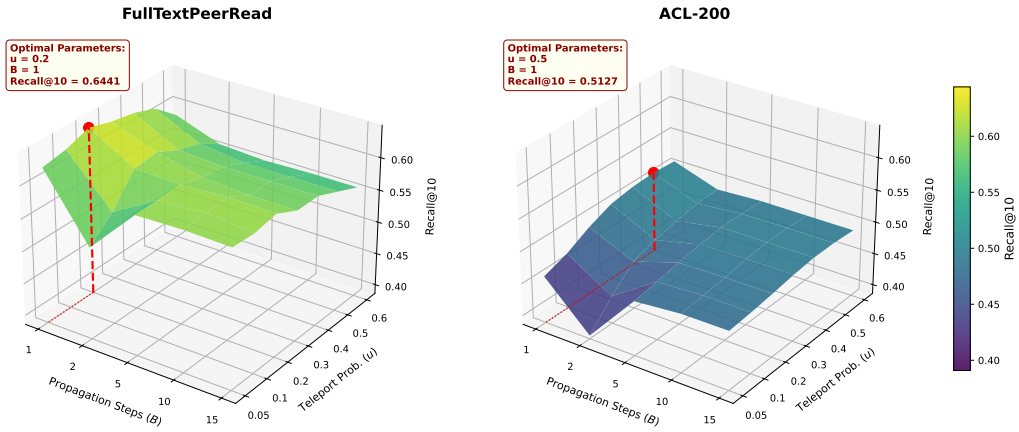
Hyperparameter landscape analysis. APPNP introduces two hyperparameters: the teleport probability u and the number of propagation steps B. We conduct a systematic grid search over $u \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$ and $B \in \{1, 2, 5, 10, 15\}$ on the validation sets of ACL-200 and FullTextPeerRead, evaluating MRR, Recall@10, and NDCG@10. The resulting performance landscapes are shown in Figure 5.7.

A key observation from these landscapes is that retrieval performance varies meaningfully across values of u even when $B = 1$. This is because the single propagation step produces, for each document D_i with at least one inward citation:

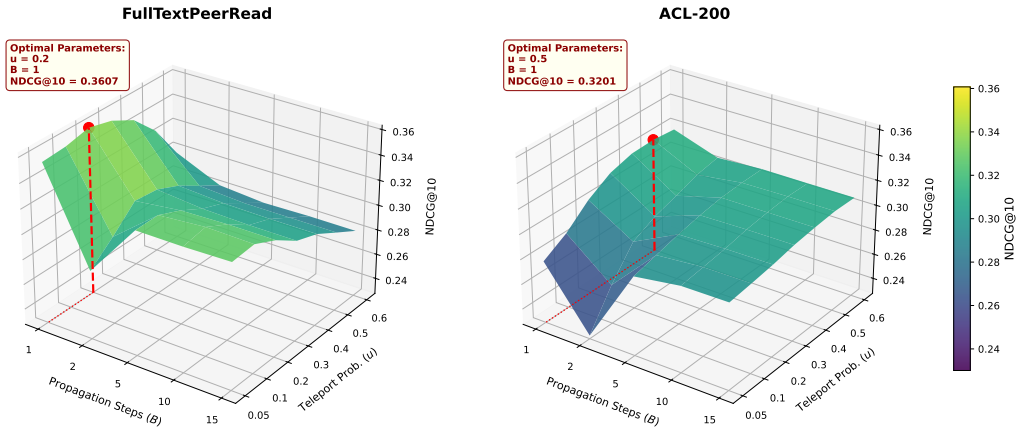
$$Z_i^{(1)} = (1-u)\,\frac{1}{|N_{in}(D_i)|}\sum_{D_j \in N_{in}(D_i)} v_j \;+\; u\,v_i \tag{5.14}$$



(a) MRR Landscape



(b) Recall@10 Landscape



(c) NDCG@10 Landscape

Figure 5.7: Navigating the performance landscape of APPNP propagation on the FullTextPeerRead and ACL-200 validation sets. Each plot shows a different evaluation metric: (a) MRR, (b) Recall@10, and (c) NDCG@10.

The aggregated neighbour term $\hat{A}v_i$, i.e., $Z_i^{(1)} - u\,v_i$, points in a different direction from the base embedding v_i, since citing papers carry their own semantic content that need not align with the cited paper's abstract. The result $Z_i^{(1)}$ is therefore a directional mixture of two non-parallel vectors, not a scalar multiple of v_i. Since cosine similarity is direction-sensitive, varying u changes the retrieval ranking even at $B = 1$. The only exception is nodes with $|N_{in}(D_i)| = 0$, for which $Z_i^{(1)} = u \cdot v_i$ reduces to a pure scaling, but these form a small minority in a citation corpus.

The consistent preference for $B = 1$ indicates that beyond a single propagation step, the accumulated neighbour signal begins to dilute the base SPECTER2 embedding, degrading retrieval. The optimal teleport probability differs across datasets: ACL-200 favours $u = 0.5$, placing equal weight on the base embedding and the propagated neighbour signal, while FullTextPeerRead favours $u = 0.2$, allowing a stronger contribution from the inward citation neighbourhood. This difference likely reflects the distinct citation density and structural properties of the two corpora in context of APPNP propagation scheme. We therefore select $\{u = 0.5, B = 1\}$ for ACL-200 and $\{u = 0.2, B = 1\}$ for FullTextPeerRead, and apply these two sets of values from the landscape analysis to the remaining three datasets.

Structural differences from Profiler. Despite the surface similarity, Profiler and APPNP with $B = 1$, differ in three important respects. First, and most critically, Profiler incorporates the citation context snippet embedding $v_{s_{ji}}$ from each inward neighbour (Eq. 5.3), capturing the specific intellectual purpose for which D_i was cited. With $\alpha = 0.8$, the dominant enrichment signal is not *who* cited a paper but *how* it was cited, i.e., the textual context in which it was invoked. APPNP propagates only node embeddings v_j with no awareness of the citation context, meaning this discriminative signal is entirely absent. Second, Profiler uses an additive formulation: the aggregated neighbour signal is added on top of the base embedding v_i, whereas APPNP produces a convex combination in which the base embedding and neighbour signal are mixed. Third, nodes with no inward citations, Profiler naturally returns $\hat{v}_i = v_i$ exactly since the sum is empty, while APPNP returns $u \cdot v_i$, i.e., a scaled version that is numerically different though equivalent under cosine similarity search.

Results. Table 5.8 reports the test-set performance of Profiler against APPNP with $\{u = 0.5, B = 1\}$ and $\{u = 0.2, B = 1\}$ across all five benchmark datasets on MRR, Recall@$\{10, 50, 300\}$ and NDCG@$\{10, 50, 300\}$. Profiler consistently outperforms both APPNP configurations across all datasets and metrics except for Recall@300. The performance gap is dominant for NDCG metrics showing that Profiler scheme retrieves better candidates and that too in a better ranking order. The convergence of APPNP's optimal configuration to $B = 1$ with high teleport probability opposed to the recommended values $\{u = 0.1, B = 10\}$ by Gasteiger et al. (2019), independently validates the design choices made for Profiler — the setting structurally closest to Profiler's single-hop inward design.

5.6 Comparing DAVINCI with Massive-Scale Rerankers

We conduct an experiment to answer a critical question: Can a compact, purpose-built reranker like DAVINCI outperform general-purpose reranking models with orders

Dataset	Model	MRR	R@10	R@50	R@300	N@10	N@50	N@300
ACL-200	APPNP (u=0.2, B=1)	25.21	47.51	71.27	89.52	29.44	34.78	37.47
	APPNP (u=0.5, B=1)	26.50	48.93	73.84	**91.78**	30.71	36.32	38.98
	Profiler	**30.17**	**53.79**	**74.63**	89.58	**34.88**	**39.57**	**41.78**
FullTextPeerRead	APPNP (u=0.2, B=1)	25.32	50.29	79.47	96.28	29.96	36.52	39.04
	APPNP (u=0.5, B=1)	26.75	51.33	79.37	**97.06**	31.35	37.64	40.29
	Profiler	**31.62**	**57.23**	**82.05**	96.27	**36.62**	**42.23**	**44.35**
Refseer	APPNP (u=0.2, B=1)	11.93	25.18	46.27	68.43	14.10	18.76	22.01
	APPNP (u=0.5, B=1)	12.87	26.65	48.61	72.05	15.12	19.98	23.41
	Profiler	**16.65**	**32.18**	**52.46**	**72.17**	**19.40**	**23.91**	**26.80**
arXiv	APPNP (u=0.2, B=1)	13.86	29.22	52.60	75.35	16.44	21.61	24.95
	APPNP (u=0.5, B=1)	14.96	30.61	54.20	**77.23**	17.60	22.82	26.20
	Profiler	**16.61**	**33.41**	**55.95**	76.61	**19.56**	**24.57**	**27.61**
ArSyTa	APPNP (u=0.2, B=1)	11.84	24.73	45.45	68.39	13.93	18.51	21.86
	APPNP (u=0.5, B=1)	12.47	25.13	45.22	67.98	14.54	18.99	22.31
	Profiler	**13.01**	**26.36**	**47.46**	**69.35**	**15.17**	**19.84**	**23.04**

Table 5.8: Performance comparison of Profiler against APPNP propagation variants across all five benchmark datasets. Best values are highlighted in bold. 'R' and 'N' represents Recall and NDCG respectively.

of magnitude more parameters that rely primarily on scale and broad pretraining? We evaluate against the current state-of-the-art reranking models, including the latest Qwen3-Reranker-8B (Zhang et al., 2025) and bge-reranker-v2-minicpm-40 having 2.72B parameters (Chen et al., 2024; Li et al., 2023). In contrast, our DAVINCI model is exceptionally lightweight, comprising only 110M parameters. To ensure a fair comparison, we standardise the retrieval stage for all models: each reranker is provided with the exact same list of candidate documents retrieved by our Profiler module, and we evaluate the performance on same test sets used for DAVINCI. Due to the immense size of these rerankers and their general-purpose pre-training, we employ instruction-aware prompting to adapt them to our specific task and datasets. Despite being up to *70x* smaller than the latest SOTA reranker, our specialised model markedly outperforms general-purpose models on all datasets, demonstrating the merit of task-specific design over raw parameter scale in an era of massive models, as shown in Table 5.9. Our findings demonstrate that performance gains arise from two complementary factors: (i) Task-specific architectural inductive bias, and (ii) Supervised adaptation to the citation recommendation objective.

5.6.1 Qwen Reranker Series

The Qwen model series, developed by Alibaba Cloud, represents a significant advancement in open-source language models. The rerankers from this series are specifically fine-tuned for relevance ranking tasks. Building upon the dense foundational models of the Qwen3 series, it provides a comprehensive range of reranking models in various sizes (0.6B, 4B, and 8B). This series inherits the exceptional multilingual capabilities, long-text understanding, and reasoning skills of its foundational model. The Qwen

Model	MRR	Recall@K			NDCG@K		
		5	10	20	5	10	20
ACL-200							
DAVINCI	**50.31**	**64.10**	**73.08**	**80.20**	**52.30**	**55.22**	**57.03**
Qwen3-R-8B	36.44	50.96	63.06	72.83	38.02	41.94	44.42
bge-R-v2-m-40	33.52	45.27	55.23	64.70	34.55	37.78	40.17
FullTextPeerRead							
DAVINCI	**59.68**	**74.41**	**82.17**	**87.42**	**62.16**	**64.68**	**66.02**
Qwen3-R-8B	48.15	66.84	77.62	85.62	51.08	54.60	56.63
bge-R-v2-m-40	41.22	53.71	63.87	73.16	42.44	45.75	48.11
Refseer							
DAVINCI	**32.57**	**42.19**	**49.52**	**56.37**	**33.62**	**36.00**	**37.73**
Qwen3-R-8B	24.81	35.39	44.98	54.04	25.67	28.79	31.09
bge-R-v2-m-40	22.10	30.27	38.39	46.58	22.53	25.15	27.22
arXiv							
DAVINCI	**30.46**	**40.86**	**49.89**	**58.50**	**31.38**	**34.31**	**36.49**
Qwen3-R-8B	25.48	36.02	47.19	57.35	26.10	29.72	32.30
bge-R-v2-m-40	21.70	29.79	38.10	46.87	21.99	24.68	26.89
ArSyTa							
DAVINCI	**24.01**	**33.74**	**42.83**	**51.56**	**24.73**	**27.67**	**29.89**
Qwen3-R-8B	22.39	32.44	40.71	49.33	23.26	25.95	28.13
bge-R-v2-m-40	17.79	24.70	31.73	38.79	18.03	20.31	22.08

Table 5.9: Performance of DAVINCI (110M) vs. massive-scale rerankers. 'R': Reranker; 'm': minicpm. Best values are highlighted in bold.

rerankers based on a powerful transformer architecture are trained on massive datasets of query-document pairs, learning to discern subtle relevance signals far beyond simple keyword matching. As instruction-tuned models, they operate as cross-encoders that expect a structured prompt. The model ingests the query and document by embedding them within a specific template that defines the task. This allows for deep, token-level interaction between the query and the document, conditioned on the explicit instruction. The model is trained to output a single logit, where a higher value indicates a higher probability of relevance. We select the Qwen reranker as it is widely regarded as a state-of-the-art, general-purpose reranker. Its strong performance across various public benchmarks makes it a formidable baseline to measure against. We use the latest and the

largest available open-source version, Qwen3-Reranker-8B having 8.19B parameters, from the Qwen series for our experiments.

5.6.2 BGE Reranker v2 (BAAI General Embedding)

The BGE model family, released by the Beijing Academy of Artificial Intelligence (BAAI), is another highly influential series of models optimised for text retrieval and ranking. The BGE-Reranker-v2 is particularly notable for its excellent performance and efficiency. The BGE reranker is also a cross-encoder based on a transformer architecture. It has been fine-tuned on a mixture of public and proprietary datasets specifically for relevance ranking. The model architecture, often based on efficient backbones like *minicpm*, is designed to deliver high performance without the prohibitive computational cost of the largest models. The *layerwise* aspect in some variants refers to advanced techniques that leverage representations from multiple transformer layers, which can enhance performance. The usage is identical to that of Qwen where it ingests a query, document pair and processes it through its transformer layers. It outputs a relevance logit, which is used to re-sort the candidates. BGE models are known for their strong performance on standardised retrieval benchmarks like the MTEB (Massive Text Embedding Benchmark). We employ the bge-reranker-v2-minicpm-layerwise having 40 layers and 2.72B parameters to provide another strong, publicly available baseline from a different lineage than Qwen. Its high ranking on public leaderboards and widespread adoption in the community make it an essential point of comparison for any new reranking model.

5.6.3 Implementation and Usage

To ensure a fair and direct comparison, we follow a consistent protocol for all baseline models. The pre-trained checkpoints for both the Qwen and BGE rerankers are loaded directly from the Hugging Face Hub. For each query-document pair, we use the specific instruction-based prompt formats recommended for Qwen and bge, respectively. For Qwen, an instruction, the query, and the candidate document text are combined into a single string template: ``<Instruct>: {instruction}\n<Query>: {query}\n<Document>: {doc}". For the {instruction} placeholder, we curate a clear task description as suggested in the Qwen guidelines: ``Given a citation context and citing paper information, determine if the candidate paper is relevant to be cited in this context". The sequences are truncated to the models' maximum input length. For bge, we follow its guidelines by choosing its recommended bge specific prompt: ``Given a query A and a passage B, determine whether the passage contains an answer to the query by providing a prediction of either 'Yes' or 'No'". We describe the query and candidate document construction using the functions as shown in Figure 5.8.

We run the models in inference mode on our same evaluation sets. For each formatted input, we extract the raw logit output before any final activation. This logit is used directly as the relevance score for reranking. To reiterate, the same set of retrieved candidates and the same evaluation metrics used for our own system are applied to these baselines to maintain experimental consistency.

```python
def create_query_from_citation(self, citation: Citation) -> str:
    """Create a query combining citation context and citing paper info"""
    query = f"Citation Context: {citation.context}\n\n"
    query += f"Citing Paper Title: {citation.citing_paper['title']}\n"
    query += f"Citing Paper Abstract: {citation.citing_paper['abstract']}"
    return query

def create_document_from_paper(self, paper: Dict[str, str]) -> str:
    """Create a document string from paper info"""
    doc = f"Paper ID: {paper['paper_id']}\n"
    doc += f"Title: {paper['title']}\n"
    doc += f"Abstract: {paper['abstract']}"
    return doc
```

Figure 5.8: Python functions for constructing the query and candidate document strings from the available raw data. The `create_query_from_citation` function combines the citation context with metadata from the citing paper, while `create_document_from_paper` formats the candidate paper's information.

5.7 Summary

This work presented a principled re-evaluation of the citation recommendation task, advancing the field on two fundamental fronts: the veracity of its benchmarks and the efficiency of its architectures. By instituting a rigorous inductive protocol, we first established a more faithful measure of real-world performance. Next, within this demanding framework, our proposed two-stage system, pairing a non-learnable retriever with a specialised gated reranker, set a new benchmark for both retrieval and end-to-end recommendation. The strong performance of our compact, 110M-parameter model against multi-billion parameter rerankers underscored a key finding: for specialised domains, architectural sophistication, task-aligned design choices and the integration of domain-specific knowledge are more salient drivers of success than just the raw parameter count.

Chapter 6

Prologue to Part II: Generalising to Graph-Level Augmentation for Node Classification

This second part of the thesis makes a conceptual leap, transcending the specific application and node-level focus of Part I to address a more general and fundamental challenge in Graph Representation Learning. It presents our third article documented in Chapter 7 by generalising the core principle of data enhancement from enriching node features to augmenting the graph's fundamental structure. We address the pervasive problem of underperforming GNNs on heterophilic graphs and show that our graph-level augmentation provides a significant and generalisable performance boost for node classification task. We provide the details of the article in its published form as below.

6.1 Details of Article 3

PathLens: Structurally Enhancing Heterophilic Graphs for GNNs Karan Goyal, Saankhya Samanta, Vikram Goyal, and Mukesh Mohania.
(Accepted at CIKM 2025 conference)

The notion that standard GNNs perform better on graphs with high homophily, led to the development of specialised algorithms for heterophilic datasets in recent years. In this work, we both examine and leverage this notion. Rather than creating new algorithms, we emphasise the importance of understanding and enriching the data. We introduce a novel data engineering technique, **PathLens**, that enhances the performance of both heterophilic and non-heterophilic GNNs on heterophilic datasets. Our method structurally augments a given heterophilic graph by adding supernodes, thereby creating a network of pathways connecting spectral clusters in the graph. It facilitates additional paths to bring similar nodes (intraclass) closer than dissimilar ones (interclass) by reducing the average shortest path lengths. We draw both intuitive and empirical connections between the relative decreases in intraclass and interclass average shortest path lengths and shifts in the graph's homophily levels, providing a novel perspective that extends beyond traditional homophily measures. We conduct extensive experiments on seven diverse heterophilic datasets using various GNN architectures and also compare with several data-centric techniques, demonstrating significant improvements as high as 37%

in node classification performance. Furthermore, our empirical findings highlight the strong sensitivity of several recent GNNs with respect to the random seed used for data splitting, underscoring this often-overlooked factor in GNN evaluation. The code will be available at https://github.com/goyalkaraniit/PathLens.

Chapter 7

Structural Augmentation for Enhancing Heterophilic Graphs

7.1 Introduction

Graphs can manifest the most rich form of data and are pervasively used in many real-world applications such as social networks (Tang et al., 2009; Chen et al., 2010; Qiu et al., 2018), citation networks (Gollapalli and Caragea, 2014), e-commerce (Baumann et al., 2018; Wang et al., 2019) and recommendation systems (Wu et al., 2022; Gao et al., 2023). The domain of Graph Representation Learning (GRL) has gained much traction in the last few years to model the rich information that graphs manifest and has established itself as a de-facto standard approach to solve several tasks such as graph classification, link prediction and node classification. A myriad of Graph Neural Network (GNN) algorithms have been proposed that mostly fall under the general notion of message passing, i.e., aggregation and updation (Kipf and Welling, 2017; Hamilton et al., 2017; Gilmer et al., 2017; Veličković et al., 2018; Xu et al., 2019; Abu-El-Haija et al., 2019; Pei et al., 2020; Yang et al., 2022). Message passing enables representation learning via iterative updates of a node's representation by aggregating the features of its neighbours. The selection of neighbours and aggregation operation has led to the designing of a wide variety of algorithms over the last few years.

In general, real-world networks/graphs fall into either of the two categories, i.e. homophilic or heterophilic, decided by a graph property called homophily. Homophily is the tendency to connect similar nodes via edge linkage, where class labels of the connected nodes generally govern the notion of similarity. For example, in citation networks, researchers often tend to cite research articles from the same domain (Ciotti et al., 2016). In contrast, low homophily, i.e., heterophily, is observed in heterophilic datasets, where edge formations do not favour similar class labels or actually favour dissimilar class labels. E.g. in social media platforms, people tend to form connections irrespective of gender, whereas, in dating networks, most people prefer to form connections with the opposite gender (Zhu et al., 2021).

A large number of GNN algorithms tend to perform better on homophilic graphs (Xu et al., 2018; Gasteiger et al., 2019; Wu et al., 2019; Deng et al., 2020a; Bojchevski et al., 2020; Huang et al., 2021; Brody et al., 2022) and are assumed to be not suitable for graphs with heterophily (Zhu et al., 2020, 2021; Wang et al., 2022a; He et al., 2022). This

assumption has led to the designing of specialised algorithms for heterophilic datasets. In the recent years, various algorithms have been proposed specifically for heterophilic datasets (Chen et al., 2020; Zhu et al., 2020; Chien et al., 2021; Jin et al., 2021; Lim et al., 2021; Bodnar et al., 2022; Li et al., 2022; Zheng et al., 2022). As highlighted by Platonov et al. (2023), these recently proposed heterophilic GNNs are evaluated on six heterophilic datasets used by Pei et al. (2020) wherein two datasets have a major drawback of train-test data leakage due to the presence of duplicate nodes. Recently Lim et al. (2021) released several new large-scale and diverse heterophilic datasets.

The research for specialised GNNs for heterophilic graphs has been driven by three primary factors (i) the notion that most GNNs perform better on homophilic graphs (as discussed above), (ii) in heterophilic graphs, vertices with high structural and label similarities are likely far away from each other (Suresh et al., 2021; Liu et al., 2024), and (iii) uniform neighbourhood aggregation and updation is oblivious to the distinction between similar and dissimilar neighbours (Xu et al., 2023). As discussed in Section 7.2 and Section 7.5, many specialised methods have attended to the above factors. This motivates us to make further advancement along these directions. In this work, we examine and leverage the above notion by increasing homophily of the heterophilic graphs, also shown empirically in Sec. 7.8.1. Intuitively, we enable information flow between different regions of the heterophilic graph by comparatively bringing vertices with similar labels closer to each other than the dissimilar ones (Sec. 7.8.2). To this end, we make the following contributions and discuss them in detail as the chapter follows:

1. We propose **PathLens** that offers a unique perspective (Sec. 7.8.2) of analysing graphs for GNN node classification through the **lens** of intraclass and interclass average shortest **path** lengths. To the best of our knowledge, we are the first to relate GNN performance with intraclass and interclass average shortest path lengths.

2. PathLens is a novel technique that **structurally enhances** a **heterophilic graph** with additional nodes and connections forming pathways over the original graph. These pathways enable better information exchange between different spectral regions of the heterophilic graph, boosting the performance of both heterophilic and non-heterophilic **GNNs** for node classification.

3. We correlate the performance of PathLens with homophily across heterophilic and homophilic datasets. We intuitively discuss and empirically show a generic correlation between the changes in graph homophily levels and the relative drop in intraclass and interclass average shortest path lengths.

4. PathLens achieves substantial improvement in node classification accuracy, with gains of up to 37%, highlighting its effectiveness across several diverse heterophilic datasets and GNN models.

5. Empirical findings of our exhaustive experimentation shed light on the high sensitivity of several recently proposed GNNs to the random seed used for data split.

7.2 Related Work

7.2.1 General GNNs

A wide variety of GNN models are based on the convolution principle which is defined as neighbourhood aggregation and updation. GCN (Kipf and Welling, 2017) aggregates the features of a node's neighbours by learning a weight matrix and uses them to update the node's feature vector. GraphSAGE (Hamilton et al., 2017) samples nodes from the 1-hop and 2-hop neighbourhood for aggregation. GAT (Veličković et al., 2018) uses an attention mechanism to give varied importance to various neighbours. Xu et al. (2018) introduced Jumping Knowledge networks to capture varied neighbourhood ranges for different nodes where subgraphs have diverse local structures. Wu et al. (2019) proposed a Simple Graph Convolution by successively dropping non-linearities and collapsing weight matrices between consecutive network layers, resulting in a linear classifier following a low pass filter. Gasteiger et al. (2019) explored the relationship between personalised PageRank and GCN to fast approximate the propagation of neural predictions. Liu et al. (2020) proposed DAGNN to decouple representation transformation and propagation in convolution operations. He et al. (2021) introduced BernNet to learn arbitrary graph spectral filters by an order-K Bernstein polynomial approximation. Brody et al. (2022) designed GATv2 to introduce dynamic attention by reversing the order of attention and non-linearity operations in GAT. Topping et al. (2022) studied bottleneck and over-squashing in message passing neural networks from a geometric perspective. Wang et al. (2023) proposed Allen-Cahn message passing, using interacting particle dynamics, where nodes are particles and edges represent attractive and repulsive forces between particles. Yang et al. (2023) introduced PMLP, which is identical to standard MLP in training but then adopts GNN's architecture in testing. AeroGNN (Lee et al., 2023) highlights vulnerability to over-smoothed features and smooth cumulative attention in attention-based GNNs. Bo et al. (2023) devised Specformer to encode the set of all eigenvalues and performs self-attention in the spectral domain, leading to a learnable set-to-set spectral filter. Huang et al. (2024) proposed UniFilter that integrated the heterophily basis with the homophily basis to construct a universal polynomial basis thus limiting over-smoothing and alleviating over-squashing.

7.2.2 Heterophilic GNNs

Pei et al. (2020) directed focus towards heterophilic datasets by introducing Geom-GCN that does bi-level aggregation over the structural neighbourhood obtained by mapping the original graph into a latent continuous space. Zhu et al. (2020) discussed the limitations of GNNs for learning under heterophily and proposed H2GCN. Zhu et al. (2021) proposed CPGNN to learn a class compatibility matrix to model graph homophily. Chien et al. (2021) proposed the use of Generalised PageRank (GPR) for GNN where GPR automatically learns to adjust weights in accordance with node label pattern. Lim et al. (2021) proposed LINKX, a simple technique of embedding adjacency matrix and node features separately through MLPs and combining them by concatenation. Fu et al. (2022) introduced p-Laplacian based GNN as an approximation of a polynomial graph filter over the spectral domain of p-Laplacians. Wang et al. (2022a) suggested an

adaptive propagation mechanism and aggregation process as per the homophily between node pairs based on attribute and topological information. Li et al. (2022) suggested two models, GloGNN and GLoGNN++, that capture node correlations by learning a coefficient matrix to guide the neighbourhood aggregation further. GBK_GNN (Du et al., 2022) suggested the use of bi-kernel feature transformation to capture homophily and heterophily followed by a selection gate over kernels for given node pairs. He et al. (2022) suggested block-guided classified aggregation to learn separate aggregation rules for neighbours of varied classes. Luan et al. (2022) proposed ACM to adaptively exploit aggregation, diversification and identity channels node-wisely to extract richer localised information for diverse node heterophily situations. Cavallo et al. (2023) proposed incorporating a learnable importance coefficient per layer to balance the contributions of neighbours and ego node. Zheng et al. (2023) proposed neural architecture search to build heterophilic GNN models automatically. Liao et al. (2023) decoupled the full-graph dependency from the iterative training and adopted an efficient precomputation algorithm for approximating multi-channel embeddings. Further, we discuss the recent data-centric methods that align with our direction of work in Section 7.5.

7.3 Proposed Technique

PathLens is a network of pathways that run over the top of regions formed by Spectral Clustering over a graph (please see Figure 7.1). Spectral Clustering uses connectivity information between data points to form clusters using eigenvalues and eigenvectors of the data matrix. Let $G = (V, E)$ be an undirected graph with vertex set $V = \{v_1, v_2, \ldots, v_n\}$ and edge set E. Let $W = (w_{ij})_{i,j=1,\ldots,n}$ be the weighted adjacency matrix of the graph G where w_{ij} represents edge weight between nodes v_i and v_j. If G is unweighted, then $w_{ij} = 1$ for an edge present between nodes v_i and v_j; otherwise $w_{ij} = 0$. We define degree matrix D as a diagonal matrix with degrees $d_1, \ldots, d_n$ on its diagonal where degree of a node $v_i \in V$ is

$$d_i = \sum_{j=1}^{n} w_{ij} \tag{7.1}$$

We then define the unnormalised graph Laplacian matrix as

$$L = D - W \tag{7.2}$$

We perform Spectral Clustering according to the procedure laid down by Shi and Malik (2000). Let K be the number of clusters that we want to construct in G. Then, we compute the first K generalised eigenvectors $u_1, \ldots, u_n$ of the generalised eigenproblem

$$Lu = \lambda Du \tag{7.3}$$

We then stack $u_1, \ldots, u_n$ as column vectors to construct $U \in \mathbb{R}^{n \times K}$. We then directly extract clusters from eigenvectors by cluster_qr method (Damle et al., 2019).

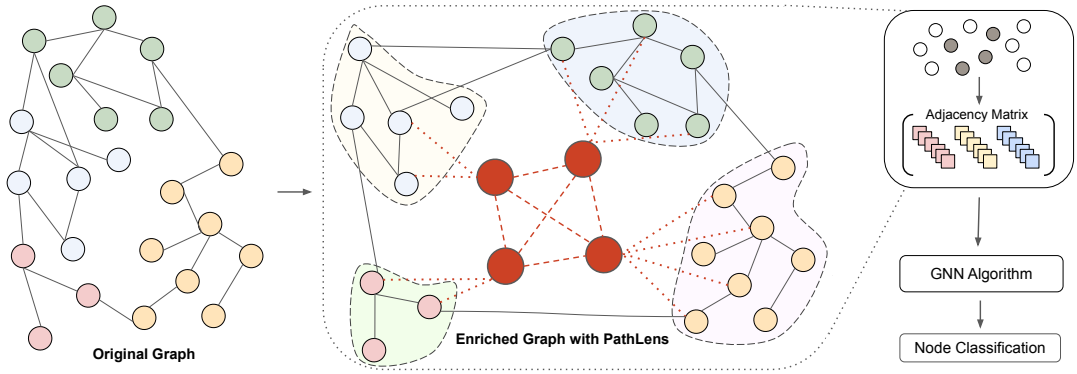


Figure 7.1: Overview of the use of PathLens. For a given heterophilic graph, we use PathLens to construct an enriched graph. We run available heterophilic or non-heterophilic GNN algorithm on the enriched graph for a downstream node classification task. In this representative enriched graph, the values of K, $mincon$ and $pcon$ are 4, 2 and 0.5 respectively. Colour of a node depicts the node belonging to a particular spectral cluster.

7.3.1 Structural Formation

Let $C = \{c_1, \ldots, c_K\}$ be the set of clusters obtained by Spectral Clustering where each such cluster represents a subgraph or a region formed corresponding to the graph topology. We construct pathways over the obtained spectral clusters to allow information exchange between different regions of the graph. We instantiate a new node called Spectral node for a cluster $c_i \in C \; \forall \; i \in \{1, \ldots, K\}$. We then connect these Spectral nodes among each other to form a network layer. To construct pathways, we need to connect the network of spectral nodes to the underlying graph. For each Spectral node s_i, we connect it to the corresponding spectral cluster c_i via a suitable connectivity principle. Instead of making random connections, we define the connectivity principle based on node importance. We make use of either of two popular node ranking algorithms to rank the node importance, $rtype$: {`PageRank`, `DivRank`}.

PageRank (Page, 1999) determines a node's importance by considering the incoming edges it receives from other important nodes in the graph. It outputs a probability distribution over the graph to represent the likelihood of a random surfer arriving at a particular node. We assume a uniform initial probability distribution at time step $t = 0$ such that the PageRank score of a node v_i is given by

$$R(i;0) = 1/n \tag{7.4}$$

Iteratively, at any time step t, PageRank score is

$$R(i;t+1) = \frac{1-d}{n} + d \sum_{j \in M(i)} \frac{R(j;t)}{L(j)} \tag{7.5}$$

where $M(i)$ represents the incoming neighbours of node v_i, $L(j)$ is the number of outgoing links of node v_j and d is the damping factor. The value of d is generally taken as 0.85 and corresponds to the probability that a random surfer continues to follow the outgoing links at node v_i.

PageRank relates to the prestige of the nodes in a graph, but diversity is another important property that we can account for ranking important nodes. DivRank (Mei et al., 2010) ranks nodes in a graph by setting up an interplay between prestige and diversity. Similar to PageRank, DivRank outputs a probability distribution over the graph, indicating the node ranking. We experimentally observed both PageRank and DivRank giving similar accuracy in our task.

We then connect a Spectral node s_i to a certain number of nodes in the spectral cluster c_i based on a percentage connectivity parameter $pcon$ consistent across all clusters. We choose percentage as the connectivity measure rather than a fixed integer because spectral clusters are of variable sizes. It ensures that we have a uniform extent of coverage across clusters. For small datasets, we can observe a few clusters that are small in size such that they end up having zero connections as per $pcon$. To account for this scenario, we introduce a $mincon$ parameter that ensures a minimum number of connections to be formed. Still, if cluster size is too small to accommodate the $mincon$, we do not connect to that cluster and drop the corresponding spectral node.

We have discussed the ranking algorithms and the connectivity coverage above for our connectivity principle. These offer us two new hyperparameter choices, namely $mode$ and $ctype$. We choose $mode$ as a hyperparameter to decide whether to run ranking on a cluster level or graph level, i.e., `local` or `global`. $ctype$ decides the type of nodes to choose for making connections. We explore four different ways to select from ranked nodes within a cluster: `low`, `mid`, `high` and `lmh`. Opting `low` enables connections to the nodes at the bottom of the ranked node spectrum. Similarly, `mid` and `high` lead to connection formation to the nodes in the middle and at the top of the ranked node spectrum, respectively. `lmh` enforces an equally distributed number of link formations with each of the low, mid and high ranked nodes. Since we design our method as generic to run variety of existing GNNs across diverse datasets, just like any neural network architecture, PathLens offer the following hyperparameters to tune mainly in the **concise range**:

- Number of spectral clusters, K: $\{30, 40, 50\}$
- Choice of ranking algorithm, $rtype$: {`PageRank`, `DivRank`}
- Percentage connectivity, $pcon$: $\{0.3, 0.4, 0.5\}$
- Minimum number of connections, $mincon$: $\{3\}$
- Mode of ranking, $mode$: {`local`, `global`}
- Connectivity type, $ctype$: {`low`, `mid`, `high`, `lmh`}

The above steps ensure structural formation of PathLens where nodes (not all) interact with other nodes (not all) in the farther regions of graph as well as in the same spectral cluster via a pathway of Spectral nodes, thus leading to an enhanced information flow.

7.3.2 Node Initialisation and Label Assignment

To initialise the embeddings of a Spectral node, we would not want to compute the average of the representations of nodes forming a connection with it, as this will lead to

oversmoothing (Xu et al., 2023). Hence, we initialise the embedding of a spectral node with a random sequence of zeroes and ones keeping the same embedding dimension as those of its neighbouring nodes.

To assign a class label to each Spectral node, we take the majority voting of class labels of nodes belonging to the cluster and assign it as the class label of the Spectral node. Since, a cluster node c_i^j can belong either to the training set, validation set or testing set, we take y as the true class label only for the training nodes. We assign pseudo labels to validation and testing nodes via modelling a probability distribution (P) over the induced training graph (G^{tr}) constituting only the training nodes and the corresponding edges.

Let P_{AB} denotes the probability of having an edge between two nodes with class labels A and B in G^{tr} respectively. Hence, for a validation/testing node c_i^j, we consider its 1-hop neighbours from the training set denoted by N_{tr}. Then the likelihood of assigning a pseudo class label ($l \in L_s$) is given by

$$\mathcal{L}(l) = \sum_{n_{tr} \in N_{tr}} P_{ly(n_{tr})} \tag{7.6}$$

where L_s is the set of node class labels in G. Thus we assign the pseudo class label with the maximum likelihood as

$$y = \arg\max_{l \in L_s} \mathcal{L}(l) \tag{7.7}$$

Hence, assigning a class label to Spectral node constitutes three operations:

- local label profiling captured by summation operator in $\arg\max_{l \in L_s} \mathcal{L}(l)$
- spectral label profiling captured by mathematical mode operation M over cluster node labels (see Eq. 7.13).
- global label profiling encapsulated in $P(G^{tr})$

To highlight the meaningfulness of various steps involved in the construction of PathLens, we present the results of several ablation studies in Sec. 7.6.3.

To summarise, we describe PathLens (PL) for a given input graph $G(V, E)$ as a **data engineering technique** outlined by the following process:

$$PL(G(V, E)) \Rightarrow G'(V', E') \equiv G'(V + S, E + E'' + E''') \tag{7.8}$$

where $S = \{s_1, \ldots, s_K\}$ is the set of Spectral nodes, E'' is the set of all possible connections formed amongst the Spectral nodes in the network layer and E''' is the set of connections formed by the spectral nodes with the original underlying graph G given by

$$E''' = \{N_e(s_1), \ldots, N_e(s_K)\}. \tag{7.9}$$

$N_e(s_i)$ represents the edge neighbourhood of s_i in the underlying graph G and is given by a function f as below

$$N_e(s_i) = f(mincon, pcon, mode, ctype, rtype, c_i, G) \tag{7.10}$$

and the number of connections formed by a spectral node with its spectral cluster of the original graph is calculated as

$$|N_e(s_i)| = max(mincon, [pcon * |c_i|]_+) \tag{7.11}$$

where $[x]_+$ represents greatest integer less than or equal to x.

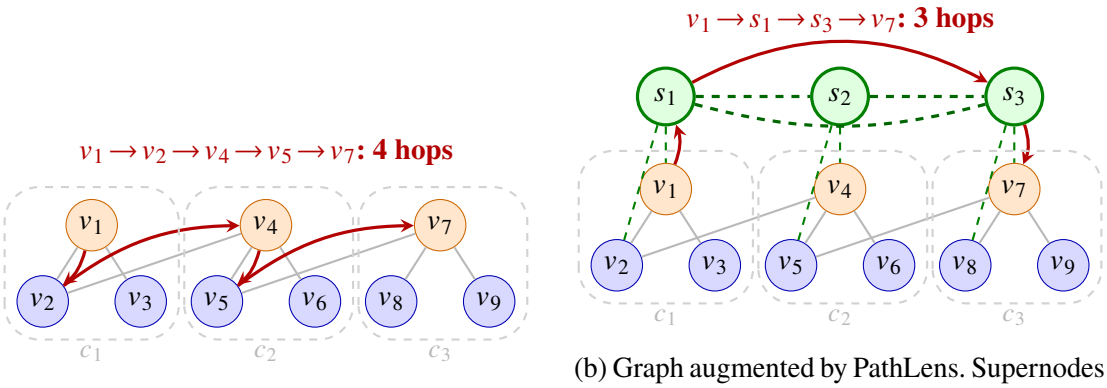
Also, the respective embedding and the class label of Spectral node s_i are as follows:

$$s_i = [rand\{0, 1\}]^d \tag{7.12}$$

$$y(s_i) = M[y(c_i^1), y(c_i^2), \ldots, y(c_i^{|N_e(s_i)|})] \tag{7.13}$$

where d is the dimension of node features, y is the class label, and M is mathematical mode operator.

7.3.3 A Worked Through Example



(a) Original heterophilic subgraph of Cornell. All hyperlinks are cross-class (student–faculty). The highlighted path shows the only route between student pages v_1 and v_7: 4 hops through faculty intermediates in bridging cluster c_2.

(b) Graph augmented by PathLens. Supernodes s_1, s_2, s_3 (green) are wired to the top-ranked nodes in each cluster and connected to each other via a fully connected supernode layer. The path $v_1 \to s_1 \to s_3 \to v_7$ now reaches student page v_7 in 3 hops, bypassing c_2 entirely.

Figure 7.2: PathLens illustrated on a representative subgraph of the Cornell dataset (WebKB). Nodes represent university webpages: orange = student pages (v_1, v_4, v_7); blue = faculty pages ($v_2, v_3, v_5, v_6, v_8, v_9$). All original hyperlinks are cross-class (student–faculty), reflecting the natural heterophily of academic web graphs. Green dashed edges are added by PathLens. The same-class pair (v_1, v_7) — two student pages in non-adjacent clusters — has its shortest path reduced from 4 hops to 3 hops via the supernode shortcut $v_1 \to s_1 \to s_3 \to v_7$, which bypasses the bridging cluster c_2 entirely. The different-class pair (v_1, v_5) remains at 3 hops in both graphs. This asymmetric reduction produces $ADR > 1$ and constitutes the structural mechanism of homophily injection discussed in Sec. 7.8.

To make the construction of supernodes and the resulting change in graph structure concrete, we walk through a representative example grounded in the Cornell dataset, on

which PathLens achieves among its strongest results (Section 7.4). Cornell is a WebKB webpage graph where nodes represent webpages from a university computer science department and edges represent hyperlinks between them. We consider two classes — *student* pages (orange) and *faculty* pages (blue) — and construct a representative nine-node subgraph that captures the essential heterophilic structure of this dataset.

Student pages predominantly link to faculty supervisors, course pages, and departmental staff, while faculty pages link back to student project pages, course listings, and other faculty. This cross-class linking pattern makes the graph naturally heterophilic: most hyperlinks connect a student page to a faculty page or vice versa. The representative nine-node subgraph in Figure 7.2 reflects this structure. Nodes v_1, v_4, and v_7 are student pages; nodes v_2, v_3, v_5, v_6, v_8, and v_9 are faculty pages. Every one of the eight original hyperlinks is cross-class (student–faculty): v_1–v_2, v_1–v_3 within cluster c_1; v_4–v_5, v_4–v_6 within c_2; v_7–v_8, v_7–v_9 within c_3; and two cross-cluster bridges v_2–v_4 and v_5–v_7, both also cross-class. There is no direct cross-cluster hyperlink between c_1 and c_3.

Supernode construction. PathLens applies spectral clustering to identify the three structural partitions c_1, c_2, c_3, and instantiates one supernode per cluster: s_1, s_2, s_3. Using the PageRank-based connectivity principle with $mincon = 2$, each supernode is wired to the two highest-ranked nodes in its cluster. The PageRank scores (proportional to degree in this small graph) are: $v_1 = 2$, $v_2 = 2$, $v_3 = 1$ in c_1; $v_4 = 3$, $v_5 = 2$, $v_6 = 1$ in c_2; $v_7 = 3$, $v_8 = 1$, $v_9 = 1$ in c_3. This gives:

$$s_1 \leftrightarrow \{v_1, v_2\}, \quad s_2 \leftrightarrow \{v_4, v_5\}, \quad s_3 \leftrightarrow \{v_7, v_8\}.$$

The three supernodes are mutually connected, forming a fully connected supernode layer with edges s_1–s_2, s_2–s_3, and s_1–s_3.

Change in graph structure and path lengths. Consider the same-class pair v_1 (student) and v_7 (student), located in non-adjacent clusters c_1 and c_3 with no direct hyperlink between them. In the original graph, the only route between these two student pages must pass through the bridging cluster c_2, traversing four heterophilic hyperlinks and passing through intermediate faculty pages:

$$v_1 \xrightarrow{\text{S–F}} v_2 \xrightarrow{\text{F–S}} v_4 \xrightarrow{\text{S–F}} v_5 \xrightarrow{\text{F–S}} v_7 \qquad \textbf{(4 hops)}$$

After PathLens, the supernode layer creates a direct structural shortcut between c_1 and c_3, bypassing c_2 entirely:

$$v_1 \longrightarrow s_1 \longrightarrow s_3 \longrightarrow v_7 \qquad \textbf{(3 hops)}$$

This path is valid because s_1 is directly wired to v_1, the supernodes s_1 and s_3 are directly connected within the supernode layer, and s_3 is directly wired to v_7.

By contrast, consider the different-class pair v_1 (student) and v_5 (faculty). The original graph already provides a 3-hop route via the cross-cluster bridge: $v_1 \to v_2 \to v_4 \to v_5$. The best supernode route is $v_1 \to s_1 \to s_2 \to v_5 = 3$ hops, offering no improvement. The same-class path therefore shortens while this different-class path does not, yielding $\nabla ASPL_{SC} > \nabla ASPL_{DC}$ and implying $ADR > 1$. This asymmetric reduction is the structural mechanism through which PathLens injects homophily into the Cornell graph, as discussed in Section 7.8.

7.4 Experiments

7.4.1 Experimental setup

We conduct extensive experimentation for node classification on a variety of heterophilic datasets using both heterophilic and non-heterophilic GNNs. As PathLens augments the existing heterophilic graph, its merit is determined by the performance of downstream GNNs. We take a heterophilic graph and use PathLens to generate an enhanced graph and then run an available GNN model on this enhanced graph to predict the class of a node. For a fair comparison, we only keep all the Spectral nodes in the train set and do not use them for validation or in the test set. This step ensures that there is no data leakage in our evaluation setting. We use different GNN hyperparameters for PathLens as the underlying graph is now modified. For each dataset, we consider 5 different random seeds (0, 5, 66, 244, 2020) for data split and run 3 rounds of experiments for each of the splits. Following Fu et al. (2022), we take 60/20/20 as the train/val/test split ratio. All the experiments are run for 100 epochs. We choose the commonly used accuracy as a metric and report its mean and standard deviation over the 15 runs. We run all experiments on 1 NVIDIA A100 80GB GPU. We use the official code repositories of the authors for implementing GPRGNN [1], pGNN [2], LINKX [3], DAGNN [4], BernNet [5], AeroGNN [6], DirGNN [7], PMLP [8], UniFilter [9], and Specformer [10]. For the rest of the baseline GNNs, we use the respective implementations in PyTorch Geometric provided by pGNN. We utilise scikit-learn (Pedregosa et al., 2011) implementation of Spectral Clustering.

Type	Dataset	# Nodes	# Edges	# Features
WebKB Webpage	Cornell	183	295	1703
	Texas	183	309	1703
	Wisconsin	251	499	1703
Actor Co-occurrence	Actor	7,600	33,544	931
Wikipedia Webpage	Chameleon filtered	890	8,904	2,325
	Squirrel filtered	2223	47,138	2,089
Citation Network	arXiv-Year	169,343	1,166,243	128

Table 7.1: Statistics of heterophilic datasets.

[1] https://github.com/jianhao2016/GPRGNN
[2] https://github.com/guoji-fu/pGNNs
[3] https://github.com/CUAI/Non-Homophily-Large-Scale
[4] https://github.com/divelab/DeeperGNN
[5] https://github.com/ivam-he/BernNet
[6] https://github.com/syleeheal/AERO-GNN
[7] https://github.com/emalgorithm/directed-graph-neural-network
[8] https://github.com/chr26195/PMLP
[9] https://github.com/kkhuang81/UniFilter
[10] https://github.com/DSL-Lab/Specformer

	Cornell	Texas	Wisconsin	Actor	ChameleonF	SquirrelF	arXiv	Avg (%↑)
MLP	84.34 ± 5.86	77.00 ± 12.98	94.81 ± 5.16	**43.51 ± 2.72**	54.13 ± 5.05	34.46 ± 10.48	39.48 ± 2.26	
PL	**87.17 ± 5.98**	**80.08 ± 6.31**	**95.12 ± 3.31**	40.81 ± 2.51	**57.22 ± 3.63**	**44.00 ± 10.18**	**40.11 ± 2.61**	5.20
GraphSAGE	**86.67 ± 4.10**	71.83 ± 7.97	89.01 ± 5.98	**40.46 ± 2.26**	56.94 ± 3.80	40.25 ± 8.54	**50.17 ± 0.60**	
PL	79.29 ± 5.36	**81.50 ± 6.11**	**93.02 ± 3.32**	38.67 ± 0.85	**58.33 ± 2.93**	**46.04 ± 7.64**	43.74 ± 1.58	1.29
GAT	45.96 ± 14.44	56.92 ± 20.98	64.94 ± 5.77	**34.51 ± 1.80**	**58.51 ± 2.74**	**42.65 ± 7.26**	21.81 ± 4.07	
PL	**47.68 ± 16.13**	**65.75 ± 12.58**	**71.42 ± 5.02**	33.18 ± 2.11	58.44 ± 4.23	41.31 ± 2.65	**41.34 ± 7.72**	15.95
APPNP	86.06 ± 6.12	**81.83 ± 5.09**	96.60 ± 1.47	**43.56 ± 3.70**	59.93 ± 2.64	38.53 ± 4.18	37.46 ± 6.24	
PL	**86.67 ± 7.13**	80.50 ± 5.86	**96.98 ± 2.58**	41.47 ± 1.91	**61.94 ± 2.16**	**42.20 ± 10.46**	**39.61 ± 2.72**	1.89
GPRGNN	82.02 ± 9.93	75.75 ± 12.29	92.96 ± 3.18	**41.80 ± 2.09**	60.52 ± 2.94	**45.91 ± 3.90**	21.58 ± 6.76	
PL	**82.73 ± 5.08**	**78.75 ± 7.92**	**94.14 ± 4.19**	39.53 ± 1.85	**60.97 ± 2.07**	38.90 ± 8.43	**37.95 ± 9.00**	8.85
LINKX	67.88 ± 14.22	62.42 ± 14.60	81.17 ± 9.27	33.88 ± 3.55	57.74 ± 2.98	43.14 ± 8.33	**52.94 ± 2.43**	
PL	**81.82 ± 7.58**	**78.67 ± 11.51**	**95.12 ± 3.00**	**35.62 ± 3.80**	**61.46 ± 3.91**	**47.44 ± 6.29**	45.40 ± 4.16	10.15
GATv2	39.49 ± 22.88	48.67 ± 28.00	65.06 ± 8.24	**33.27 ± 1.87**	57.60 ± 2.98	42.56 ± 6.15	24.86 ± 7.94	
PL	**48.38 ± 14.73**	**61.25 ± 9.89**	**73.21 ± 3.51**	32.10 ± 2.06	**58.89 ± 3.99**	**43.53 ± 3.95**	**44.84 ± 3.60**	20.32
pGNN	73.03 ± 10.41	68.83 ± 8.58	80.06 ± 6.87	33.79 ± 2.18	58.19 ± 3.84	**48.90 ± 3.58**	41.11 ± 0.75	
PL	**78.28 ± 9.73**	**81.00 ± 6.88**	**85.99 ± 3.62**	**35.15 ± 1.90**	**58.92 ± 3.37**	46.05 ± 4.00	**42.26 ± 1.32**	4.93
DAGNN	60.30 ± 14.15	55.00 ± 21.66	71.98 ± 4.78	34.10 ± 2.44	59.34 ± 3.26	**39.18 ± 5.87**	23.21 ± 9.28	
PL	**62.42 ± 7.27**	**68.17 ± 12.57**	**72.72 ± 3.05**	**34.36 ± 2.62**	**59.51 ± 3.12**	37.21 ± 8.84	**40.70 ± 9.39**	14.26
BernNet	83.74 ± 4.91	82.67 ± 4.35	93.52 ± 3.25	**38.80 ± 1.32**	58.89 ± 2.20	42.83 ± 3.08	22.88 ± 2.78	
PL	**86.77 ± 3.64**	**83.83 ± 6.17**	**97.04 ± 2.13**	37.63 ± 1.44	**61.74 ± 2.97**	**43.33 ± 7.16**	**47.09 ± 2.22**	16.79
AeroGNN	**52.12 ± 32.38**	35.67 ± 34.01	58.83 ± 12.26	29.38 ± 11.38	47.36 ± 9.96	39.70 ± 29.09	**29.92 ± 17.77**	
PL	47.27 ± 25.15	**42.83 ± 38.09**	**63.40 ± 19.66**	**32.39 ± 5.80**	**48.75 ± 9.28**	**50.14 ± 24.95**	29.36 ± 18.39	8.02
DirSAGE	71.41 ± 11.01	79.00 ± 11.03	92.41 ± 4.48	**38.80 ± 1.91**	57.15 ± 5.76	42.58 ± 7.64	42.43 ± 2.21	
PL	**76.97 ± 7.54**	**81.83 ± 11.00**	**93.40 ± 3.73**	38.30 ± 2.20	**59.27 ± 2.87**	**52.96 ± 4.40**	**44.45 ± 2.72**	6.28
PMLPGCN	41.01 ± 14.58	37.83 ± 26.40	55.62 ± 11.22	30.53 ± 6.36	**57.78 ± 2.79**	50.27 ± 5.15	34.56 ± 7.52	
PL	**46.87 ± 20.49**	**61.92 ± 15.75**	**68.58 ± 7.76**	**31.90 ± 5.00**	54.41 ± 2.71	**56.39 ± 7.44**	**37.42 ± 12.85**	17.19
PMLPAPPNP	29.80 ± 17.89	24.67 ± 24.66	52.84 ± 14.45	**31.86 ± 7.19**	**57.12 ± 4.38**	46.66 ± 9.71	34.41 ± 10.46	
PL	**48.79 ± 23.37**	**68.83 ± 13.67**	**63.15 ± 11.62**	31.83 ± 5.15	53.58 ± 4.96	**54.45 ± 8.04**	**34.88 ± 15.19**	37.70
UniFilter	27.47 ± 32.83	30.25 ± 37.25	**74.32 ± 26.48**	32.41 ± 10.30	41.81 ± 8.93	25.27 ± 16.69	22.69 ± 16.35	
PL	**53.64 ± 33.92**	**42.67 ± 42.14**	61.36 ± 31.48	**34.72 ± 5.69**	**45.07 ± 9.11**	**31.21 ± 18.21**	**29.33 ± 15.59**	26.65
Specformer	**55.56 ± 31.16**	37.00 ± 32.31	**75.56 ± 21.29**	30.31 ± 7.58	43.40 ± 17.10	**38.73 ± 24.08**	OOM	7.43
PL	55.35 ± 23.93	**53.08 ± 36.81**	70.31 ± 24.56	**35.80 ± 9.03**	**45.59 ± 18.57**	33.04 ± 16.99	OOM	

Table 7.2: Performance comparison of PathLens w.r.t. various models on seven heterophilic datasets. We report the accuracy values for GNN models and PathLens (PL). ChameleonF and SquirrelF represents the filtered versions of Chameleon and Squirrel datasets. arXiv denotes arXiv-Year dataset. We highlight global best result across GNNs for each dataset. Furthermore, we highlight best result for each combination of dataset and GNN. Last column reports the average accuracy jump across datasets observed for a baseline GNN. OOM represents Out Of Memory.

7.4.2 Baseline GNNs

We employ various neural architectures as baseline models and compare their respective performances with the use of PathLens. Hence, for exhaustive benchmarking, we choose: **Only node features** (MLP), **General GNNs** (GraphSAGE (Hamilton et al., 2017), GAT (Veličković et al., 2018), APPNP (Gasteiger et al., 2019), GATv2 (Brody et al., 2022), DAGNN (Liu et al., 2020), BernNet (He et al., 2021), AeroGNN (Lee et al., 2023), PMLP (Yang et al., 2023), Specformer (Bo et al., 2023), UniFilter (Huang et al., 2024)) and **Heterophilic GNNs** (GPRGNN (Chien et al., 2021), LINKX (Lim et al., 2021), pGNN (Fu et al., 2022), DirSAGE (Rossi et al., 2024)). Further, we consider two versions of PMLP based on GCN and APPNP.

7.4.3 Benchmark datasets

For benchmarking and evaluating the performance of our proposed technique, we choose seven datasets with varied statistics, as shown in Table 7.1. We choose **Cornell**, **Texas**, **Wisconsin**, and **Chameleon Filtered** heterophilic datasets for their small size; **Squirrel Filtered** and **Actor** datasets for their medium size; and **arXiv-Year** dataset for its large size. Cornell, Texas and Wisconsin are datasets of WebKB page data gathered from computer science departments of various universities. Lim et al. (2021) introduced arXiv-Year dataset with the task of predicting the year of publication or patent grant in citation graph. Squirrel and Chameleon datasets are introduced for node prediction by Pei et al. (2020) and have been extensively used for evaluating heterophilic GNNs. Recently, Platonov et al. (2023) identified the issue of node duplication in these datasets and released their corrected versions, namely Squirrel Filtered and Chameleon Filtered.

7.4.4 Results

Table 7.2 shows the performance of several models with and without applying PathLens (PL) on various heterophilic datasets. We see average accuracy improvements for a GNN across all datasets ranging from 1.29% - 37.7% as shown in the last column of Table 7.2. We observe that the Wisconsin dataset obtains the highest accuracy, whereas the Actor dataset proves to be the toughest to learn. The highest improvement in accuracy averaged over all GNNs is observed for the Texas dataset with a value of 30.06%. Also, we achieve the best performance across all models on 5 out of 7 datasets. Interestingly, the experimental results reveal that recently proposed GNNs like AeroGNN, PMLP, UniFilter and Specformer yield very high standard deviations in accuracy, clearly depicting that their performance largely depends on the random seed used for data splitting. Furthermore, we conduct three ablation studies dissecting PathLens in Sec. 7.6.3. Also, we analyse the time and space aspect of PathLens in Sec. 7.6.1 and Sec. 7.6.2 respectively.

7.5 Comparison with Data-centric/Rewiring Techniques

At present, two different lines of thought prevail in the GRL field. One set of work discusses the performance of GNNs regardless of the homophily levels (Luan et al.,

	Cornell	Texas	Wisconsin	Actor	ChamF	SquiF
WR-GCN	64.62 ± 6.76	77.81 ± 5.01	71.73 ± 5.79	34.64 ± 0.98	41.64 ± 3.30	35.06 ± 3.48
WR-GAT	64.44 ± 7.25	78.51 ± 6.12	76.53 ± 4.81	35.29 ± 1.05	40.00 ± 3.07	38.06 ± 2.07
ALT-APPNP	51.83 ± 7.12	58.62 ± 4.24	63.58 ± 5.30	34.13 ± 0.50	39.35 ± 2.03	35.66 ± 0.99
SIGMA	64.56 ± 8.07	78.07 ± 7.62	79.87 ± 6.44	32.93 ± 0.90	43.71 ± 3.21	40.10 ± 2.07
HalfHop	47.68 ± 24.13	34.83 ± 15.40	60.19 ± 12.68	25.24 ± 5.41	39.34 ± 10.64	39.17 ± 10.44
IPRMPNN	72.32 ± 2.37	78.26 ± 1.96	80.70 ± 0.85	36.31 ± 0.60	55.23 ± 1.63	42.10 ± 2.16
Dir-SAGE	71.41 ± 11.01	79.00 ± 11.03	92.41 ± 4.48	38.80 ± 1.91	57.15 ± 5.76	42.58 ± 7.64
ACMGCN++	44.75 ± 27.75	44.50 ± 39.23	66.98 ± 25.33	27.98 ± 12.78	50.69 ± 13.97	32.56 ± 22.59
ACMIIGCN++	50.00 ± 30.84	41.33 ± 38.87	67.47 ± 25.08	28.99 ± 12.72	47.85 ± 17.23	30.10 ± 19.09
PL (DirSAGE)	76.97 ± 7.54	**81.83 ± 11.00**	93.40 ± 3.73	**38.30 ± 2.20**	59.27 ± 2.87	52.96 ± 4.40
PL (LINKX)	**81.82 ± 7.58**	78.67 ± 11.51	**95.12 ± 3.00**	35.62 ± 3.80	**61.46 ± 3.91**	**47.44 ± 6.29**
PL (BernNet)	86.77 ± 3.64	83.83 ± 6.17	97.04 ± 2.13	37.63 ± 1.44	61.74 ± 2.97	43.33 ± 7.16

Table 7.3: Comparison of PathLens (PL) with other data-centric/rewiring techniques. We highlight the best result and the second best for each dataset, respectively, clearly showing PL performing substantially better than the compared methods.

2023), or the idea of good homophily and bad homophily (Ma et al., 2022), or the heterophily not always being harmful to GNN's performance (Luan et al., 2022). The other set of work shows that GNN's performance is indeed proportional to the homophily (Rossi et al., 2024; Liu et al., 2024; Xu et al., 2023; Suresh et al., 2021). DirGNN (Rossi et al., 2024) showed that treating graphs as directed improves learning on heterophilic graphs and attributed it to the increase in homophily. SIGMA (Liu et al., 2024) used SimRank (Jeh and Widom, 2002) as an aggregator to establish distinct relationships between similar nodes even when they are not connected and bypassing dissimilar nodes in the local neighbourhood. ALT (Xu et al., 2023) presented a data-centric solution by decomposing the original graph into two modified graphs and using a mixture of complementary filters. WRGNN (Suresh et al., 2021) transformed the input graph, keeping the same number of nodes, into a computation graph containing proximity and structural information as distinct types of edges. They showed that this obtained multi-relational graph possessed an enhanced level of assortativity. The above-discussed methods modify the original graph and use an existing GNN for prediction but not from the principle of inserting super nodes like PathLens. Specifically, PathLens is a complete structural augmentation technique, whereas the above techniques are only data-centric. Azabou et al. (2023) introduced HalfHop that upsamples edges in the original graph by adding "slow nodes" at each edge that can mediate communication between a source and a target node. Qian et al. (2024) proposed IPRMPNN which integrate implicit probabilistic graph rewiring into MPNNs to alleviate the under-reaching and over-squashing issues. We then empirically compare our method with the above discussed methods. We also show a comparison with Luan et al. (2022) that discusses heterophily not always being harmful and proposes Adaptive Channel Mixing (ACM) and a measure called Aggregated Similarity/Homophily.

We consider GCN and GAT variants of WRGNN, APPNP variant of ALT, SAGE for Dir-GNN, and ACMGCN++ and ACMIIGCN++ variants of ACM. We showed the

	Cornell	Texas	Wisc	Actor	ChamF	SquiF	arXiv
LINKX	0.290 (P)	0.352 (P)	0.335 (P)	44.637 (D)	0.735 (P)	2.402 (P)	13360.130 (D)
BernNet	0.326 (P)	0.292 (P)	0.647 (D)	4.998 (P)	0.699 (P)	18.016 (D)	22328.283 (D)
DirSAGE	0.275 (P)	0.341 (P)	0.643 (D)	5.429 (P)	0.752 (P)	2.256 (P)	3832.083 (P)

Table 7.4: Graph construction time for downstream node prediction task using different GNNs. The numerical values represent the time in seconds. 'P' represents PageRank and 'D' represents DivRank.

results for PathLens with GNN variants of DirSAGE, LINKX, and BernNet. We could not report the results on the arXiv-Year dataset as it led to out-of-memory (OOM) for many of the compared methods. The results in Table 7.3 show that PathLens performs best on all datasets except Actor, which is just second to DirSAGE. Interestingly, we also observe that ACM yields high standard deviations in predictive performance, just like AeroGNN, PMLP, UniFilter and Specformer. To reiterate, we separately construct Table 7.3 to compare different rewiring models. The results in Table 7.3 shows the baseline GNNs chosen by the authors of the proposed data centric/ rewiring methods. For example, WRGNN choose only GCN and GAT in their proposed work. We do not create any separate artificial setups that do not belong to the original proposed works.

7.6 Computational and Ablation Analysis

7.6.1 Time Analysis

In this section, we analyse the time aspect of PathLens in two different aspects. (1) Analyse the model runtime on original graph as well as on the augmented graph, and (2) Analyse the construction time of augmented graph which includes the time taken for (i) Spectral Clustering, (ii) running ranking algorithm, (iii) forming connections between spectral nodes and the original graph and also amongst themselves, and (iv) label and feature assignment of spectral nodes utilising likelihood estimation. We show the results for downstream node prediction task in Table 7.5 for three GNNs to better analyse the variability in the construction time as different hyperparameters lead to different augmented graphs. We then show the results for graph construction time in Table 7.4 that also highlights the effect of chosen ranking algorithm. As we can see DivRank considerably takes higher runtime than PageRank. As performance of both DivRank and PageRank are equivalent in our experimentation, we thus prefer PageRank for making connections with spectral nodes.

7.6.2 Space Analysis

We follow the setting discussed above in Section 7.6.1 uncovering the number of nodes and edges that are added through construction of PathLens by comparing their count in the original graph and the augmented graph in Table 7.6.

		Cornell	Texas	Wisc	Actor	ChamF	SquiF	arXiv
LINKX	G	0.921	0.908	0.914	0.961	0.915	1.059	1.612
	PL	1.036	1.017	1.007	1.058	1.022	1.075	1.676
BernNet	G	2.554	2.662	2.658	2.537	2.19	2.343	6.146
	PL	3.048	3.126	3.018	2.691	2.37	2.546	7.224
DirSAGE	G	0.902	0.915	0.967	0.95	0.997	1.613	2.746
	PL	1.146	1.032	1.132	1.14	1.137	1.773	3.778

Table 7.5: Model runtime comparison for downstream node prediction task. The numerical values represent the time in seconds. values 'G' represents original graph and 'PL' represents augmented graph after PathLens.

	Cornell		Texas		Wisconsin		Actor		ChamF		SquirrelF		arXiv	
	V	E	V	E	V	E	V	E	V	E	V	E	V	E
G	183	295	183	309	251	499	7600	33544	890	8904	2223	47138	169343	1166243
PL (LINKX)	204	554	211	764	276	893	7640	37306	924	9911	2263	49026	169373	1242898
PL (BernNet)	213	808	210	727	287	1206	7640	36547	918	9567	2253	48680	169373	1242898
PL (DirSAGE)	204	554	205	610	286	1170	7630	36204	918	9648	2253	48680	169373	1242899

Table 7.6: Space analysis for downstream node prediction task using different GNNs. 'V' and 'E' represents the total number of vertices and edges in the graph respectively. 'G' represents the original graph and 'PL' represents augmented graph after PathLens.

	Cornell	Texas	Wisc	Actor	ChamF	SquiF	arXiv
A1	30.97	35.01	13.03	3.55	5.47	11.26	5.12
A2	5.78	17.57	6.53	5.45	6.04	6.85	2.48
A3	7.14	13.50	3.81	1.62	4.66	9.51	2.07

Table 7.7: Ablation analysis showing the impact of switching off Spectral Clustering (A1), probabilistic modeling (A2), and connections between spectral nodes (A3) respectively. The table shows the percentage difference for each of these ablations w.r.t. our complete approach. The reported numerical values are the average drops (%) observed in accuracy across all the discussed GNNs.

7.6.3 Ablation Study

To show the importance of various steps involved in the construction of PathLens, we conduct three ablation studies as follows. (1) We switch off Spectral Clustering and randomly connect the super nodes to the original graph. (2) We use random labeling for super nodes instead of our designed probabilistic modeling. (3) We do not form connections amongst super nodes. We show the percentage difference for each of these ablations w.r.t. our complete approach in Table 7.7. The reported numerical values are the average drops observed in accuracy across all the discussed GNNs as chosen in Table 7.2.

7.7 Analysis of Design Choices

Apart from the above conducted ablation study, and to isolate and empirically validate the contribution of each architectural component within PathLens, we conduct an exhaustive analysis of our design choices. Specifically, we deconstruct the framework and substitute our proposed mechanisms with several standard alternatives available in the graph learning and clustering literature. The results for all these comparative experiments are aggregated and presented in Tables 7.8, 7.9 and 7.10. We categorise our design choices into four fundamental operations:

1. Choice of Clustering Algorithm PathLens inherently relies on Spectral Clustering to partition the graph into structurally meaningful subgraphs based on the graph Laplacian. To validate this choice, we replace Spectral Clustering with a wide array of alternative clustering techniques: distance-based methods (*K-Means, Bisecting K-Means, Birch*), density-based methods (*DBSCAN, HDBSCAN, OPTICS, Mean Shift*), hierarchical methods (*Agglomerative Clustering*), distribution-based methods (*Gaussian Mixture Models*), and message-passing/modularity methods (*Affinity Propagation, Louvain*). Empirical results demonstrate that Spectral Clustering outperforms these alternatives. The fundamental reason for this superiority lies in the unique topological challenges of heterophilic graphs. Distance-based and density-based algorithms typically operate under the assumption of a Euclidean space, attempting to group nodes by looking for tightly knit, spatial clumps. However, in sparse, heterophilic graphs, local neighbourhoods are highly deceptive; nodes predominantly connect to dissimilar neighbours, meaning nodes of the same class are often scattered multiple hops away rather than forming dense local communities. Consequently, traditional density and distance-based methods struggle to find meaningful patterns, often classifying the scattered nodes as noise or erroneously merging distinct structural regions into a single giant cluster. Spectral clustering overcomes this limitation by leveraging the eigenvalues and eigenvectors of the graph Laplacian. Instead of relying on raw local edges, it mathematically unfolds the complex, non-Euclidean topology of the graph into a lower-dimensional space. In this spectral space, nodes are clustered together not because they share a direct edge, but because they exhibit similar global connectivity patterns and share equivalent structural roles within the network's overall information flow. By bypassing the misleading local edges inherent to heterophily and focusing on the global macro-structure, Spectral

Clustering provides PathLens with the most accurate structural foundation for constructing effective long-range pathways.

2. Spectral Node Label Assignment In our proposed pipeline, a spectral node is assigned a pseudo-label via a probabilistic likelihood estimation that aggregates local, spectral, and global label profiling (as formulated in Eq. 7.6 and 7.7). To evaluate this, we replace our probabilistic majority voting approach with centrality-based cluster representations. Specifically, we identify the training node within each cluster that exhibits the highest network centrality and assign its label to the corresponding spectral node. We experiment with *Betweenness Centrality, Closeness Centrality, Degree Centrality, Eigenvector Centrality, Katz Centrality, PageRank, and DivRank*. Our findings indicate that centrality-based label assignment degrades downstream GNN performance. In highly heterophilic graphs, a central hub node is often connected to numerous nodes of diverse classes; hence, its individual label is rarely a reliable semantic representation for the entire structural cluster.

3. Spectral Node Feature Initialisation By default, PathLens initialises the feature vector of a spectral node with a random sequence of zeroes and ones to act as a neutral structural bridge, deliberately avoiding the averaging of neighbour features to prevent over-smoothing. We contrast this neutral initialisation against borrowing the exact feature vector of the most central node in the cluster. Using the same centrality measures as above (*Betweenness, Closeness, Degree, Eigenvector, Katz, PageRank, and DivRank*), we map the hub node's features directly to the spectral node. The empirical drop in performance confirms that borrowing explicit node features biases the spectral node too heavily toward a specific local neighbourhood. The random binary initialisation, conversely, ensures the spectral node acts solely as an unbiased structural conduit for information flow without injecting feature-space noise.

4. Pathway Connectivity Principle Finally, we evaluate the method by which spectral nodes are wired back into the original graph. PathLens uses a structured connectivity principle based on node importance (via PageRank or DivRank) to select the anchor nodes for pathway formation. We ablate this by replacing the importance-based wiring with random connectivity, where the spectral node is randomly connected to the exact same number of intra-cluster nodes belonging to the same class. The results clearly favour the importance-based wiring. Connecting based on importance based criteria ensures that the new pathways are anchored to influential regions of the graph, maximising the global reach and structural efficiency of the reduced shortest paths. Random connections, even when strictly constrained to same-class nodes, limits the efficacy of the enhanced information flow.

Design Choice Variation	Cornell	Texas	Wisconsin	Actor	ChamF	SquiF
1. Choice of Clustering Algorithm						
Affinity Propagation	74.95 ± 10.92	70.50 ± 10.67	90.86 ± 6.82	34.40 ± 3.62	55.21 ± 4.16	45.88 ± 7.73
Agglomerative	73.33 ± 12.39	67.75 ± 14.78	92.41 ± 4.89	35.44 ± 3.68	- -	43.57 ± 5.38
Birch	76.06 ± 10.18	72.75 ± 13.91	92.47 ± 4.02	34.12 ± 5.08	57.43 ± 6.01	45.51 ± 7.15
Bisecting K-Means	79.19 ± 8.84	75.17 ± 7.75	90.00 ± 6.67	37.44 ± 3.61	56.81 ± 6.07	46.49 ± 6.78
DBSCAN	- -	- -	- -	34.77 ± 4.01	56.74 ± 6.11	40.74 ± 10.08
Gaussian Mixture Models	73.33 ± 12.39	67.42 ± 15.17	92.41 ± 4.89	34.37 ± 3.18	- -	43.57 ± 5.38
HDBSCAN	- -	- -	- -	34.72 ± 2.06	55.10 ± 4.25	41.32 ± 10.30
K-Means	73.33 ± 12.39	67.75 ± 14.78	92.41 ± 4.89	35.17 ± 4.10	- -	43.62 ± 5.37
Louvain	75.25 ± 8.74	80.25 ± 9.55	90.12 ± 6.31	33.38 ± 3.26	56.60 ± 3.33	43.02 ± 8.92
Mean Shift	- -	- -	87.16 ± 7.36	33.36 ± 3.75	- -	- -
OPTICS	- -	- -	35.21 ± 3.91	56.94 ± 5.78	41.63 ± 9.45	- -
2. Spectral Node Label Assignment						
Betweenness Centrality	78.79 ± 10.85	65.50 ± 16.51	91.91 ± 4.99	34.31 ± 4.45	59.24 ± 5.50	45.23 ± 6.97
Closeness Centrality	78.18 ± 11.01	64.75 ± 19.74	91.98 ± 4.95	34.58 ± 4.03	56.11 ± 5.57	46.68 ± 9.39
Degree Centrality	78.28 ± 10.95	72.00 ± 12.81	91.79 ± 4.94	34.04 ± 4.01	57.64 ± 4.47	45.19 ± 7.52
Eigenvector Centrality	78.79 ± 10.85	70.17 ± 14.98	91.79 ± 4.94	35.96 ± 4.15	57.64 ± 5.42	44.16 ± 6.49
Katz Centrality	78.18 ± 11.01	71.50 ± 13.58	91.79 ± 4.94	34.57 ± 3.00	58.72 ± 4.72	44.82 ± 6.84
PageRank	77.98 ± 11.18	72.42 ± 13.28	91.79 ± 4.94	34.46 ± 4.60	56.67 ± 5.13	47.11 ± 7.55
DivRank	78.38 ± 10.90	66.58 ± 14.81	91.91 ± 4.89	34.93 ± 3.89	59.38 ± 6.12	44.76 ± 5.27
3. Spectral Node Feature Initialisation						
Betweenness Centrality	73.64 ± 12.27	64.75 ± 14.21	83.64 ± 9.17	33.80 ± 3.73	57.99 ± 2.80	42.56 ± 6.90
Closeness Centrality	73.64 ± 12.27	64.75 ± 14.21	86.30 ± 6.61	34.63 ± 4.52	58.26 ± 3.50	42.34 ± 6.72
Degree Centrality	73.64 ± 12.27	64.25 ± 15.48	82.47 ± 9.27	33.03 ± 4.68	57.78 ± 2.77	43.01 ± 7.22
Eigenvector Centrality	73.64 ± 12.27	64.25 ± 15.48	82.47 ± 9.27	33.80 ± 4.17	57.50 ± 3.23	43.58 ± 8.67
Katz Centrality	73.64 ± 12.27	64.25 ± 15.48	82.47 ± 9.27	34.14 ± 4.11	55.69 ± 2.77	42.59 ± 5.49
PageRank	73.64 ± 12.27	64.25 ± 15.48	82.47 ± 9.27	33.54 ± 4.18	57.36 ± 4.87	41.97 ± 7.32
DivRank	73.64 ± 12.27	64.50 ± 14.52	86.30 ± 6.61	34.11 ± 3.43	59.17 ± 2.79	42.43 ± 7.33
4. Pathway Connectivity Principle						
Random Connectivity	76.46 ± 8.96	76.67 ± 10.71	93.27 ± 4.19	36.93 ± 3.73	56.70 ± 4.45	43.28 ± 5.33
Base GNN - - LINKX	67.88 ± 14.22	62.42 ± 14.60	81.17 ± 9.27	33.88 ± 3.55	57.74 ± 2.98	43.14 ± 8.33
PathLens (LINKX)	81.82 ± 7.58	78.67 ± 11.51	95.12 ± 3.00	35.62 ± 3.80	61.46 ± 3.91	47.44 ± 6.29

Table 7.8: Performance comparison of various alternative design choices for PathLens components across six heterophilic datasets where the base GNN is **LINKX**. The results demonstrate the accuracy obtained when substituting our chosen methods with standard alternatives, compared against both the vanilla Base GNN and the complete PathLens architecture. "- -" represents clustering did not converge.

Design Choice Variation	Cornell	Texas	Wisconsin	Actor	ChamF	SquiF
1. Choice of Clustering Algorithm						
Affinity Propagation	81.62 ± 7.51	77.83 ± 8.50	- -	- -	- -	- -
Agglomerative	84.34 ± 7.91	77.67 ± 8.62	94.44 ± 2.57	37.17 ± 1.44	- -	42.71 ± 6.45
Birch	84.75 ± 5.07	75.58 ± 11.92	93.58 ± 3.86	36.71 ± 1.48	60.14 ± 2.13	41.66 ± 5.87
Bisecting K-Means	84.55 ± 7.20	76.67 ± 7.30	93.52 ± 4.33	35.83 ± 1.66	59.51 ± 3.88	42.49 ± 10.58
DBSCAN	- -	- -	- -	- -	59.72 ± 2.29	42.37 ± 8.40
Gaussian Mixture Models	84.44 ± 7.88	77.75 ± 8.64	94.01 ± 2.62	37.18 ± 1.84	- -	42.16 ± 6.55
HDBSCAN	- -	- -	- -	- -	59.34 ± 2.51	42.46 ± 7.41
K-Means	84.24 ± 7.95	77.75 ± 8.57	94.01 ± 2.62	37.17 ± 1.66	- -	42.55 ± 6.74
Louvain	81.92 ± 7.71	76.58 ± 8.31	95.37 ± 2.86	37.11 ± 0.90	60.10 ± 2.96	42.95 ± 7.42
Mean Shift	- -	- -	- -	- -	- -	- -
OPTICS	- -	- -	- -	- -	59.62 ± 2.25	42.49 ± 7.97
2. Spectral Node Label Assignment						
Betweenness Centrality	84.55 ± 4.73	82.92 ± 5.46	92.53 ± 5.57	36.52 ± 1.58	60.07 ± 1.87	40.88 ± 9.97
Closeness Centrality	84.55 ± 4.73	82.92 ± 5.46	92.04 ± 5.53	36.95 ± 1.81	60.56 ± 2.50	40.80 ± 8.87
Degree Centrality	84.55 ± 4.73	82.17 ± 5.05	93.77 ± 4.61	36.53 ± 1.72	61.32 ± 1.71	38.45 ± 10.32
Eigenvector Centrality	84.55 ± 4.73	81.67 ± 5.23	93.77 ± 4.61	36.65 ± 1.69	60.83 ± 2.40	40.55 ± 9.08
Katz Centrality	84.65 ± 4.72	81.67 ± 4.99	93.77 ± 4.61	36.97 ± 1.57	60.45 ± 2.20	41.61 ± 9.63
PageRank	84.65 ± 4.72	81.67 ± 5.23	93.77 ± 4.61	36.88 ± 2.13	58.54 ± 3.04	38.09 ± 10.54
DivRank	84.95 ± 4.74	82.42 ± 5.48	93.77 ± 4.54	36.83 ± 1.51	60.90 ± 3.20	37.51 ± 9.53
3. Spectral Node Feature Initialisation						
Betweenness Centrality	82.93 ± 7.07	77.17 ± 6.15	92.84 ± 3.31	38.80 ± 1.17	60.76 ± 2.73	40.40 ± 6.40
Closeness Centrality	82.73 ± 6.77	77.33 ± 6.05	91.60 ± 3.20	38.71 ± 1.11	59.90 ± 3.33	38.48 ± 10.51
Degree Centrality	83.33 ± 7.08	79.83 ± 4.67	92.84 ± 3.69	38.67 ± 1.52	59.24 ± 3.91	43.06 ± 5.61
Eigenvector Centrality	82.93 ± 7.07	79.92 ± 4.81	92.59 ± 2.45	38.80 ± 1.41	59.65 ± 3.24	43.06 ± 5.61
Katz Centrality	83.03 ± 7.09	80.17 ± 4.77	92.47 ± 2.69	39.00 ± 1.67	59.31 ± 3.77	40.14 ± 8.08
PageRank	82.83 ± 6.79	80.08 ± 4.71	92.47 ± 2.69	38.54 ± 1.49	59.24 ± 3.43	40.23 ± 8.50
DivRank	82.12 ± 8.00	77.17 ± 6.15	92.90 ± 2.98	38.82 ± 1.38	61.01 ± 2.69	39.38 ± 10.11
4. Pathway Connectivity Principle						
Random Connectivity	84.14 ± 5.61	74.58 ± 8.94	94.01 ± 3.48	36.31 ± 2.20	59.65 ± 2.89	38.86 ± 7.21
Base GNN - - BernNet	83.74 ± 4.91	82.67 ± 4.35	93.52 ± 3.25	38.80 ± 1.32	58.89 ± 2.20	42.83 ± 3.08
PathLens (BernNet)	86.77 ± 3.64	83.83 ± 6.17	97.04 ± 2.13	37.63 ± 1.44	61.74 ± 2.97	43.33 ± 7.16

Table 7.9: Performance comparison of various alternative design choices for PathLens components across six heterophilic datasets where the base GNN is **BernNet**. The results demonstrate the accuracy obtained when substituting our chosen methods with standard alternatives, compared against both the vanilla Base GNN and the complete PathLens architecture. "- -" represents clustering did not converge.

Design Choice Variation	Cornell	Texas	Wisconsin	Actor	ChamF	SquiF
1. Choice of Clustering Algorithm						
Affinity Propagation	65.56 ± 12.54	77.58 ± 8.73	- -	- -	- -	- -
Agglomerative	75.05 ± 10.60	76.42 ± 10.79	88.52 ± 7.66	38.70 ± 1.65	- -	52.38 ± 3.87
Birch	82.02 ± 7.93	70.08 ± 9.77	91.54 ± 3.41	37.16 ± 1.63	57.01 ± 3.81	48.15 ± 4.73
Bisecting K-Means	74.14 ± 10.36	71.58 ± 12.68	92.96 ± 3.97	37.02 ± 2.71	55.76 ± 5.31	49.88 ± 3.08
DBSCAN	- -	- -	- -	- -	56.56 ± 3.66	50.47 ± 6.52
Gaussian Mixture Models	75.05 ± 10.60	76.42 ± 10.79	88.52 ± 7.66	38.24 ± 2.54	- -	52.21 ± 4.11
HDBSCAN	- -	- -	- -	- -	56.46 ± 3.87	50.69 ± 6.41
K-Means	75.05 ± 10.60	76.42 ± 10.79	88.52 ± 7.66	38.37 ± 2.32	- -	52.03 ± 3.63
Louvain	68.79 ± 12.61	72.33 ± 8.53	90.43 ± 4.92	- -	58.33 ± 6.76	47.96 ± 5.74
Mean Shift	- -	- -	88.40 ± 7.26	- -	- -	- -
OPTICS	- -	- -	- -	- -	56.67 ± 3.70	51.05 ± 7.10
2. Spectral Node Label Assignment						
Betweenness Centrality	75.05 ± 10.56	77.17 ± 13.88	93.15 ± 4.09	37.56 ± 1.99	58.26 ± 3.53	51.29 ± 5.27
Closeness Centrality	74.75 ± 11.01	77.92 ± 14.23	92.41 ± 4.17	37.80 ± 2.47	59.04 ± 2.71	52.22 ± 4.80
Degree Centrality	75.25 ± 11.00	79.00 ± 14.14	92.22 ± 3.97	37.76 ± 2.06	58.51 ± 2.98	51.30 ± 5.36
Eigenvector Centrality	74.95 ± 11.12	79.00 ± 14.14	92.90 ± 3.78	37.32 ± 2.35	58.54 ± 2.86	50.31 ± 7.15
Katz Centrality	76.87 ± 8.43	79.08 ± 14.22	93.02 ± 3.83	37.65 ± 1.90	59.04 ± 2.56	50.88 ± 5.83
PageRank	75.15 ± 11.25	78.92 ± 14.29	92.78 ± 3.72	37.88 ± 2.40	58.99 ± 2.48	50.73 ± 5.45
DivRank	75.05 ± 11.54	77.00 ± 14.38	92.90 ± 4.42	37.90 ± 1.92	57.36 ± 2.40	49.81 ± 7.29
3. Spectral Node Feature Initialisation						
Betweenness Centrality	78.38 ± 6.86	73.33 ± 7.33	92.47 ± 3.48	39.38 ± 2.09	56.63 ± 3.38	45.74 ± 4.84
Closeness Centrality	78.08 ± 6.70	74.25 ± 6.71	90.68 ± 5.29	39.28 ± 1.89	56.67 ± 3.09	46.00 ± 5.74
Degree Centrality	78.38 ± 6.86	66.42 ± 11.69	92.04 ± 4.03	39.89 ± 1.88	56.98 ± 4.05	45.34 ± 6.34
Eigenvector Centrality	78.18 ± 6.87	66.42 ± 11.69	90.62 ± 4.03	39.60 ± 1.98	57.36 ± 3.53	46.32 ± 5.75
Katz Centrality	77.98 ± 6.87	65.83 ± 12.61	90.62 ± 4.03	39.56 ± 1.91	56.91 ± 4.02	46.78 ± 4.93
PageRank	78.28 ± 6.91	66.17 ± 12.04	90.68 ± 4.04	39.41 ± 2.13	57.26 ± 2.98	45.58 ± 6.64
DivRank	78.18 ± 6.87	74.17 ± 6.69	92.47 ± 3.48	39.45 ± 1.86	56.81 ± 3.49	45.11 ± 6.22
4. Pathway Connectivity Principle						
Random Connectivity	75.05 ± 9.77	78.08 ± 10.18	88.70 ± 7.11	42.33 ± 2.50	57.78 ± 2.21	37.68 ± 6.45
Base GNN - - DirSAGE	71.41 ± 11.01	79.00 ± 11.03	92.41 ± 4.48	38.80 ± 1.91	57.15 ± 5.76	42.58 ± 7.64
PathLens (DirSAGE)	76.97 ± 7.54	81.83 ± 11.00	93.40 ± 3.73	38.30 ± 2.20	59.27 ± 2.87	52.96 ± 4.40

Table 7.10: Performance comparison of various alternative design choices for PathLens components across six heterophilic datasets where the base GNN is **DirSAGE**. The results demonstrate the accuracy obtained when substituting our chosen methods with standard alternatives, compared against both the vanilla Base GNN and the complete PathLens architecture. "- -" represents clustering did not converge.

7.8 Working Analysis and Novel Measures

7.8.1 Homophily Perspective

As discussed in Sections 7.4 and 7.5, PathLens gives superior performance on several heterophilic datasets and downstream GNN models. To evaluate the assumption that most GNNs perform better on graphs with high homophily, we explored several homophily measures that are available in the literature. Let $G = (V, E)$ is a graph with n nodes, and each node $u \in V$ has a class label $y_u \in \{0, 1, \ldots, C-1\}$, where C is the total number of classes and C_k represents the set of nodes in class k. Node homophily (Pei et al., 2020), which computes the ratio of neighbours that have the same class for an ego node and then computes the mean of these ratios across all nodes, is defined as

$$H_{node} = \frac{1}{|V|} \sum_{v \in V} \frac{|\{u \in N(v) : y_v = y_u\}|}{|N(v)|} \tag{7.14}$$

where $N(v)$ denotes the neighbourhood of node v

Edge homophily (Zhu et al., 2020) is another standard measure for homophily, which is the fraction of edges connecting two nodes with the same class and is calculated as

$$H_{edge} = \frac{|\{(v, u) \in E : y_v = y_u\}|}{|E|} \tag{7.15}$$

Lim et al. (2021) showed that these two simple and intuitive homophily measures are sensitive to the number of classes and their balance, and proposed Improved homophily measure as

$$H_{imp} = \frac{1}{C-1} \sum_{k=0}^{C-1} [h_k - \frac{|C_k|}{n}]_+ \tag{7.16}$$

where $[a]_+ = max(a, 0)$, and h_k is the class-wise homophily metric defined as

$$h_k = \frac{\sum_{u \in C_k} |\{u \in N(v) : y_v = y_u|}{\sum_{u \in C_k} |N(v)|} \tag{7.17}$$

Platonov et al. (2022) showed that Improved homophily can also lead to unreliable results and thus proposed a new measure, Adjusted homophily, by correcting the number of intra-class edges by their expected value and is thus insensitive to the number of classes and their balance. Based on Edge homophily, Adjusted homophily is computed as

$$H_{adj} = \frac{H_{edge} - \sum_{k=1}^{C} D_k^2/(2|E|)^2}{1 - \sum_{k=1}^{C} D_k^2/(2|E|)^2} \tag{7.18}$$

where $D_k = \sum_{v:y_v=k} d(v)$, and $d(v)$ represents degree of node v.

Luan et al. (2022) proposed Aggregated homophily based on post aggregation node similarity. Please refer the source for more details.

PathLens on heterophilic graphs. We compute all the above-discussed homophily scores on all seven heterophilic datasets before and after using PathLens. From Table

7.11, we can observe that PathLens consistently increases the Adjusted homophily and Edge homophily scores across all the datasets. We observe an almost similar trend for Node homophily. As shown in Platonov et al. (2022), Improved homophily leads to unreliable results with no clear pattern in the scores. We also observe a similar unclear pattern in Aggregated homophily. Analysing the results from Table 7.2 and the homophily scores, we can observe that PathLens achieves better results in datasets where it leads to a high increase in homophily scores.

PathLens on homophilic graphs. To further solidify our hypothesis empirically, we conduct another set of experiments on five commonly used homophilic datasets, namely **Cora**, **Citeseer**, **Photo**, **Computers**, and **Pubmed**. We show the statistics of these five datasets in Table 7.12. We perform a similar experimental setup for homophilic datasets to that used for heterophilic datasets. We show the results for GNN performance on homophilic graphs in Table 7.13 and report the homophily scores in Table 7.11. We observe that using PathLens for homophilic graphs leads to a decrease in the homophily level as measured by all five available homophily measures, with a minor exception in the case of Aggregated homophily. The effect of homophily reduction reflects the drop in performance across almost every homophilic dataset and the chosen GNN. We would like to again emphasise that **our research focuses on improving heterophilic graph learning.**

The exhaustive experimentation provides ample empirical evidence that homophily is a desired graph property, enabling most GNNs to perform better. We show empirically that PathLens introduces homophily into heterophilic datasets, thus ***structurally enhancing heterophilic graphs for GNNs***. Therefore, we both examine and leverage the common notion that most GNNs perform better on high homophily datasets by increasing homophily in heterophilic datasets.

7.8.2 Beyond Homophily

PathLens' Intuition. We design PathLens to enable information flow between different regions of the heterophilic graph by bringing vertices with similar labels closer to each other than the dissimilar ones. PathLens will likely facilitate additional paths between a pair of nodes in a given heterophilic/homophilic graph, potentially reducing the shortest distance or keeping it unchanged for the node pair under consideration. Globally, it reduces the average shortest path length in the given graph for nodes with the same class labels as well as different class labels. Intuitively, for a heterophilic graph, where the number of direct connections between similar nodes is less than the dissimilar ones, the additional paths are likely to reduce the average shortest path length for similar nodes comparatively more than the dissimilar nodes. Similarly, in the homophilic graph, where the number of direct connections between dissimilar nodes is less than the similar ones, it brings vertices with dissimilar labels closer to each other than similar ones.

Novel Measures. For a given graph G, let d_{ij} denotes the shortest path length between a pair of nodes i and j. If the two nodes are not connected, we consider it one plus the graph's diameter. Following the notations used above in homophily equations, we define

	Heterophilic Datasets							Homophilic Datasets				
	Cornell	Texas	Wisc	Actor	ChamF	SquiF	arXiv	Cora	Cite	Comp	Photo	Pubmed
Node Homophily												
G	0.1182	0.0872	0.1706	0.2219	0.2481	0.1961	0.2893	0.8251	0.7062	0.7853	0.8364	0.7924
PL (%↑)	25.63	39.22	12.89	1.44	3.91	5.20	-0.72	-5.58	-5.82	-0.90	-2.57	-4.22
Edge Homophily												
G	0.1321	0.1118	0.2061	0.2194	0.2403	0.2095	0.2181	0.8099	0.7355	0.7772	0.8272	0.8023
PL (%↑)	34.89	69.23	21.68	5.6	8.03	0.95	1.74	-13.10	-15.05	-0.85	-1.30	-6.03
Adjusted Homophily												
G	-0.2029	-0.2260	-0.1323	0.0061	0.0347	0.0115	0.0051	0.7710	0.6706	0.6823	0.7850	0.6860
PL (%↑)	49.82	66.77	99.09	121.31	57.06	19.13	139.21	-17.08	-21.20	-1.56	-1.84	-11.72
Improved Homophily												
G	0.0499	0	0.0495	0.0074	0.0465	0.0409	0.0671	0.7657	0.6267	0.7001	0.7722	0.6641
(%↑)	-39.67	∞	104.84	131.08	31.39	20.44	-1.34	-12.66	-19.97	-1.35	-1.45	-11.09
Aggregated Homophily												
G	0.2077	0.0984	0.2829	0.2362	0.3067	0.1053	0.1251	0.4679	0.4385	0.3873	0.2065	0.7094
PL (%↑)	-12.22	-16.66	-71.33	-5.75	-19.53	78.91	12.15	-70.82	-56.78	-7.84	25.03	-51.15

Table 7.11: Homophily analysis for different homophily measures across 7 heterophilic and 5 homophilic datasets. It shows injection of homophily into heterophilic datasets using PathLens. 'G' represents original graph and its corresponding values show the various homophily scores. 'PL' represents augmented graph after PathLens and its corresponding values show the percentage increment in various homophily scores respectively. Please refer Sec. 7.8.1 for detailed insights.

Dataset	# Nodes	# Edges	# Features	# Classes
Cora	2,708	5,278	1,433	7
Citeseer	3,327	4,552	3,703	6
Photo	7,487	119,043	745	8
Computers	13,381	245,778	767	10
Pubmed	19,717	44,324	500	3

Table 7.12: Statistics of homophilic datasets.

intraclass average shortest path length as follows:

$$ASPL_{SC} = \frac{1}{C}\sum_{k=0}^{C-1}\frac{\sum_{i,j} d_{ij}}{\binom{|C_k|}{2}} \quad \forall\, i, j \in C_k;\ i \neq j \tag{7.19}$$

and interclass average shortest path length as follows:

	Cora	Pubmed	Citeseer	Computers	Photo
MLP	76.64 ± 1.37	88.69 ± 0.57	77.03 ± 1.23	86.47 ± 1.58	90.44 ± 2.69
PL	76.77 ± 1.39	88.56 ± 0.53	77.23 ± 0.82	86.71 ± 1.07	91.69 ± 2.09
GraphSAGE	88.93 ± 1.22	90.83 ± 0.47	81.44 ± 1.49	87.10 ± 1.60	92.41 ± 1.91
PL	84.45 ± 1.72	90.26 ± 0.35	79.63 ± 1.14	86.46 ± 1.21	91.60 ± 1.22
GAT	89.42 ± 0.73	90.31 ± 0.30	82.12 ± 1.46	89.29 ± 0.85	93.85 ± 0.59
PL	84.76 ± 1.22	88.60 ± 0.31	80.90 ± 1.09	86.94 ± 0.89	92.21 ± 0.69
APPNP	88.83 ± 0.65	89.25 ± 0.48	81.73 ± 1.73	88.73 ± 0.77	94.48 ± 0.85
PL	83.82 ± 1.05	89.16 ± 0.53	80.66 ± 0.68	88.15 ± 1.02	94.09 ± 1.04
GPRGNN	89.76 ± 1.00	91.56 ± 0.36	82.48 ± 1.73	88.94 ± 1.18	93.26 ± 1.34
PL	85.77 ± 1.94	89.74 ± 0.25	81.84 ± 1.06	86.42 ± 2.93	92.56 ± 1.12
LINKX	81.22 ± 2.78	88.09 ± 0.96	74.18 ± 1.27	89.50 ± 1.03	94.65 ± 1.07
PL	70.48 ± 5.81	87.88 ± 0.70	68.82 ± 2.05	88.14 ± 0.80	94.02 ± 0.92
GATv2	88.95 ± 1.05	90.34 ± 0.33	82.06 ± 0.94	90.19 ± 0.59	93.90 ± 0.80
PL	85.40 ± 1.13	88.71 ± 0.26	81.01 ± 1.40	87.43 ± 0.67	92.32 ± 0.56
pGNN	89.94 ± 1.43	91.75 ± 0.29	81.28 ± 1.10	89.30 ± 0.71	94.09 ± 0.91
PL	83.11 ± 1.05	90.00 ± 0.52	78.30 ± 0.94	86.91 ± 1.46	92.31 ± 1.32
DAGNN	89.61 ± 1.16	91.97 ± 0.43	81.81 ± 1.21	87.34 ± 7.13	93.07 ± 3.22
PL	84.92 ± 1.46	89.54 ± 0.38	80.28 ± 1.32	80.15 ± 10.46	80.25 ± 2.39
BernNet	89.52 ± 0.83	90.75 ± 0.63	82.13 ± 0.91	77.96 ± 19.51	82.38 ± 31.82
PL	84.26 ± 2.41	90.17 ± 0.37	80.54 ± 1.01	79.21 ± 8.64	89.45 ± 7.18
AeroGNN	40.51 ± 23.92	55.53 ± 6.70	33.97 ± 13.89	29.13 ± 17.03	36.99 ± 17.85
PL	35.21 ± 22.33	56.04 ± 5.04	32.90 ± 7.29	17.27 ± 18.74	21.37 ± 18.70
DirSAGE	85.14 ± 3.45	90.41 ± 0.52	79.40 ± 1.50	89.65 ± 0.96	95.85 ± 0.70
PL	84.08 ± 1.84	90.75 ± 0.38	77.53 ± 1.18	89.00 ± 1.49	95.49 ± 0.72
PMLPGCN	78.19 ± 9.25	89.47 ± 0.54	74.52 ± 6.03	83.18 ± 3.46	81.74 ± 11.59
PL	72.34 ± 8.10	85.68 ± 1.55	72.90 ± 3.51	82.93 ± 4.30	76.26 ± 14.48
PMLPAPPNP	73.63 ± 13.49	88.81 ± 1.31	73.44 ± 5.46	79.63 ± 6.41	77.44 ± 17.49
PL	70.58 ± 9.50	86.00 ± 1.51	72.87 ± 3.51	80.25 ± 8.39	72.63 ± 17.48
UniFilter	35.32 ± 25.68	59.45 ± 14.18	33.52 ± 7.70	28.73 ± 22.65	26.37 ± 21.03
PL	30.37 ± 24.18	60.00 ± 14.20	29.19 ± 10.43	31.26 ± 24.32	27.76 ± 19.08
Specformer	48.38 ± 24.03	60.82 ± 18.14	43.49 ± 12.45	34.31 ± 20.15	34.97 ± 19.64
PL	43.41 ± 24.15	60.29 ± 13.17	41.11 ± 10.61	35.57 ± 25.01	35.73 ± 18.91

Table 7.13: Performance comparison of PathLens w.r.t. various models on five homophilic datasets. We report accuracy for GNN models and PathLens. Performance drop is observed for all GNNs as expected due to the decrease in homophily.

	Heterophilic Datasets						Homophilic Datasets				
	Cornell	Texas	Wisc	Actor	ChamF	SquiF	Cora	Citeseer	Comp	Photo	Pubmed
Average Shortest Path Length – – Original Graph											
SC	3.262	39.307	3.161	4.101	3.835	3.139	387.016	1273.644	592.844	269.256	6.315
DC	3.3	3.205	3.229	4.101	3.921	3.165	416.893	1315.283	595.756	271.303	6.6199
Average Shortest Path Length – – PathLens											
SC	3.032	2.518	2.81	3.578	3.334	2.992	233.757	980.329	326.735	206.88	3.661
DC	3.071	2.705	2.903	3.578	3.411	3.012	236.863	995.742	327.806	207.986	3.693
% ↓ Average Shortest Path Length – – PathLens											
SC	**7.050**	**93.594**	**11.104**	**12.752**	**13.063**	4.683	39.600	23.029	44.886	23.166	42.026
DC	6.939	15.600	10.096	**12.752**	13.006	**4.834**	**43.183**	**24.294**	**44.976**	**23.338**	**44.213**

Table 7.14: Analysis of average shortest path length computed between nodes with same class (SC) and different class (DC) respectively. Please refer Sec. 7.8.2 for detailed insights.

$$ASPL_{DC} = \frac{1}{\binom{C}{2}} \sum_{\substack{p,q=1 \\ p<q}}^{C} \frac{\sum_{i \in C_p} \sum_{j \in C_q} d_{ij}}{|C_p||C_q|} \tag{7.20}$$

As intuitively discussed above, we now empirically show in Table 7.14 the values of $ASPL_{SC}$ and $ASPL_{DC}$, and the corresponding % drops (∇) after applying PathLens. Analysing Table 7.11 and Table 7.14 together offer valuable insights. For heterophilic graphs, the increase in homophily levels (Adjusted Homophily and Edge Homophily) directly correlates with the ASPL Drop Ratio, $ADR = \nabla ASPL_{SC}/\nabla ASPL_{DC}$. We observe the highest homophily injection for Texas, where ADR is the highest, and the lowest homophily injection for Squirrel-Filtered, where ADR is the lowest. Further insights into the results show that Texas obtains the highest performance gains after applying PathLens corresponding to its highest ADR. From Table 7.2, we see that the Actor is the toughest to fit for various GNNs because it has the same intraclass and interclass ASPL. Also, it obtains minimal accuracy gains after PathLens as it observed the same drops in $ASPL_{SC}$ and $ASPL_{DC}$. For homophilic graphs, the decrease in homophily levels directly correlates with the Inverse ASPL Drop Ratio, $Inverse\ ADR = \nabla ASPL_{DC}/\nabla ASPL_{SC}$.

To summarise, we say that increasing homophily in a graph is desirable but we would also like to achieve high ASPL Drop Ratio. Computing ASPL is a costly operation that limits the analysis of massive graphs, like arXiv-Year, in our case.

7.9 Summary

In this chapter, we introduced a perspective of data enrichment that enabled better performance of heterophilic and non-heterophilic GNN algorithms on heterophilic

graphs by injecting homophily. We proposed PathLens that enables better information flow in heterophilic graphs by introducing additional paths, thus bringing similar nodes that may be present in faraway regions in the graph closer to each other. We proved the effectiveness of our technique through several experiments and analyses. We offered a fresh perspective of intraclass and interclass average shortest path lengths that goes beyond homophily. Exhaustive experimentation revealed the high sensitivity of many recent GNNs to the random seed used for data splitting. This work highlighted the importance of data enrichment rather than the need to design specialised models.

Chapter 8

Conclusion

This thesis embarked on a research journey to address the central challenge of effective representation learning on real-world graphs. We posited that a deliberate, data-centric progression from enriching the information available at the node level to augmenting the graph's fundamental structure, provides a powerful paradigm for unlocking new levels of performance. This final chapter recapitulates the path of this investigation, synthesises the core contributions and insights, acknowledges the limitations of the present work, and outlines promising avenues for future inquiry with concluding remarks.

8.1 Recapitulation of the Research Trajectory

Our investigation was structured as a two-part narrative, progressing from a specific application to a general methodology. In Part I, we focused on the high-impact domain of citation recommendation in scholarly networks. Our initial, formative exploration, detailed in Chapter 4, introduced the SymTax framework. This work established the value of data enrichment by incorporating two novel strategies in a transductive setting: 1) behavioural enrichment through a Symbiosis module that modelled human research patterns by exploring the outgoing neighbourhood of candidate papers, and 2) feature enrichment via Taxonomy Fusion, which created richer conceptual representations by leveraging hyperbolic geometry. While successful, this work revealed critical challenges in scalability and the need for a more realistic evaluation standard.

Building directly upon these challenges, Chapter 5 detailed a more scalable, robust, and practically-grounded framework. We first addressed the methodological shortcomings of prior work by defining a rigorous Inductive Evaluation Protocol with strict temporal enforcement. We then introduced Profiler, a highly efficient, non-learnable module that enriched each paper's representation by statically fusing signals from its inward neighbourhood, i.e., the very contexts in which it has been cited. This powerful signal was then adaptively integrated into the DAVINCI reranker. This work not only achieved new state-of-the-art results in speed and accuracy but also established a more robust and reliable standard for evaluating future citation recommendation systems.

In Part II, we transcended this specific application to address a more fundamental problem in graph machine learning. Chapter 7 generalised our core principle from data enrichment to structural augmentation to tackle the challenge of GNNs on heterophilic graphs. We introduced PathLens, a novel, structure-driven framework that fundamentally

augments a graph's topology. By creating new information pathways via supernodes derived from spectral clustering, PathLens facilitates information flow between distant but similar nodes, making the graph inherently more learnable for a wide array of standard GNN architectures without requiring any model-level changes.

8.2 The Central Thesis: A Data-Centric Lens as a Powerful Framework

Viewed in totality, this body of work validates our central thesis: that a deliberate, data-centric progression from node-level enrichment to graph-level augmentation is a highly effective paradigm for advancing machine learning on complex graphs. We demonstrated that before resorting to more complex model architectures, significant gains can be realised by first enhancing the quality and richness of the data itself.

In the initial stages, we showed that enriching nodes with signals from their local neighbourhood, whether outward-facing (Symbiosis) or inward-facing (Profiler), provides the model with more potent and discriminative information. Subsequently, we showed that a higher level of abstraction, i.e., augmenting the graph's structure itself to improve global information flow (PathLens), can provide a universally beneficial pre-processing step for an entire class of models. This journey underscores another key finding: for specialised domains, architectural sophistication, task-aligned design choices, and the integration of domain-specific knowledge are more salient drivers of success than just the raw parameter count.

8.3 Limitations of the Present Work

While this thesis presents a series of significant advancements, it is important to acknowledge the boundaries of the current work:

- Static graph assumption: The techniques proposed in our research articles of Public Profile, SymTax, and PathLens – all operate on a static snapshot of a graph. They do not inherently account for dynamic graphs where nodes and edges are added or removed over time.

- Scalability of structural augmentation: While PathLens was effective on a range of datasets, its reliance on spectral clustering could become a computational bottleneck when applied to truly massive, web-scale graphs with millions of nodes.

- Generalisability of enrichment techniques: The enrichment strategies developed in Part I (Profiler, Symbiosis, Taxonomy Fusion) are highly tuned for the semantics of scholarly networks. Translating concepts like "citation context" or "symbiotic behaviour" to other domains may be a non-trivial task.

- Hyperparameter selection in augmentation: PathLens introduces a new set of hyperparameters related to the structural augmentation process (e.g., number of spectral clusters, connectivity principles). While robust, finding the optimal configuration for a new graph requires a tuning process.

8.4 Avenues for Future Inquiry

The findings and limitations of this thesis open up several exciting and promising directions for future research.

- Unifying Enrichment and Augmentation: The most compelling next step is to create a unified, hybrid framework. Can the graph-level structural augmentation of PathLens be combined with the node-level signal enrichment of Profiler? Such a system may first create more efficient information pathways and then propagate richer, pre-enriched signals across them, potentially leading to a new state-of-the-art.
- Dynamic and Adaptive Systems: A critical area for development is adapting these static techniques for dynamic environments. This could involve creating efficient update rules for the Profiler representations as new citations appear, or developing methods for incremental structural augmentation with PathLens in evolving graphs.
- Generalising the "Public Profile" Concept: The idea of enriching a node's representation from its "inward signals" is powerful. Future work can potentially explore what this concept means in other domains. For a user on a social network, it could be the posts they are tagged in; for a product in an e-commerce network, it could be its appearance in user-created wishlists.
- Learning to Augment: The PathLens framework uses a principled but fixed strategy for augmentation. A fascinating future direction can be developing a learnable framework, perhaps using reinforcement learning, to determine the optimal structural augmentations for a given graph and downstream task.

8.5 Concluding Remarks

The deluge of interconnected data presents both a grand challenge and a remarkable opportunity for machine learning. This thesis has demonstrated that by looking beyond model-centric design and focusing on the data itself, we can make significant strides in our ability to learn from complex networks. Through a carefully charted journey from the specific task of citation recommendation to the general problem of learning on heterophilic graphs, we have developed a suite of novel techniques for data enrichment and structural augmentation. These contributions not only advance the state-of-the-art in their respective domains but also advocate for a powerful, data-centric philosophy that holds immense promise for the future of graph machine learning.

Chapter 9

Research Output and Course Work

9.1 Publications

- Karan Goyal, Mayank Goel, Vikram Goyal, and Mukesh Mohania. 2024. SymTax: Symbiotic Relationship and Taxonomy Fusion for Effective Citation Recommendation. In Findings of the Association for Computational Linguistics: ACL 2024, pages 8997–9008, Bangkok, Thailand. Association for Computational Linguistics.
- Karan Goyal, Dikshant Kukreja, Vikram Goyal, and Mukesh Mohania. Public Profile Matters: A Scalable Integrated Approach to Recommend Citations in the Wild. (Under review)
- Karan Goyal, Saankhya Samanta, Vikram Goyal, and Mukesh Mohania. 2025. PathLens: Structurally Enhancing Heterophilic Graphs for GNNs. In Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM 25), November 10–14, 2025, Seoul, Republic of Korea. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3746252. 3761256

9.2 Courses

S.No.	Course Name	Credit	Grade
1	CSE556-Natural Language Processing	4	A-
2	CSE642-Advanced Machine Learning	4	A
3	CSE568-Social Network Analysis	4	A
4	PIS790-Independent Study	4	A
5	ENG599s-Research Methods	2	A
	Total credits completed	**18**	**9.78**

Bibliography

Sami Abu-El-Haija, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. 2019. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *international conference on machine learning*. PMLR, 21–29.

Zafar Ali, Guilin Qi, Khan Muhammad, Pavlos Kefalas, and Shah Khusro. 2021. Global citation recommendation employing generative adversarial network. *Expert Systems with Applications* 180 (2021), 114888.

Mehdi Azabou, Venkataramana Ganesh, Shantanu Thakoor, Chi-Heng Lin, Lakshmi Sathidevi, Ran Liu, Michal Valko, Petar Veličković, and Eva L Dyer. 2023. Half-Hop: A graph upsampling approach for slowing down message passing. In *International Conference on Machine Learning*. PMLR, 1341–1360.

Annika Baumann, Johannes Haupt, Fabian Gebert, and Stefan Lessmann. 2018. Changing perspectives: Using graph metrics to predict purchase probabilities. *Expert Systems with Applications* 94 (2018), 137–148.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 3615–3620. https://doi.org/10.18653/v1/D19-1371

Chandra Bhagavatula, Sergey Feldman, Russell Power, and Waleed Ammar. 2018. Content-Based Citation Recommendation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 238–251. https://doi.org/10.18653/v1/N18-1022

Deyu Bo, Chuan Shi, Lele Wang, and Renjie Liao. 2023. Specformer: Spectral Graph Neural Networks Meet Transformers. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=0pdSt3oyJa1

Cristian Bodnar, Francesco Di Giovanni, Benjamin Chamberlain, Pietro Liò, and Michael Bronstein. 2022. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in gnns. *Advances in Neural Information Processing Systems* 35 (2022), 18527–18541.

Aleksandar Bojchevski, Johannes Gasteiger, Bryan Perozzi, Amol Kapoor, Martin Blais, Benedek Rózemberczki, Michal Lukasik, and Stephan Günnemann. 2020. Scaling Graph Neural Networks with Approximate PageRank *(KDD '20)*. Association for Computing Machinery, New York, NY, USA, 2464–2473. https://doi.org/10.1145/3394486.3403296

Lutz Bornmann, Robin Haunschild, and Rüdiger Mutz. 2021. Growth rates of modern science: a latent piecewise growth curve approach to model publication numbers from established and new literature databases. *Humanities and Social Sciences Communications* 8, 1 (2021), 1–15.

Shaked Brody, Uri Alon, and Eran Yahav. 2022. How Attentive are Graph Attention Networks?. In *International Conference on Learning Representations*. https://openreview.net/forum?id=F72ximsx7C1

Andrea Cavallo, Claas Grohnfeldt, Michele Russo, Giulio Lovisotto, and Luca Vassio. 2023. GCNH: A Simple Method For Representation Learning On Heterophilous Graphs. *2023 International Joint Conference on Neural Networks (IJCNN)* (2023), 1–8. https://api.semanticscholar.org/CorpusID:258291678

Ege Yiğit Çelik and Selma Tekir. 2025. CiteBART: Learning to Generate Citations for Local Citation Recommendation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 1703–1719. https://doi.org/10.18653/v1/2025.emnlp-main.89

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 2318–2335. https://doi.org/10.18653/v1/2024.findings-acl.137

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. 2020. Simple and Deep Graph Convolutional Networks. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 1725–1735. https://proceedings.mlr.press/v119/chen20v.html

Wei Chen, Chi Wang, and Yajun Wang. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1029–1038.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. 2021. Adaptive Universal Generalized PageRank Graph Neural Network. In *International Conference on Learning Representations*. https://openreview.net/forum?id=n6jl7fLxrP

Valerio Ciotti, Moreno Bonaventura, Vincenzo Nicosia, Pietro Panzarasa, and Vito Latora. 2016. Homophily and missing links in citation networks. *EPJ Data Science* 5 (2016), 1–14.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level Representation Learning using Citation-informed Transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 2270–2282. https://doi.org/10.18653/v1/2020.acl-main.207

Tao Dai, Li Zhu, Yaxiong Wang, and Kathleen M Carley. 2019. Attentive stacked denoising autoencoder with bi-lstm for personalized context-aware citation recommendation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2019), 553–568.

Anil Damle, Victor Minden, and Lexing Ying. 2019. Simple, direct and efficient multi-way spectral clustering. *Information and Inference: A Journal of the IMA* 8, 1 (2019), 181–203.

Priyangshu Datta, Suchana Datta, and Dwaipayan Roy. 2024. RAGing Against the Literature: LLM-Powered Dataset Mention Extraction. In *Proceedings of the 24th ACM/IEEE Joint Conference on Digital Libraries*. 1–12.

Chenhui Deng, Zhiqiang Zhao, Yongyu Wang, Zhiru Zhang, and Zhuo Feng. 2020a. GraphZoom: A Multi-level Spectral Approach for Accurate and Scalable Graph Embedding. In *International Conference on Learning Representations*. https://openreview.net/forum?id=r1lGO0EKDH

Yao Deng, Xi Zheng, Tianyi Zhang, Chen Chen, Guannan Lou, and Miryung Kim. 2020b. An analysis of adversarial attacks and defenses on autonomous driving models. In *2020 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE, 1–10.

John A Drozdz and Michael R Ladomery. 2024. The peer review process: past, present, and future. *British Journal of Biomedical Science* 81 (2024), 12054.

Lun Du, Xiaozhou Shi, Qiang Fu, Xiaojun Ma, Hengyu Liu, Shi Han, and Dongmei Zhang. 2022. GBK-GNN: Gated Bi-Kernel Graph Neural Networks for Modeling Both Homophily and Heterophily. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 1550–1558. https://doi.org/10.1145/3485447.3512201

Travis Ebesu and Yi Fang. 2017. Neural citation network for context-aware citation recommendation. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*. 1093–1096.

Paul Esau, Nazar Shmatko, and Alex Adam. 2025. GPTZero Finds Over 50 New Hallucinations in ICLR 2026 Submissions. *GPTZero* (5 Dec. 2025). https://gptzero.me/news/iclr-2026/

Guoji Fu, Peilin Zhao, and Yatao Bian. 2022. p-Laplacian Based Graph Neural Networks. In *International Conference on Machine Learning*. PMLR, 6878–6917.

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic neural networks. *Advances in neural information processing systems* 31 (2018).

Chen Gao, Yu Zheng, Nian Li, Yinfeng Li, Yingrong Qin, Jinghua Piao, Yuhan Quan, Jianxin Chang, Depeng Jin, Xiangnan He, et al. 2023. A survey of graph neural networks for recommender systems: Challenges, methods, and directions. *ACM Transactions on Recommender Systems* 1, 1 (2023), 1–51.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. 2019. Combining Neural Networks with Personalized PageRank for Classification on Graphs. In *International Conference on Learning Representations*. https://openreview.net/forum?id=H1gL-2A9Ym

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR, 1263–1272.

Sujatha Das Gollapalli and Cornelia Caragea. 2014. Extracting keyphrases from research papers using citation networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 28.

Karan Goyal, Mayank Goel, Vikram Goyal, and Mukesh Mohania. 2024. SymTax: Symbiotic Relationship and Taxonomy Fusion for Effective Citation Recommendation. In *Findings of the Association for Computational Linguistics: ACL 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 8997–9008. https://doi.org/10.18653/v1/2024.findings-acl.533

Nianlong Gu, Yingqiang Gao, and Richard HR Hahnloser. 2022. Local citation recommendation with hierarchical-attention text encoder and scibert-based reranking. In *European Conference on Information Retrieval*. Springer, 274–288.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

Dongxiao He, Chundong Liang, Huixin Liu, Mingxiang Wen, Pengfei Jiao, and Zhiyong Feng. 2022. Block modeling-guided graph convolutional neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 36. 4022–4029.

Mingguo He, Zhewei Wei, Hongteng Xu, et al. 2021. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in Neural Information Processing Systems* 34 (2021), 14239–14251.

Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. 2010. Context-aware citation recommendation. In *Proceedings of the 19th international conference on World wide web*. 421–430.

Keke Huang, Yu Guang Wang, Ming Li, and Pietro Lio. 2024. How Universal Polynomial Bases Enhance Spectral Graph Neural Networks: Heterophily, Over-smoothing, and Over-squashing. In *Forty-first International Conference on Machine Learning*. https://openreview.net/forum?id=Z2LH6Va7L2

Qian Huang, Horace He, Abhay Singh, Ser-Nam Lim, and Austin Benson. 2021. Combining Label Propagation and Simple Models out-performs Graph Neural Networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=8E1-f3VhX1o

Wenyi Huang, Zhaohui Wu, Chen Liang, Prasenjit Mitra, and C Giles. 2015. A neural probabilistic model for context based citation recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29.

Glen Jeh and Jennifer Widom. 2002. SimRank: a measure of structural-context similarity. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Edmonton, Alberta, Canada) *(KDD '02)*. Association for Computing Machinery, New York, NY, USA, 538–543. https://doi.org/10.1145/775047.775126

Chanwoo Jeong, Sion Jang, Eunjeong Park, and Sungchul Choi. 2020. A context-aware citation recommendation model with BERT and graph convolutional networks. *Scientometrics* 124 (2020), 1907–1922.

Wei Jin, Tyler Derr, Yiqi Wang, Yao Ma, Zitao Liu, and Jiliang Tang. 2021. Node similarity preserving graph convolutional networks. In *Proceedings of the 14th ACM international conference on web search and data mining*. 148–156.

Rob Johnson, Anthony Watkinson, and Michael Mabe. 2018. The STM report. *An overview of scientific and scholarly publishing. 5th edition October* (2018), 94.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, Vol. 1. 2.

Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SJU4ayYgl

Soo Yong Lee, Fanchen Bu, Jaemin Yoo, and Kijung Shin. 2023. Towards deep attention in graph neural networks: Problems and remedies. In *International Conference on Machine Learning*. PMLR, 18774–18795.

Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023. Making Large Language Models A Better Foundation For Dense Retrieval. arXiv:2312.15503 [cs.CL]

Xiang Li, Renyu Zhu, Yao Cheng, Caihua Shan, Siqiang Luo, Dongsheng Li, and Weining Qian. 2022. Finding Global Homophily in Graph Neural Networks When Meeting Heterophily. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 162)*, Kamalika Chaudhuri, Stefanie

Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (Eds.). PMLR, 13242–13256. https://proceedings.mlr.press/v162/li22ad.html

Ningyi Liao, Siqiang Luo, Xiang Li, and Jieming Shi. 2023. LD2: Scalable Heterophilous Graph Neural Network with Decoupled Embeddings. In *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Eds.), Vol. 36. Curran Associates, Inc., 10197–10209. https://proceedings.neurips.cc/paper_files/paper/2023/file/206191b9b7349e2743d98d855dec9e58-Paper-Conference.pdf

Derek Lim, Felix Hohne, Xiuyu Li, Sijia Linda Huang, Vaishnavi Gupta, Omkar Bhalerao, and Ser Nam Lim. 2021. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Advances in Neural Information Processing Systems* 34 (2021), 20887–20902.

Haoyu Liu, Ningyi Liao, and Siqiang Luo. 2024. SIGMA: Similarity-based Efficient Global Aggregation for Heterophilous Graph Neural Networks. arXiv:2305.09958 [cs.LG] https://arxiv.org/abs/2305.09958

Meng Liu, Hongyang Gao, and Shuiwang Ji. 2020. Towards Deeper Graph Neural Networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Virtual Event, CA, USA) *(KDD '20)*. Association for Computing Machinery, New York, NY, USA, 338–348. https://doi.org/10.1145/3394486.3403076

Avishay Livne, Vivek Gokuladas, Jaime Teevan, Susan T Dumais, and Eytan Adar. 2014. CiteSight: supporting contextual citation recommendation using differential search. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 807–816.

Sitao Luan, Chenqing Hua, Qincheng Lu, Jiaqi Zhu, Mingde Zhao, Shuyuan Zhang, Xiao-Wen Chang, and Doina Precup. 2022. Revisiting heterophily for graph neural networks. *Advances in neural information processing systems* 35 (2022), 1362–1375.

Sitao Luan, Chenqing Hua, Minkai Xu, Qincheng Lu, Jiaqi Zhu, Xiao-Wen Chang, Jie Fu, Jure Leskovec, and Doina Precup. 2023. When Do Graph Neural Networks Help with Node Classification? Investigating the Homophily Principle on Node Distinguishability. In *Thirty-seventh Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=kJmYu3Ti2z

Yao Ma, Xiaorui Liu, Neil Shah, and Jiliang Tang. 2022. Is Homophily a Necessity for Graph Neural Networks?. In *International Conference on Learning Representations*. https://openreview.net/forum?id=ucASPPD9GKN

Zoran Medić and Jan Šnajder. 2020. Improved local citation recommendation based on context enhanced with global information. In *Proceedings of the first workshop on scholarly document processing*. 97–103.

Qiaozhu Mei, Jian Guo, and Dragomir Radev. 2010. Divrank: the interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1009–1018.

Laurent Meunier, Raphael Ettedgui, Rafael Pinot, Yann Chevaleyre, and Jamal Atif. 2022. Towards Consistency in Adversarial Classification. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 8538–8549. https://proceedings.neurips.cc/paper_files/paper/2022/file/38d6af46cca4ce1f7d699bf11078cb84-Paper-Conference.pdf

Gabriela F Nane, Nicolas Robinson-Garcia, François van Schalkwyk, and Daniel Torres-Salinas. 2023. COVID-19 and the scientific publishing system: growth, open access and scientific fields. *Scientometrics* 128, 1 (2023), 345–362.

Ping Ni, Xianquan Wang, Bing Lv, and Likang Wu. 2024. GTR: An explainable Graph Topic-aware Recommender for scholarly document. *Electronic Commerce Research and Applications* 67 (2024), 101439.

Malte Ostendorff, Nils Rethmeier, Isabelle Augenstein, Bela Gipp, and Georg Rehm. 2022. Neighborhood Contrastive Learning for Scientific Document Representations with Citation Embeddings. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 11670–11688. https://doi.org/10.18653/v1/2022.emnlp-main.802

Lawrence Page. 1999. *The PageRank citation ranking: Bringing order to the web*. Technical Report. Technical Report.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. 2020. Geom-GCN: Geometric Graph Convolutional Networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=S1e2agrFvS

Oleg Platonov, Denis Kuznedelev, Artem Babenko, and Liudmila Prokhorenkova. 2022. Characterizing graph datasets for node classification: Beyond homophily-heterophily dichotomy. *arXiv preprint arXiv:2209.06177* (2022).

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. 2023. A critical look at the evaluation of GNNs under heterophily: Are we really making progress?. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=tJbbQfw-5wv

Chendi Qian, Andrei Manolache, Christopher Morris, and Mathias Niepert. 2024. Probabilistic Graph Rewiring via Virtual Nodes. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=LpvSHL9lcK

Jiezhong Qiu, Jian Tang, Hao Ma, Yuxiao Dong, Kuansan Wang, and Jie Tang. 2018. DeepInf: Social Influence Prediction with Deep Learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom) *(KDD '18)*. Association for Computing Machinery, New York, NY, USA, 2110–2119. https://doi.org/10.1145/3219819.3220077

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.

Emanuele Rossi, Bertrand Charpentier, Francesco Di Giovanni, Fabrizio Frasca, Stephan Günnemann, and Michael M. Bronstein. 2024. Edge Directionality Improves Learning on Heterophilic Graphs. In *Proceedings of the Second Learning on Graphs Conference (Proceedings of Machine Learning Research, Vol. 231)*, Soledad Villar and Benjamin Chamberlain (Eds.). PMLR, 25:1–25:27. https://proceedings.mlr.press/v231/rossi24a.html

Ronald Rousseau, Carlos Garcia-Zorita, and Elias Sanz-Casado. 2023. Publications during COVID-19 times: An unexpected overall increase. *Journal of Informetrics* 17, 4 (2023), 101461.

Ramit Sawhney, Ritesh Soun, Shrey Pandit, Megh Thakkar, Sarvagya Malaviya, and Yuval Pinter. 2022. Ciaug: Equipping interpolative augmentation with curriculum learning. In *NAACL*. 1758–1764.

Jianbo Shi and J. Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 8 (2000), 888–905. https://doi.org/10.1109/34.868688

Nazar Shmatko, Alex Adam, and Paul Esau. 2026. GPTZero Finds 100 New Hallucinations in NeurIPS 2025 Accepted Papers. *GPTZero* (21 Jan. 2026). https://gptzero.me/news/neurips/

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. 2022. SciRepEval: A Multi-Format Benchmark for Scientific Document Representations. In *Conference on Empirical Methods in Natural Language Processing*. https://api.semanticscholar.org/CorpusID:254018137

Susheel Suresh, Vinith Budde, Jennifer Neville, Pan Li, and Jianzhu Ma. 2021. Breaking the Limit of Graph Neural Networks by Improving the Assortativity of Graphs with Local Mixing Patterns. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (Virtual Event, Singapore) *(KDD '21)*. Association for Computing Machinery, New York, NY, USA, 1541–1551. https://doi.org/10.1145/3447548.3467373

Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. 2009. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 807–816.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. 2022. Understanding over-squashing and bottlenecks on graphs via curvature. In *International Conference on Learning Representations*. https://openreview.net/forum?id=7UmjRGzp-A

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations*. https://openreview.net/forum?id=rJXMpikCZ

Daixin Wang, Jianbin Lin, Peng Cui, Quanhui Jia, Zhen Wang, Yanming Fang, Quan Yu, Jun Zhou, Shuang Yang, and Yuan Qi. 2019. A semi-supervised graph attentive network for financial fraud detection. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 598–607.

Tao Wang, Di Jin, Rui Wang, Dongxiao He, and Yuxiao Huang. 2022a. Powerful graph convolutional networks with adaptive propagation mechanism for homophily and heterophily. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 36. 4210–4218.

Yifan Wang, Yiping Song, Shuai Li, Chaoran Cheng, Wei Ju, Ming Zhang, and Sheng Wang. 2022b. DisenCite: Graph-Based Disentangled Representation Learning for Context-Specific Citation Generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 11449–11458.

Yuelin Wang, Kai Yi, Xinliang Liu, Yu Guang Wang, and Shi Jin. 2023. ACMP: Allen-Cahn Message Passing with Attractive and Repulsive Forces for Graph Neural Networks. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=4fZc_79Lrqs

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*. PMLR, 6861–6871.

Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. 2022. Graph neural networks in recommender systems: a survey. *Comput. Surveys* 55, 5 (2022), 1–37.

Qianqian Xie, Yutao Zhu, Jimin Huang, Pan Du, and Jian-Yun Nie. 2021. Graph neural collaborative topic model for citation recommendation. *ACM Transactions on Information Systems (TOIS)* 40, 3 (2021), 1–30.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations*. https://openreview.net/forum?id=ryGs6iA5Km

Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. 2018. Representation Learning on Graphs with Jumping Knowledge Networks. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 5453–5462. https://proceedings.mlr.press/v80/xu18c.html

Zhe Xu, Yuzhong Chen, Qinghai Zhou, Yuhang Wu, Menghai Pan, Hao Yang, and Hanghang Tong. 2023. Node Classification Beyond Homophily: Towards a General Solution. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 2862–2873. https://doi.org/10.1145/3580305.3599446

Chenxiao Yang, Qitian Wu, Jiahua Wang, and Junchi Yan. 2023. Graph Neural Networks are Inherently Good Generalizers: Insights by Bridging GNNs and MLPs. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=dqnNW2omZL6

Tianmeng Yang, Yujing Wang, Zhihan Yue, Yaming Yang, Yunhai Tong, and Jing Bai. 2022. Graph pointer neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 36. 8832–8839.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176* (2025).

Xin Zheng, Yixin Liu, Shirui Pan, Miao Zhang, Di Jin, and Philip S Yu. 2022. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082* (2022).

Xin Zheng, Miao Zhang, Chunyang Chen, Qin Zhang, Chuan Zhou, and Shirui Pan. 2023. Auto-HeG: Automated Graph Neural Network on Heterophilic Graphs. In *Proceedings of the ACM Web Conference 2023* (Austin, TX, USA) *(WWW '23)*. Association for Computing Machinery, New York, NY, USA, 611–620. https://doi.org/10.1145/3543507.3583498

Jiong Zhu, Ryan A Rossi, Anup Rao, Tung Mai, Nedim Lipka, Nesreen K Ahmed, and Danai Koutra. 2021. Graph neural networks with heterophily. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 11168–11176.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. 2020. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems* 33 (2020), 7793–7804.