RACP: Retrieval Augmented Connected Papers

Tao Yicheng Wu Haoyu

Date: December 14, 2023

1 Introduction

After conducting research on commonly used literature search platforms such as arXiv, Google Scholar, ConnectedPapers, Papers with Code, etc., we have observed a lack of balance between accuracy and diversity in their search results, often failing to effectively meet users' information needs. Accuracy refers to the degree to which the retrieved literature fulfills users' information requirements, while diversity pertains to the proportion of literature that satisfies users' information needs among all potentially relevant documents.

Several factors contribute to this deficiency. The complexity and abundance of scientific terminology pose challenges, and the intricate content of research papers surpasses the limitations of field and keyword classifications in providing sufficient information. Consequently, we aim to integrate existing language models with two mechanisms: enhanced comprehension of lengthy texts and intelligent retrieval. This integration seeks to achieve a balance between the accuracy and diversity of literature search results, striving to accurately identify previously unseen and valuable documents for users.

For instance, in the fields of computer vision (CV) and computer graphics, terms like "texture" and "albedo" are used to denote different aspects of 3D models. Our tests on various literature search platforms revealed a common limitation: if a user inputs "texture," the platforms typically do not return literature related to "albedo." Such shortcomings pose obstacles to effective literature research.

In summary, we propose a solution that leverages language models to enhance long-text comprehension and intelligent retrieval mechanisms. This aims to address the imbalances in accuracy and diversity in literature search results, facilitating the accurate retrieval of previously undiscovered, valuable literature for users.

2 Method

2.1 Dataset

We construct our dataset by crawling arXiv papers under the tag cs.AI from 2019-2023. The full text content is provided by arXiv[1] while the citation data and other metadata are provided by Semantics Scholar[2].

Our dataset consists of 57821 papers. Each item contains entries as Table 1.

Entry	Meaning
arxivId	The arXiv id of the paper.
paperId	The semantics scholar id of the paper.
citations	A list of semantics scholar ids which cite this paper.
references	A list of semantics scholar ids which this paper cites(null if not included in semantics scholar database).
authors	This paper's author's id, name, citation counts and paper counts.
publication	The publication type of this paper.
date	The publication date of this paper.
title	The title of this paper.
abstract	The abstract of this paper.
content	The raw text extracted from PDF file.

Table 1: All the entries except content are provided by semantics scholar database.

The papers were chosen by their arXiv tag and publication date. Since the arXiv publication date may not be the real publication date, we use the data from Semantics Scholar. Thus some of the papers are published before 2019. Each paper can have multiple publication type, including journal article, conference, book, review and so on. For detailed information, please refer to Fig 1 and Fig 2.

Including the papers referred to or cited by the papers in our dataset, the total number of papers related to our dataset is 948517. And we have 137889 authors with their citations counts.

The citations of the papers are expected to satisfy power-law distribution. That is to say, if X denotes the citation count of a paper, then $P(X) \sim X^{-\alpha}$. We use Python package **powerlaw**[3] to fit the citation counts of all related papers. The **powerlaw** package automatically calculates the best min value for the x axis, represented as xmin, and ignores the values less than xmin. The result is shown in Fig 3.

[1] https://arxiv.org/

[2] https://www.semanticscholar.org/

[3] https://pypi.org/project/powerlaw/

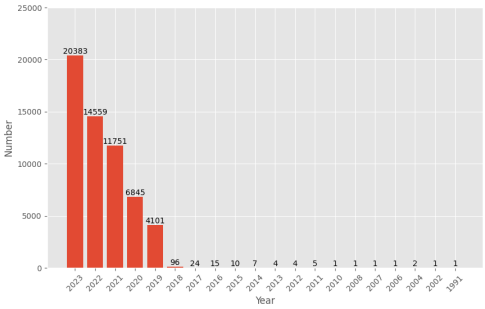

Figure 1: The distribution of publication years of the dataset. Note that the crawling process identifies the years by arXiv ids, but the data showed here is from Semantics Scholar.

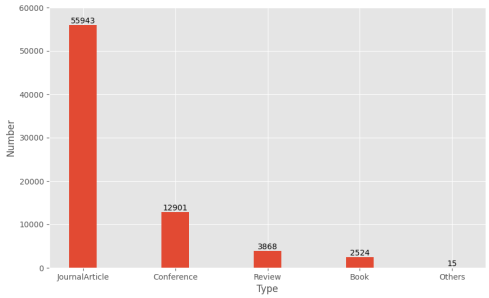

Figure 2: The distribution of publication types of the dataset. The "Others" contains News, LettersAndComments, CaseReport, Editorial, MetaAnalysis, Dataset.

2.2 Retriever

2.2.1 weighted CCBC function

We first re-implemented Co-citation and Bibliographic Coupling(CCBC) function. According to CCBC function, two papers that have highly overlapping citations and references are presumed to have a higher chance of treating a related subject matter.

In the common citations of the same paper in two different articles, each article assigns a distinct level of importance to the shared reference. For instance, two papers citing ResNet may belong to disparate domains such as object detection and natural language processing, and yet, the papers themselves may lack inherent relevance. This is because these papers hold foundational significance, reflected in their exceptionally high citation counts.

Therefore, among the references concurrently cited by two documents, those with lower overall citation counts tend to better indicate the relevance between the two articles.

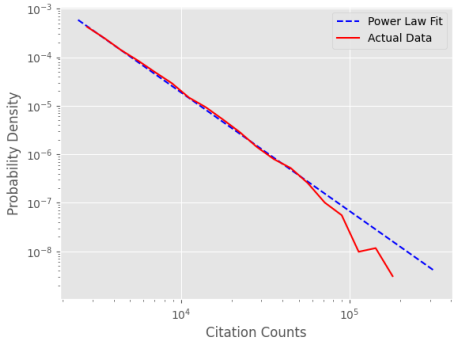

Figure 3: Fit using 948517 papers' citations. $\alpha \approx 2.4486$. xmin $= 2460$.

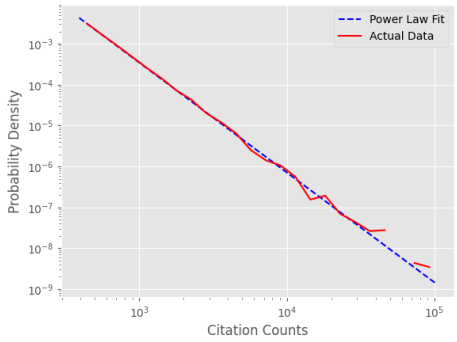

Figure 4: Fit using 5980304 papers' citations from S2AG. $\alpha \approx 2.6894$, xmin $= 393$.

To formalize the question, let A and B denote two papers, $c(A)$ and $c(B)$ denote their number of citations. Let $I(\cdot)$ denotes characteristic function. Let refer(A) denotes the papers referred by A, cite(B) denotes the papers which cite A. Then a naive version of CCBC is Eq 1.

$$\text{CCBC}(A,B) = \frac{2}{5}\left(\frac{1}{2}I(\text{A cites B or B cites A}) + \frac{|\text{refer}(A) \cap \text{refer}(B)|}{|\text{refer}(A) \cup \text{refer}(B)|} + \frac{|\text{cite}(A) \cap \text{cite}(B)|}{|\text{cite}(A) \cup \text{cite}(B)|}\right) \tag{1}$$

The coefficient $\frac{2}{5}$ is used to control the range of CCBC.

To avoid the bias mentioned above, we assign a weight to each paper according to their number of citations. The more citations of the paper, the lower weight it gets. Here we treat the number of citations as a random variable. We know it satisfies the power-law distribution. So we use its survival function as the paper's weight.

We use a larger set of citations to fit this power-law distribution, which is 5980304 papers from S2AG[4]. The result is shown in Fig 4. After fitting the cumulative distribution function(cdf) of the citations, we can calculate the weight of paper A can as Eq 2.

$$w(A) = 1 - \text{cdf}(c(A)) \tag{2}$$

By adding weight to the naive version of CCBC, we get the weighted CCBC function as Eq 3.

$$\begin{aligned}\text{CCBC}(A,B) = &\frac{1}{6}w(A)I(\text{B cites A}) + \frac{1}{6}w(B)I(\text{A cites B}) \\ &+ \frac{1}{3}\left(\frac{\sum\limits_{C \in \text{refer}(A) \cap \text{refer}(B)} w(C)}{\sum\limits_{C \in \text{refer}(A) \cup \text{refer}(B)} w(C)}\right) \\ &+ \frac{1}{3}w(A)w(B)\frac{|\text{cite}(A) \cap \text{cite}(B)|}{|\text{cite}(A) \cup \text{cite}(B)|}\end{aligned} \tag{3}$$

2.2.2 Semantic Relevance Retrieval

Connected Papers is built upon the principles of Co-citation and Bibliographic Coupling (CCBC), relying solely on citation information. However, we have observed that Connected Papers may fall short in identifying inspiring papers that are not closely related. To address this limitation, we have introduced a Semantic Embedding Module to provide detailed information beyond the citation structure.

For each paper in the constructed database, we initially extract a summary of the entire paper and compute a latent feature vector $F \in \mathbb{R}^{768}$ for it. Subsequently, we store all these vectors in a vector store for retrieval purposes.

We retrieve the most similar paper in the semantic latent space using the following formula:

[4] https://api.semanticscholar.org/api-docs/datasets

$$\text{Similarity}(A, B) = \frac{A \cdot B}{\|A\|\|B\|}$$

Here, A represents the latent feature vector of the query paper, and B represents the latent feature vector of potential retrieved papers. The $\cdot$ denotes the dot product, and $\|\cdot\|$ signifies the Euclidean norm.

This approach enhances the capability of Connected Papers to discover papers with a higher degree of semantic relevance.

2.2.3 Confidence Quality Re-range

After select a set of optional papers, we will evaluate confidence quality score because the papers on the arxiv website are not all reviewed by peers. So we build a function that thoroughly consider the overall confidence quality using citation amount, citation increasing speed, authors' previous publication history and timeliness.

3 Implementation details

Our semantic embedding model uses mpnet [1] as default option. Other embedding models from transformers are also supported. To enable large scale of database, we uses chroma as vector database backend.

4 Conclusion

Our system can properly leverage semantic information that is ignored by previous academic paper retrieval systems. However the semantic embedding module are not fully fine-tuned to gain maximal embedding ability. Our future work is to train a more powerful semantic embedding model and provide more retrieval policies.

Bibliography

[1] Mahnaz Koupaee and William Yang Wang. "Wikihow: A large scale text summarization dataset". In: *arXiv preprint arXiv:1810.09305* (2018).