Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Fachbereich Informatik
Department of Computer Science

Master Project

Computer Science

LitQEval: Measuring the Effectiveness of Litreature Search Queries

Computer Science

Mohammad Sakinini

Supervisor	Philipp Baaden Fraunhofer INT
Examiner 1	Prof. Dr. Jörn Hees
Examiner 2	Dr. Milos Jovanovic
Draft as of	2024-12-02 15:49:46+01:00 (For submission: set `final` option in thesis.tex!)
To be submitted on	2024-07-26

Abstract

This work is based on a larger initiative known as the Search Query Writer (SQW), an internal tool developed at Fraunhofer INT to aid scientific researchers in creating comprehensive literature search queries. These queries are intended to provide researchers with a strong starting point in a topic area they may have limited knowledge about.

The current state of the SQW tool presents a key challenge: the absence of a mechanism to evaluate the quality of the generated queries. As a result, the evaluation has so far been conducted subjectively. This project aims to address this issue by introducing a dataset that contains publications deemed relevant to specific topics. Additionally, it introduces several metrics to account for different aspects of query evaluation, given the complexity of the task.**(Explain performed experiments after completing them)**

Contents

List of Figures

1. Introduction

The Fraunhofer Institute for Technological Trend Analysis (INT)[1] is a research institute that frequently undertakes new tasks and research questions across various fields. Often, these inquiries require systematic and robust scientific responses, even when the initial knowledge in the area may be limited at the time. Given the recurring nature of this challenge, a tool that can support researchers by providing a head start and entry point into unfamiliar fields is essential.

To address this, several internal tools have been developed to analyze large volumes of scientific data from sources like Dimensions.ai[2] and Web of Science[3]. The rise of large language models (LLMs) has further enhanced the appeal and accessibility of automation across numerous domains, including scientific research, spanning from idea generation and experimental iteration to paper composition[6].

In the realm of search queries, the main focus has been on text-to-SQL[2], where an LLM is prompted via natural language to generate a precise and valid SQL query. However, to our knowledge, there has been limited and non-diverse effort dedicated to the development of text-to-literature search queries. Thus this work introduces a pipeline and curates a dataset designed to address this gap, with a particular focus on enhancing the Fraunhofer Search Query Writer tool.

1.1. Motivation

The SQW tool is currently under development by the company and has generated significant interest among researchers. However, a primary challenge we face after testing earlier versions is evaluating the quality of the generated queries. Initially, we considered gathering human feedback from users by requesting them to rate the generated query on a scale of 0 to 5. While this approach could be useful for fine-tuning the underlying model, the quantity of feedback has so far been limited and remains subjective. This is especially problematic because the tool's purpose is to generate queries for researchers who are new to a given topic. Consequently, if the query quality is poor, the researcher may not immediately recognize this.

Identifying suitable evaluation metrics and datasets to assess the quality of the generated queries is a complex task, which forms the basis of this master's project. The project's objective is to find a robust solution for assessing the quality of literature search queries, enabling the further development of the SQW tool to

[1] https://www.int.fraunhofer.de/
[2] https://www.dimensions.ai/
[3] https://clarivate.com/

provide more accurate results and improve productivity through the integration of large language models.

1.2. Research Questions

There are three main research questions that we aim to address while curating the dataset and formulating the metrics to evaluate the generated queries. These questions are based on the following hypothesis: Given that we know which publications are the most important for a given field, which we will refer to as **Core Publications** (CP).

- **RQ1:** How many of the core publications can the generated search query recall?
- **RQ2:** How many of the non-core publications are relevant?
- **RQ3:** Which metric can we use to penalize the model for exploiting the ability to generate large queries to achieve high recall?

1.3. Structure of this Work

The remainder of this work is structured as follows:

After this introduction, we will first focus on the foundations in Chapter 2, where we will briefly explain the SQW tool, primarily focusing on the format of the input and output. We will also cover the necessary basics and information about the main data source, Dimensions.ai, which will be used to curate the dataset **??**, as well as common metrics **??** used for evaluation in the community, before discussing related works in Section 2.2.

Next, in Chapter 3, we will detail our approach and its components, which involve curating a dataset that contains core publications and establishing a pipeline to streamline the evaluation process, thereby accelerating the development of the SQW tool.

We will evaluate our approach in Chapter 4, beginning with a description of our experimental setup, where we acquire the generated query via the SQW tool and explain the reasoning behind the selection of topics for which the queries are generated.

Finally, we will conclude this work with a summary of our main contributions and an outlook in Chapter 5.

2. Foundations

In this section, we will begin by briefly introducing the SQW tool to provide a foundational understanding of how its settings may influence the overall results. This introduction will also establish the groundwork for designing an evaluation process that ensures a fair and accurate assessment.

Next, we will review prior works that tackle the challenge of using large language models (LLMs) to generate literature search queries, examining their potential in this domain.

Following this, we will describe the dataset curation process, including relevant statistics and exploratory data analysis. This analysis will help us evaluate the dataset's quality and determine the topics suitable for evaluation.

Then, we will introduce new metrics designed to assess the quality of generated search queries. These metrics are intended not only to identify true positives from core publications but also to account for other potentially relevant publications. Once the evaluation pipeline is established, we will conduct a comparative analysis between a baseline and the queries generated by the SQW tool.

Finally, we will provide an outlook on the next steps and potential optimization options for the SQW tool. Additionally, we will discuss other tools that could be developed to build upon these advancements.

2.1. Search Query Writer

The Search Query Writer is a tool based on a large language model (LLM), specifically using GPT-4o, to systematically generate literature search queries. The only required input for this tool, which is the main focus of this work, is the **Topic**. Users are required to provide a topic for generating a search query, irrespective of the scientific field—for example, *Synthetic Biology*.

Several optional inputs are available to enhance the quality of the generated query, including:

- **Negative Keywords:** Terms that should be excluded to avoid unwanted results.
- **Description:** A description that serves as an alignment mechanism to clarify the task's intent.
- **Modes:** Three selectable modes (Strict, Moderate, Creative) that control the temperature of the LLM to manage the level of randomness in responses.

- **Depth:** A parameter that specifies how comprehensively the topic should be analyzed.
- **Supporting Documents:** Users can upload a PDF, ideally a survey or overview document on the topic, which helps the tool acquire knowledge about the scientific field and better align with the research intent.

These additional inputs are intended to refine and tailor the search query to more closely match the user's research goals, but will not be extensively tested in this work.

To generate a literature search query, we designed the SQW to take a human-like approach, divided into two main steps: **Knowledge Enrichment** and **Iterative Scientific Fine-Tuning**.

The objective of the Knowledge Enrichment step is to provide the LLM with contextual information about the research topic. This is achieved by first retrieving information from Wikipedia based on the given topic. Specifically, the first 4,000 characters from the top-k pages are collected and summarized before being passed into the LLM's memory. ArXiv is queried in a similar manner to gather relevant research content. Additionally, we perform an online search using DuckDuckGo[1], aggregating results to offer a broader understanding of the topic.

Notably, each of the steps is conducted within a separate memory session, with results stored independently for future use. This setup allows the model to explore the topic using various sources, helping to mitigate any potential recency bias and ensure a well-rounded context.

The output of this first stage will be a list of keywords that are usually presented in a transfer-list as shown in Figure A.1, that contains two lists; specific and general keywords, along side some additional information such as the number of publications found per keyword. The goal is to let the user decide weather a keyword is too broad in which he is supposed to move it to the general list, and if it targets the specified topic quite well then it should stay in the specific list, and at the end they user should also provide an overarching topic for which the scope of the general keywords is limited to be more focused to words the research intent. The output of this step will be the queries that will be used for the evaluation at the end.

The iterative scientific fine-tuning on the other hand approaches more scientific sources, namely dimensions.ai, which is a literature database that offers quick access to publications across a wide range of journals. The query generated in the earlier stage is then used to prompt dimensions three times, once to retrieve the most cited 1k literature, a second time to retrieve the newest 1k literature, and one last 1k to retrieve the most relevant documents based on their altmetric rating. This leaves us with a total of 3k publications in which we extract the title and abstract for, and use an Openai's embedding model, and apply a simple RAG pipeline to retrieve the most relevant keywords based on the extracted passages.

[1] https://duckduckgo.com/

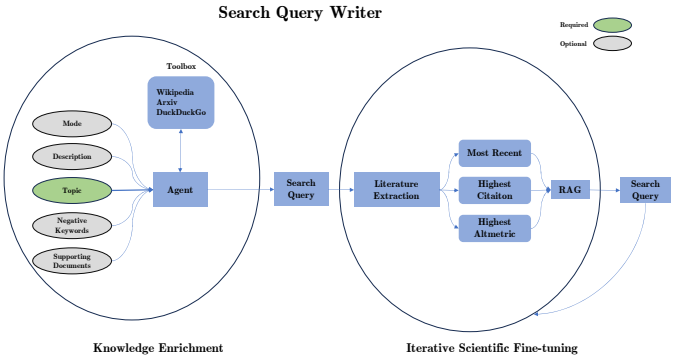

Figure 2.1.: A simplified overview of the Search Query Writer. The process begins with the Knowledge Enrichment stage, where the model receives input data and sends it to an LLM agent equipped with a suite of tools to gain insights into the topic. Based on this understanding, the model generates a well-structured search query, formatted and executed across multiple dimensions to retrieve a relevant selection of literature. Within this literature set, RAG is applied to identify the most pertinent keywords, which are compiled into an optimized search query. This query can be iteratively refined to enhance overall search quality.

2.2. Related Work

Systematic literature reviews are widely used across various fields, allowing researchers to conduct a comprehensive manual review of scientific topics and identify publications that answer a set of important research questions. However, one significant challenge with this approach has been the exponential growth in the number of publications, which makes conducting unbiased reviews increasingly difficult. In the age of technological advancements, we can now leverage these technologies to assist in investigating topics without the need to manually sift through extensive lists of publications. To address this issue, a series of works have been proposed within the Conference and Labs of the Evaluation Forum (CLEF) [3–5]. These works focus on the evaluation of empirical medical research, utilizing a dataset of medical literature. They introduce two primary tasks: Task 1, which involves identifying relevant studies from the PubMed medical database, and Task 2, which assesses the ranking of studies following title and abstract screening. Notably, the evaluation pipeline, along with the dataset and descriptions of these tasks, are publicly accessible on GitHub[2].

Large Language Models (LLMs) have had a significant impact on modern technology, including in scientific research, where they have provided remarkable

[2] https://github.com/CLEF-TAR/tar/tree/master

improvements in speed. While the processing speed of LLMs is unprecedented, the quality of their output in various domains is still being explored. The work by Wang [11] investigated the performance of ChatGPT in generating Boolean search queries for literature reviews. Specifically, it evaluated the effectiveness of ChatGPT in constructing queries for systematic literature reviews using different prompting techniques, including zero-shot, few-shot and iterative guided approaches. The evaluation used the CLEF datasets [3–5] and an additional medical dataset containing a collection of seeds [10]. Although the results highlight the limitations of ChatGPT's performance, this work underscores the potential of LLMs to aid literature review, especially when supported by examples or more advanced, structured pipelines.

A broader and more diverse evaluation of the quality of automatically generated literature search queries for systematic literature reviews was conducted by Badami [1]. In this work, they introduced a pipeline that generates literature search queries based on a given research question and abstracts from previously identified relevant publications. The evaluation was performed against a dataset they constructed, which contains the results of 10 systematic literature reviews, including candidate papers, queries used, and relevant papers identified in each review. For example, in the review SLR_1, a total of 7,002 candidate papers were retrieved using search query S, from which a subset of 59 relevant papers RP was identified. To assess their proposed approach, they compared the generated queries in various settings using recall and precision metrics, benchmarking them against the original search query S. The dataset is publicly available on Zenodo[3].

[3] https://tinyurl.com/496zuar3

3. LitQEval

Despite ongoing research on automatic literature query generation and related evaluations with medical datasets, such as CLEF [3–5] and the Collection of Seeds [10], the insights gained from these evaluation metrics are not particularly compelling for our use case. This limitation arises from two main factors.

First, the CLEF and Collection of Seeds datasets are exclusively focused on medical data. Although Badami's work [1] offers a more diverse dataset, it lacks a suitable evaluation metric. Their evaluation primarily aims to maximize recall, with minimal consideration for precision, as literature search queries often yield far more results than necessary, making precision a less effective measure in this context.

A second limitation arises when recall is prioritized exclusively. For example, if we aim to train a model to generate queries that maximize recall, there is no penalty for generating overly broad queries, such as those that exploit wildcards, which could lead to an excessive number of irrelevant results.

To address these issues, we introduce a dataset structured similarly to that of Badami [1] but designed to be more comprehensive and covering a wider range of topics. Alongside this dataset, we propose new evaluation metrics that account for the inherently broad nature of literature search queries while penalizing excessively large queries. These metrics also emphasize the importance of accurately identifying core publications that are deemed highly relevant within the domain.

3.1. Dataset

The dataset we aim to create has three primary goals: First, it should encompass a wide range of randomly selected scientific research fields. Second, for each selected field, it should contain a set of highly relevant publications to serve as anchors for evaluating additional publications found in these areas. Lastly, the data should consider different research intents, meaning publication that is deemed relevant by a bibliometric analysis might not be relevant for a Systematic Literature Review (SLR) work.

Selecting new research topics is straightforward; however, to avoid bias from ongoing research interests, we used ChatGPT to generate a list of scientific fields that are recent and not overly broad. For instance, a topic like *Artificial Intelligence* is vast, making it challenging to accurately and comprehensively identify core publications. Instead, we chose a more specific, problem-focused topics such as *Drones in Agriculture*. To search for the corresponding bibliometric analysis

we used the following query: *<TERM> ("Bibliometric" OR "Scientometric" OR "Systematic literature" OR "Most Influential" OR "Most Cited" OR "Scientific Landscape" OR "Literature Landscape" OR "Core Literature")*

After identifying a sufficient number of diverse fields, 14 in our case, we sought to collect core publications for each field. Due to the difficulty of gathering core publications across a broad array of topics, we leveraged the bibliometrics community's expertise. Specifically, we searched for bibliometric studies that identify the most relevant publications within each research area. For example, a bibliometric analysis of *Drones in Agriculture* [8] lists the most cited publications from 1990 to 2021. In this case, 40 core publications were identified, which we manually located on Dimensions.ai and added to our dataset, omitting any publications not found in Dimensions.

This process was repeated across all selected research fields, resulting in a dataset comprising 14 topics, each containing 25–50 core publications, as shown in Figure 3.1. For the systematic literature review (SLR) data, we used previously collected data [1] in the field of Software Engineering. Notably, the SLR data used here were replicated by executing the original query in Dimensions. However, only 7 out of the 10 original datasets were included, with SLR 2, 5, and 6 omitted due to extreme variations between the original datasets and the results retrieved from Dimensions. For instance, SLR 2 originally contained 8,911 candidate papers, but when executed in Dimensions, it yielded approximately 200,000. Overall, the dataset consists of 21 topics.

3.1.1. Dataset Analysis

We recognize that potential biases may exist in our dataset due to its complete reliance on the bibliometric community for identifying core publications. This often implies that publications with higher citation counts are deemed more relevant. To assess this, we analyzed the citation distribution per topic, as provided by Dimensions, shown in Figure A.2. Additionally, we examined the distribution of publication years per topic, illustrating the time span considered in the bibliometric analyses, as shown in Figure A.3. If we compare the distribution of publication years for the medical research field *Cervical Myelopathy* with that of *IoT in Healthcare*, both of which were published in 2018, we can observe distinct differences in the year distributions of their core publications. These variations may be attributed to factors such as the recency of the field, changes in terminology over time, or the nature of the research area, where one field may prioritize more established works while the other focuses on recent advancements.

For the evaluation pipeline that we will introduce, the embeddings of the core documents are essential for effectively assessing the search query, as detailed in Section 3.2. To validate this approach, we examine the clustered embeddings of the titles and abstracts for each core topic, as well as the bibliometric analyses in which these documents were initially referenced. This enables us to assess whether core publications within each field exhibit semantic similarity while also

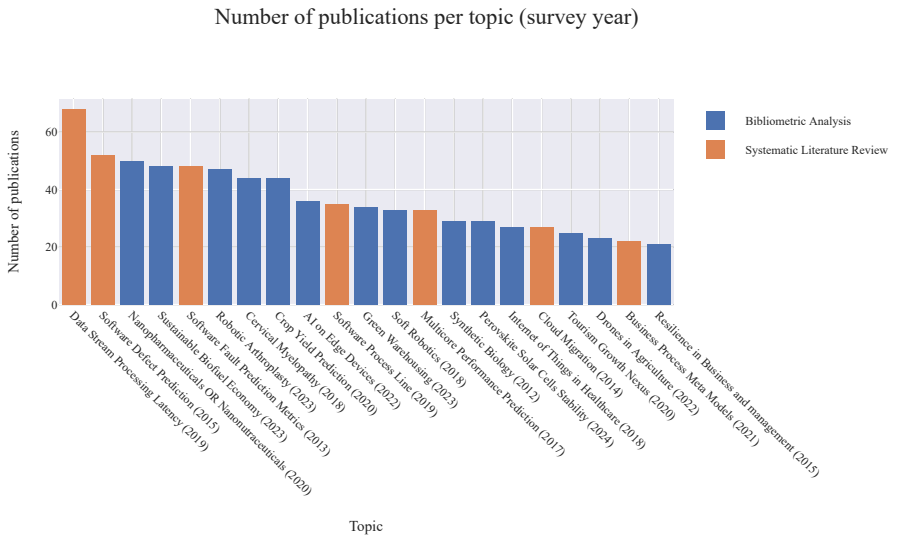

Figure 3.1.: An overview of the dataset and the selected 21 research fields with respective core publications identified through bibliometric analyses or systematic literature review. The number in brackets following the field name on the x-axis represents the year of survey publication.

demonstrating some degree of dissimilarity from publications in other topics. The resulting clusters, shown in Figure 3.2, were generated using k-means clustering, where k is set to the number of topics. For this, we use OpenAI's small embeddings model alongside t-SNE[9] to reduce the dimensionality to 2D.

3.2. Evaluation metrics

The standard evaluation metrics for query evaluation are recall and precision. We argue that while recall is of high importance, particularly within the community, precision in this context becomes less feasible. Specifically, retrieving only the exact core publications via a search query would be impractical without explicitly using DOIs to target them directly, which renders this metric largely obsolete and likely to be consistently low. However, we still aim to account for the number of matched publications when executing a search query to prevent models from exploiting overly large queries. To address this, we introduce the concept of *Semantic Precision*.

The idea behind Semantic Precision is to evaluate the relevance of retrieved publications in comparison to the core publication set. If the retrieved publications are sufficiently similar to those in the core set, they are deemed to hold some relevance rather than being entirely unrelated. To achieve this, we assume that the core publications, encompass sufficient semantic breadth to gauge the

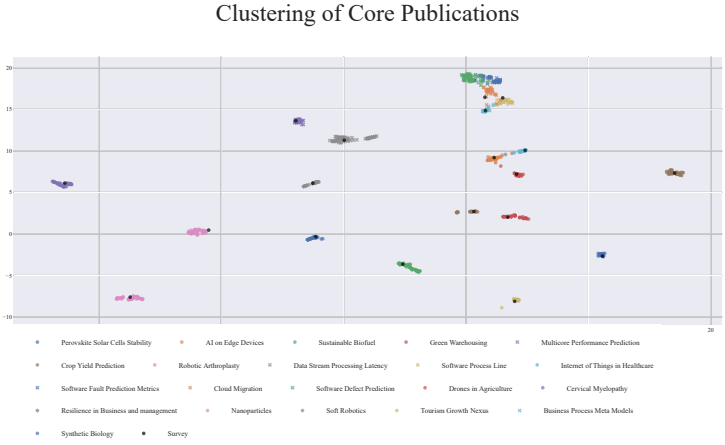

Figure 3.2.: This figure shows clusters of publication embeddings based on titles and abstracts, the ones marked with o are from the BAs and X from SLRs . Embeddings were generated with OpenAI's small model and reduced in dimensionality with UMAP, then clustered using k-means with $k = 21$ (indicating topic count). Clusters group core publications by semantic similarity, with overlaps in topics like *IoT in Healthcare* and *AI on Edge Devices*, as well as most of the SLR topic, due to the similarity in the research field.

quality of literature relevant to a specific field. We calculate Semantic Precision in three ways.

Semantic Cosine Precision

The first approach involves averaging the embeddings of the core publications. We then set an acceptance threshold based on the cosine similarity to the least similar core publication, given by. This means that if the embedding of a retrieved publication is more similar to the center than the least similar core publication, we consider it a relevant publication, as shown in Figure 3.3. We define:

- CPs as the set of core publications.
- $\mathbf{c_i}$ as the embedding vector of the i-th core publication.
- $\mathbf{p}$ as the embedding vector of a retrieved publication.
- $\cos(\mathbf{a}, \mathbf{b})$ as the cosine similarity between two vectors $\mathbf{a}$ and $\mathbf{b}$.

First, compute the centroid of the core publication embeddings:

$$\mathbf{c}_{\text{centroid}} = \frac{1}{|CPs|} \sum_{\mathbf{c_i} \in CPs} \mathbf{c_i}$$

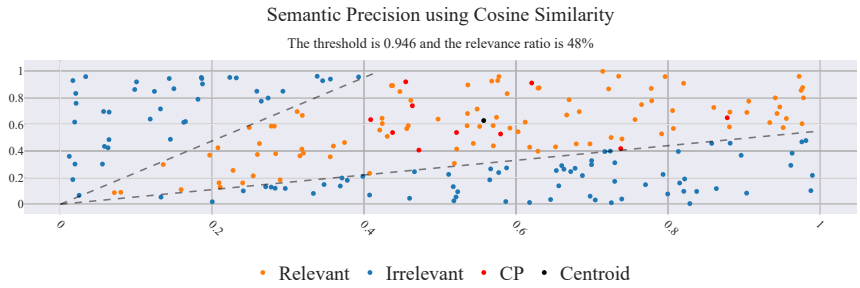

Figure 3.3.: This illustration demonstrates the effect of cosine similarity on randomly generated data in a 2D space. The core publications (CP) are shown in red, positioned between 0.4 and 0.9 on both the x- and y-axes. When we set the threshold to 0.946, based on the cosine similarity of the least similar core publication from the centroid, many retrieved publications on the opposite side of the spectrum are still assigned as relevant. This effect occurs because cosine similarity considers only the angle between vectors, ignoring their magnitude. In this case, this results in 48% of the retrieved publications being considered relevant.

Then, let the threshold similarity, θ, be the cosine similarity of the least similar core publication to the centroid:

$$\theta = \min_{\mathbf{c_i} \in CP} \cos(\mathbf{c}_{\text{centroid}}, \mathbf{c_i})$$

Finally, Semantic Precision using cosine similarity (SP_{cos}) is defined as Equation 3.1, where $\mathbb{I}$ is an indicator function that equals 1 if the retrieved publication $\mathbf{r}$ meets the similarity criterion and 0 otherwise:

$$SP_{cos} = \frac{\sum_{\mathbf{p} \in \text{pubs}} \mathbb{I}\left(\cos(\mathbf{c}_{\text{centroid}}, \mathbf{p}) \geq \theta\right)}{|\text{retrieved}|} \tag{3.1}$$

Semantic MVEE Precision

For the second approach we omit the averaging of the embeddings and use Minimum Volume Enclosing Ellipsoid (MMVE), which creates the smallest ellipsoid that includes the our CP, which we then use to determine which of the retrieved publications are relevant by checking whether they are within MMVE or not, as illustrated in Figure 3.4. This approach allows us to take into account all the dimensions by not only considering the angle but also the magnitude

The Minimum Volume Enclosing Ellipsoid (MVEE) for the core publication set CP is centered at δ with shape matrix A. To determine whether a retrieved publication $\mathbf{r}$ is relevant, we check if it lies within the ellipsoid by testing the following condition:

$$(\mathbf{p} - \delta)^T A (\mathbf{p} - \delta) \leq 1$$

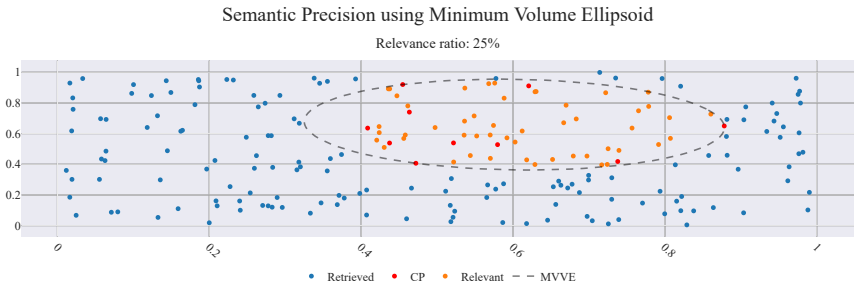

Figure 3.4.: This illustration demonstrates the effect of using the Minimum Volume Enclosing Ellipsoid (MVEE) on randomly generated data in a 2D space. The core publications (CP) are shown in red, positioned between 0.4 and 0.9 on both the x- and y-axes. An ellipsoid is generated using MVEE to define the scope of relevant publications, ensuring that only those within the maximal angles and magnitudes of the core publications are considered relevant. In this case, this approach results in only 25% of the retrieved publications being classified as relevant.

Semantic Precision (SP) for this approach is then:

$$\mathrm{SP}_{\mathrm{MVEE}} = \frac{\sum_{\mathbf{p} \in \mathrm{pubs}} \mathbb{I}\left((\mathbf{p} - \delta)^T A (\mathbf{p} - \delta) \leq 1\right)}{|\mathrm{pubs}|} \tag{3.2}$$

Additionally, it is also possible to use a convex hull, which is the smallest convex set that encloses all the points by forming a polygon. A potential advantage of this approach is that it is more robust to outliers compared to the Minimum Volume Enclosing Ellipsoid (MVEE).

Semantic Clustering Precision

For the final semantic precision approach, we apply a simple clustering algorithm, such as k-means, on the document embeddings. The process iteratively adjusts the number of clusters K, starting with $K = 2$, and increases K until a specific condition is met. We define a threshold θ that determines the stopping criterion based on the number of core publications in the smallest cluster. Specifically, we stop when the number of core publications (CPs) in the smallest cluster satisfies the condition:

$$CPs \text{ in cluster smallest cluster} \leq \theta \cdot \text{Maximum possible } CPs \tag{3.3}$$

This ensures that the smallest cluster contains at least θ of the core publications. All the above semantic precision metrics aim to identify potential true positives that were initially not considered as CPs. However, a key issue arises when the number of semantically relevant publications is large due to the broad scope of the initial query.

For instance, if a query retrieves 50,000 publications, with 30,000 deemed relevant, this still poses a challenge. Screening such a large volume of documents is infeasible, making the results problematic. To address this, we introduce a decay factor to the semantic precision, defined as follows:

- p: Controls the initial slowness of the decay.
- q: Controls the acceleration of the decay near the end.
- α: The maximum threshold for the decay, representing the point at which the decay becomes negligible.

The decay function is expressed as:

$$\lambda = \left(1 - \left(\frac{n_{\text{pubs}}}{\alpha}\right)^{p}\right)^{q}$$

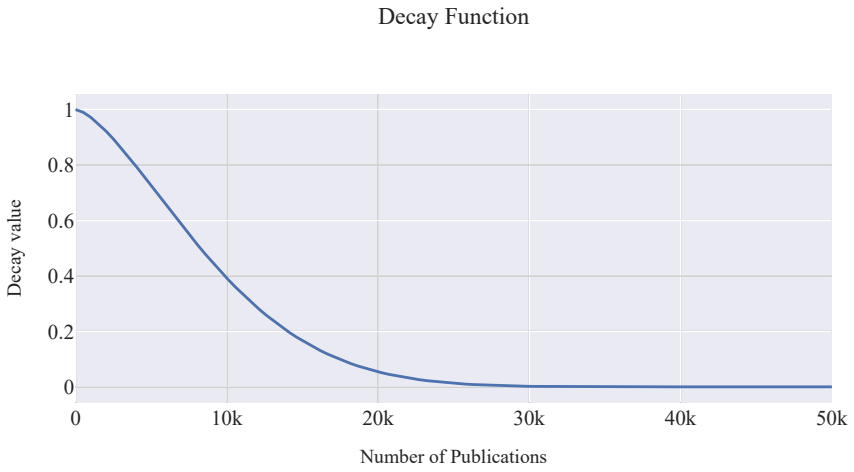

Figure 3.5.: This illustration demonstrates the effect of the decay factor, which ensures that the contribution of a large number of publications diminishes as the total count approaches the threshold. This prevents an overwhelming volume from biasing the semantic precision. For this example, we set the threshold (α) to 50k, $p = 1.5$, and $q = 10$.

Now that we have metrics that can be used penalizes the model in case of generating a too broad of a query, we use can use it as a factor to calculate the F-Score, the goal of the standard F-score is to balance out between the recall and precision, but in our case we use $F - \beta$ instead, whereby the β is the weighting factor of the recall, meaning the higher it is the more important the recall will be, in our case we set it to be 2, meaning that the recall is twice as important as

the precision.

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}} \tag{3.4}$$

For our specific case where $\beta = 2$, emphasizing the importance of recall, it is:

$$F_2 = 5 \cdot \frac{(\text{Precision} \cdot \lambda) \cdot \text{Recall}}{(4 \cdot \text{Precision} \cdot \lambda) + \text{Recall}}$$

3.2.1. Comparative Analysis

To further understand the metrics and their impact on evaluating the dataset, we conduct an in-depth analysis using a randomly selected topic, *Soft Robotics* and used its baseline query as a case study. First, we visualize the embeddings of the baseline and predicted queries Figure 3.6. The baseline query is the exact topic name, *Soft Robotics*, while the predicted query is generated by the SQW. The embeddings are derived from the title and abstract of the retrieved publications and subsequently reduced to a 2D UMAP [7] space. It is important to note that a significant amount of information is likely lost due to the extreme dimensionality reduction from 1536 dimensions to 2.

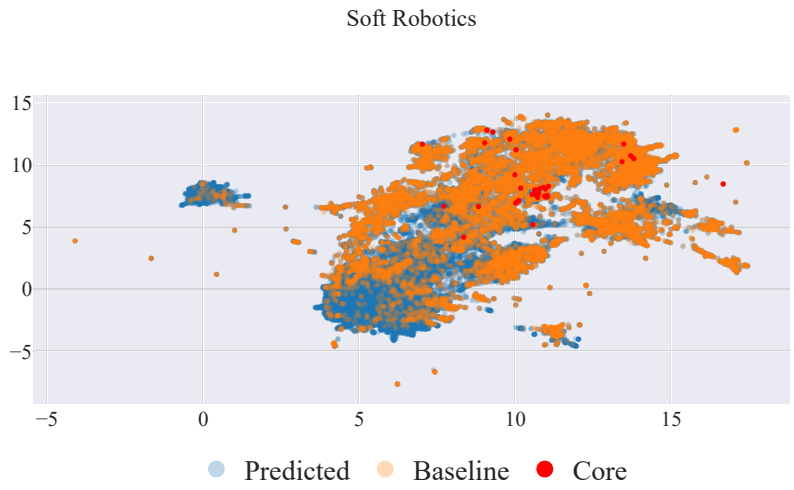

Figure 3.6.: This figure visualizes the distribution of publications retrieved by both the baseline and predicted queries in a 2D space. The baseline query retrieved 20 core publications, whereas the predicted query retrieved 26 core publications out of a total of 36.

Semantic Cosine Precision

At first, we test the Semantic Cosine Precision using the high-dimensional original embedding E_o, which was done as described in Equation 3.1. This resulted

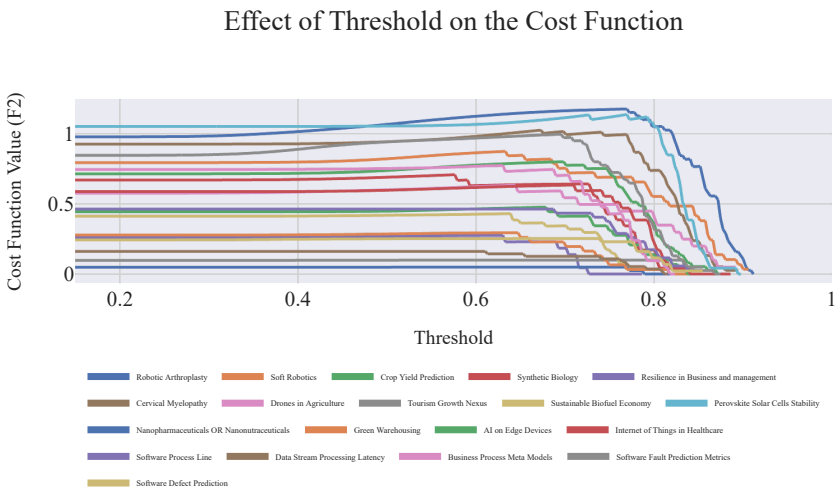

Figure 3.7.: This figure illustrates the effect of the threshold on the F_2 score. As the threshold increases, the number of semantically relevant publications and core publications identified decreases. However, in some cases, such as *Perovskite Solar Cells Stability*, the F_2 score continues to improve despite the loss of a core publication. This outcome is due to the F_2 score weighting recall twice as much as precision, allowing for stricter relevance criteria while sacrificing a single core publication.

in 13,265 out of 17,573 publications being classified as relevant Figure A.4. However, this high proportion of relevant publications appeared excessive, prompting further investigation into the threshold's effect on the number of semantically relevant documents.

Since we aim to use the F_2 score as our primary evaluation metric, we also employed it as a cost function to maximize. The goal was to identify the optimal empirical threshold that balances the retrieval of core publications with the number of relevant publications. To achieve this, we used the inverse precision, defined as $\frac{\text{Total Retrieved Publications}}{\text{Number of Relevant Publications}}$, instead of standard precision. The results Figure 3.7 reveal that a threshold maximizing the number of retrieved core publications while minimizing false positives is approximately 0.69. This suggests that sometimes sacrificing a couple of core publications is rewarding because it allows us to reduce the total number of semantically relevant publications.

After setting the threshold to the optimal empirical value, the Semantic Cosine Precision retrieves 19 out of the initially found 20 core publications while significantly reducing the number of semantically relevant publications by a factor of 4. This adjustment results in only 3,424 publications being identified as relevant, compared to the initial 13,265. However, this refinement comes at the cost of missing one core publication.

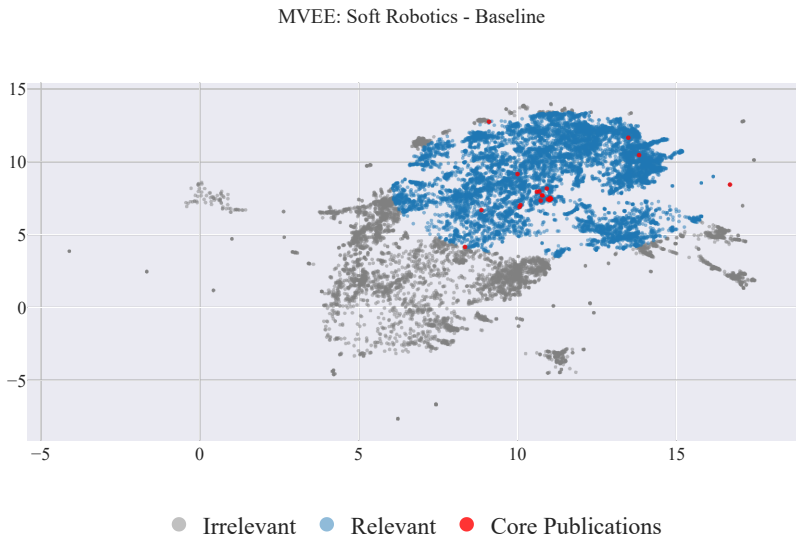

Figure 3.8.: This figure shows the relevant publications identified by the Minimum Volume Enclosing Ellipsoid (MVEE). An advantage of this approach is that we always expect that all teh core publications to be included in the identified publications.

Semantic MVEE Precision

In contrast to Semantic Cosine Precision, we opt to use the 2D embeddings generated by UMAP E_{umap}, rather than the original high-dimensional embeddings, E_o. This decision was made because earlier evaluations of the dataset showed that the MVEE consistently classified at least 50% of the total retrieved documents as relevant, which we believe is related to the high-dimensional nature of the embedding vectors.

We experiment with two enclosing shapes: the MVEE and a Convex Hull. The primary difference is that the MVEE tends to be larger due to its ellipsoidal shape, whereas the convex hull strictly bounds the points. Using the MVEE approach, 9,595 publications were identified as relevant out of the total 17,573 publications, as shown in Figure 3.8. In contrast, the convex hull, being smaller as expected, identified 7,609 publications as relevant, as illustrated in Figure A.5.

For further evaluation, understanding the quality of the UMAP embeddings (E_{UMAP}) is crucial, as they do not retain the same level of semantic meaning as the original higher-dimensional embeddings (E_o). Unlike PCA, which is a linear transformation, calculating the exact semantic loss for UMAP embeddings is challenging due to its nonlinear nature. To approximate this loss, we utilized a Partial Least Squares Regression approach as outlined by Oskolkov[1]. Based

[1] https://towardsdatascience.com/umap-variance-explained-b0eacb5b0801

on this method, we estimated that the explained variance of the two-dimensional UMAP embeddings is only 7.15%. This low explained variance underscores the significant reduction in information captured when transitioning from high-dimensional to two-dimensional space.

Semantic Clustering Precision

To cluster the embeddings, we use K-means for $2 \leq K \leq 100$, iteratively determining the smallest possible cluster containing at least 70% of the core publications which is the threshold θ described in Equation 3.2. For this process, we use the high-dimensional embeddings, E_o, as input. Cosine similarity is used as the distance measure between points which requires input normalization, which is already facilitated by the output of OpenAI's text-embedding-3-small model.

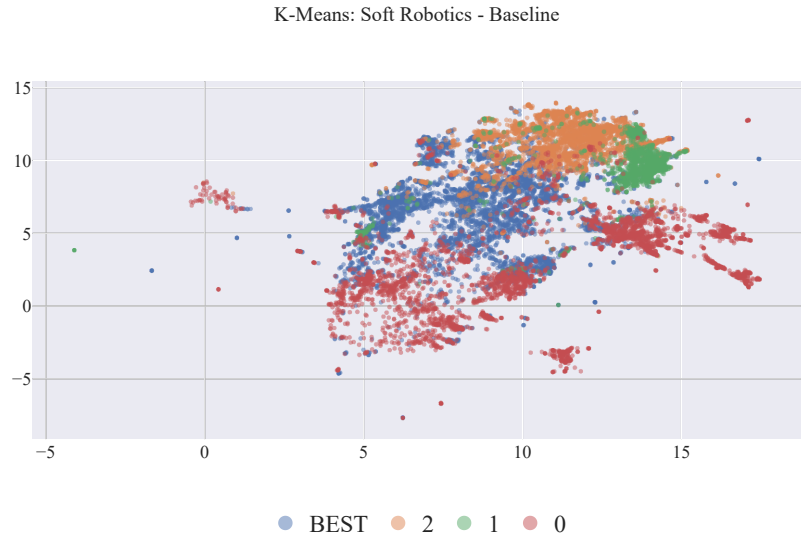

Figure 3.9.: This illustration shows the grouping of the embeddings E_o in higher-dimensional space using K-means with $K = 0, 1, 2, 3, 4$, and 5, where $K = 5$ is identified as the optimal solution. The spread-out nature of the clusters is uncommon in K-means but occurs here because clustering is performed in the higher-dimensional space before reducing the data to 2D using UMAP for visualization.

The clustering results Figure 3.9 indicate that the space can be divided into 5 groups. The best group has a size of 4,954 out of 17,573 publications and contains 15 out of the 20 core publications. We experimented with adjusting the threshold to match the quality of cosine similarity by increasing it to the next best solutions. At a threshold of approximately 0.75, the results included 6,861 publications with 17 core publications. At a threshold of approximately 0.85, all 17,573 publications were clustered together.

Additionally, we clustered the UMAP embeddings, E_{UMAP}, using the same thresholds (0.7, 0.75, and 0.85). These thresholds consistently yielded similar results, with 11,644 out of 17,573 publications identified as relevant and 19 out of the 20 core publications included. This consistency can serve as an indicator of the information loss incurred when using UMAP.

As mentioned in Section 3.2, the F_β score is the metric we will use for evaluation, with $\beta = 2$, emphasizing recall by making it twice as important as precision. However, we extend the traditional F_β score by incorporating components such as semantic precision in place of typical precision and introducing a decay factor to account for the number of semantically relevant retrieved publications. To better understand the influence of each component, we visualize their effects in Figure 3.10.

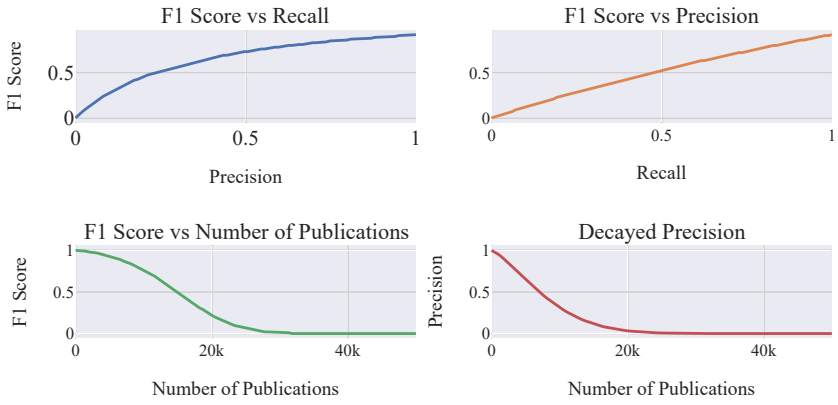

Figure 3.10.: This illustration demonstrates the effect of each component in the F_β score evaluation metric, where $\beta = 2$ and the decay hyperparameters are set to $p = 1.5$, $q = 10$, and $\alpha = 50000$. In the top left, we observe the impact of $\beta = 2$, which ensures better scaling even with lower precision if the recall is 1. The top right plot shows how the score scales almost linearly with the precision. The bottom left and right plots depict the dampening effect on the F_β score as the number of relevant publications increases, emphasizing the importance of controlling the decay to prevent inflated scores from overly large results.

4. Evaluation

In this section, we evaluate the performance of literature search queries based on the introduced metrics. This evaluation serves as a foundation for developing tools that can potentially generate automatic literature search queries in the future. It is crucial to note that the objective of this evaluation is not to assess the Search Query Writer (SQW) tool itself, but rather to evaluate any arbitrarily generated literature search query. Thus, the focus is solely on the quality of the query, independent of the method by which it was generated.

4.1. Experimental Setup

The curated dataset is constructed using two distinct methods to identify core publications: Bibliometric Analysis (14 topics) and Systematic Literature Review (7 topics), as illustrated in Figure 3.1. For the SLRs, the original queries used by the researchers are available. Consequently, we conduct two main experiments. In both experiments we use of Dimensions.ai to retrieve all required data. The retrieval process relies on their default relevance-based sorting method, which ranks publications based on the number of keyword matches between the title-abstract and the provided query.

The first experiment involves all 21 topics from both the SLRs and BAs, where we compare a baseline query against a query generated by the SQW. The baseline query consists of the exact topic name, passed into the search engine in a non-exact search fashion. For instance, the query *Soft Robotics* retrieves publications containing both words in their title or abstract, even if they do not appear consecutively.

The predicted query, however, is semi-automatically generated using the SQW tool. This process begins by providing the baseline query as input, which generates a list of keywords. These keywords are then manually sorted by the author into specific or general categories, as described in Figure A.1. The overarching topic is derived from the topic itself; for example, in the case of *Soft Robotics*, the overarching keyword *Robot* is used. In some cases, the resulting queries produced excessively large results (>100k publications). To address this, keywords were filtered to limit the results to a maximum of 50k publications, balancing evaluation cost and processing speed. Importantly, the baseline query is always included in the predicted query. This ensures that recall is at least as high for the predicted query as for the baseline, making the primary goal of the evaluation to determine whether the expanded query retrieves more core publications than the baseline without becoming overly general by retrieving irrelevant publications.

The second experiment focuses exclusively on the 7 SLR topics. It uses the exact queries and results from the first experiment but compares them to the SLR queries manually crafted by experts in the field. These expert queries are designed with well-defined research questions aimed at retrieving the most relevant publications that help tackle these exact questions.

4.2. Results

Using the data from the first experiment, we computed all the metrics, namely: Cosine Precision, Clustering Precision, MVEE Precision, Hull Precision, Recall, and the F2 score for each precision metric, as shown in Figure A.6. When examining the precision metrics, the clustering precision distinctly stands out due to its high value in certain cases, which can be directly attributed to low recall. This recall issue is also evident in some instances for the MVEE and Hull metrics, such as the baseline for *Drones in Agriculture*, where they are set to 0 because fewer than three retrieved core publications are available, which is the minimum number required to define a plane. Conversely, cosine similarity only requires a single point to function.

A strong correlation is observed between cosine precision and the MVEE and Hull methods, despite relying solely on UMAP embeddings to define the enclosing shapes. This highlights the robustness of these approaches in identifying semantically relevant publications. Additionally, we have two special case topics that had 0 recall, namely *Cloud Migration* and *Multicore Performance Prediction*. As expected, these resulted in a 0 across the board except for the cosine similarity, since it does not require any of the retrieved core publications to exist in order to compute. Instead, it only relies on the pre-computed average embeddings of the core publications. Interestingly, the results for *Cloud Migration* were not considered relevant at all, which we further experimented with and found that the first relevant publication is identified at a threshold of 0.66.

Considering the F2 score, a notable example of the impact of overly large queries without any recall improvement is *Robotic Arthroplasty*. Both the baseline and predicted queries achieved a recall of 0.957, but the expanded predicted query from the SQW retrieved significantly more results overall. Specifically, the predicted query retrieved 22,892 publications, of which only 2,834 were relevant based on cosine similarity. In contrast, the baseline query retrieved 2,151 documents, with 1,904 classified as relevant. This demonstrates how an excessively large query can dilute the precision without improving recall or the number of relevant documents retrieved.

In Table 4.1, we can better interpret the results of the first experiment by examining the differences between the scores of the predicted query and the baseline. Here, positive values indicate that the predicted query performs better, while negative values show that the baseline outperforms the predicted query.

As expected, the predicted query consistently achieves similar or better recall

Table 4.1.: In this table we can see the difference in values between the predicted query from the SQW and the baseline, whereby a negative value means that the baseline is better. As anticipated we at least always achieve a similar recall, but in most cases, the SQW yields better recall. However, it severely suffers in precision. When looking at the F2 value, we can see that the tool only notably outperforms the baseline on the three topics *Drones in Agriculture*, *Sustainable Bio Fuel Economy*, and *Multicore Performance Prediction*, whereas it shows a clear disadvantage on the topics *Perovskite Solar Cells Stability*, *Robotic Arthroplasty*, and *Cervical Myelopathy*.

		Precision				F2			
Topic	**Recall**	**Cosine**	**Clustering**	**MVEE**	**Hull**	**Cosine**	**Clustering**	**MVEE**	**Hull**
Robotic Arthroplasty	0.000	-0.761	-0.528	-0.707	-0.679	-0.560	-0.490	-0.570	-0.570
Soft Robotics	**0.111**	-0.134	-0.147	-0.292	-0.152	-0.130	-0.090	-0.400	-0.300
Crop Yield Prediction	**0.109**	-0.280	-0.118	-0.261	-0.234	-0.210	-0.230	-0.740	-0.530
Synthetic Biology	**0.310**	-0.050	-0.185	**0.637**	**0.510**	**0.030**	**0.080**	-0.420	-0.330
Resilience in Business and management	**0.185**	-0.022	-0.838	**0.150**	**0.071**	**0.060**	**0.170**	**0.330**	**0.240**
Cervical Myelopathy	**0.085**	-0.298	-0.299	-0.061	-0.017	-0.330	-0.250	-0.700	-0.590
Drones in Agriculture	**0.480**	-0.184	**0.298**	**0.069**	**0.028**	**0.220**	**0.160**	**0.240**	**0.120**
Tourism Growth Nexus	0.000	-0.562	**0.310**	0.000	0.000	-0.070	-0.040	0.000	0.000
Sustainable Biofuel Economy	**0.260**	-0.122	**0.733**	**0.513**	**0.343**	**0.190**	0.000	**0.150**	**0.320**
Perovskite Solar Cells Stability	**0.103**	-0.237	-0.213	**0.051**	**0.082**	-0.460	-0.430	-0.470	-0.540
Nanopharmaceuticals OR Nanonutraceuticals	**0.040**	**0.003**	-0.376	0.000	0.000	**0.040**	**0.010**	0.000	0.000
Green Warehousing	**0.052**	-0.227	-0.411	**0.023**	**0.004**	-0.130	-0.030	-0.530	-0.200
AI on Edge Devices	**0.250**	-0.227	**0.050**	**0.182**	**0.138**	**0.090**	-0.090	-0.050	**0.200**
Internet of Things in Healthcare	**0.172**	-0.203	-0.090	-0.146	-0.109	-0.050	-0.160	-0.500	-0.400
Software Process Line	**0.024**	-0.001	**0.239**	-0.048	-0.037	0.000	-0.020	-0.660	-0.300
Data Stream Processing Latency	**0.087**	-0.292	-0.480	-0.237	-0.196	-0.130	-0.150	-0.510	-0.480
Business Process Meta Models	**0.307**	-0.199	-0.233	-0.228	-0.148	-0.090	-0.050	-0.360	-0.300
Multicore Performance Prediction	**0.273**	-0.239	0.000	**0.158**	**0.100**	**0.200**	**0.010**	**0.400**	**0.320**
Cloud Migration	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Software Fault Prediction Metrics	**0.312**	-0.604	**0.671**	**0.006**	**0.012**	-0.050	**0.020**	-0.090	-0.010
Software Defect Prediction	**0.186**	-0.466	-0.311	-0.121	-0.233	-0.040	**0.030**	-0.390	-0.300

across all topics due to the inherent nature of the SQW. However, when evaluating precision, it is evident that the broader queries generated through query expansion often degrade the performance of the model. This effect is particularly visible in the F2 scores, where the increased number of irrelevant publications impacts the balance between recall and precision.

While the SQW demonstrates advantages in terms of recall, its over-expansion often leads to excessive noise in the results. This trade-off is especially clear for topics with a significant drop in precision or F2 scores due to the broader query scope.

The results of the second experiment, which compare the actual search queries used to identify the core publications in the SLRs, have yielded surprising yet explainable outcomes. It is important to re-emphasize that the queries initially used for the SLRs were adapted to fit dimension's query criteria and were only

Table 4.2.: This table displays the differences in values between the predicted query from the SLR and the baseline, where a negative value indicates that the baseline performs better.

		Precision				F2			
Topic	**Recall**	**Cosine**	**Clustering**	**MVEE**	**Hull**	**Cosine**	**Clustering**	**MVEE**	**Hull**
Software Process Line	-0.232	**0.286**	**0.319**	**0.162**	**0.194**	**0.250**	**0.140**	**0.290**	**0.360**
Data Stream Processing Latency	-0.073	-0.008	-0.321	-0.157	-0.125	-0.070	-0.100	-0.160	-0.160
Business Process Meta Models	**0.269**	**0.039**	-0.082	**0.421**	**0.281**	**0.190**	**0.090**	**0.390**	**0.390**
Multicore Performance Prediction	0.000	**0.217**	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Cloud Migration	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Software Fault Prediction Metrics	**0.562**	-0.596	**0.061**	**0.005**	0.000	0.000	**0.330**	-0.040	-0.020
Software Defect Prediction	-0.129	-0.584	-0.446	-0.619	-0.491	-0.280	-0.210	-0.750	-0.750

applied to search the title and abstract. In contrast, the original queries were utilized across a variety of search indices, such as title, abstract, full text, and sometimes full data for specific fields, which is a form query fine-tuning that is search engine specific. Therefore, it is important to note that the results will always be search engine dependent, which, in this case, is dimensions.ai.

As shown in Figure A.7, the results of the SLR queries are not as anticipated, particularly since they were expected to yield high recall. This discrepancy arises from the queries' reconstruction and adaptation to fit dimensions' criteria. The performance difference between the SLR queries and the baseline can be observed in Table 4.2. Overall, the baseline and the SLR queries performed on nearly equal footing, with two topics favoring the SLR and three favoring the baseline.

Interestingly, in case where the recall of the baseline significantly outperformed the SLR, namely *Software Fault Prediction Metrics*, the cosine precision was drastically lower. This results in an unwanted behavior which can be interpreted by the cosine-F2 score being 0, indicating that the recall gain was of no value due to the excessive number of irrelevant retrieved publications.

4.3. Discussion

5. Conclusion

5.1. Summary and Contributions

5.2. Outlook

A. Appendix

A.1. Further Details on Something

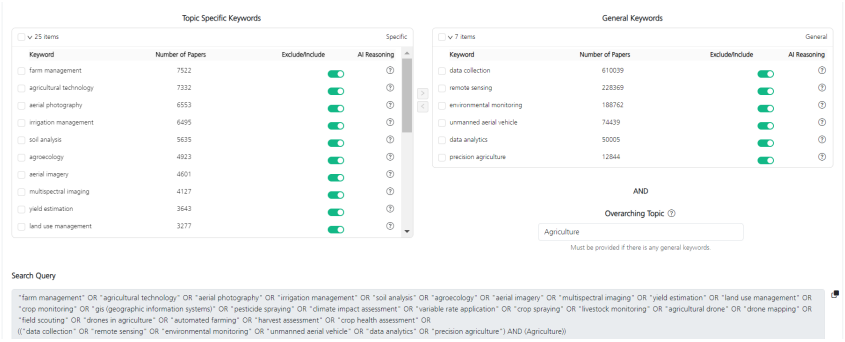

Figure A.1.: A screenshot of the SQW UI after completing the Knowledge Enrichment stage. On the left, a list of keywords is displayed alongside the number of publications associated with each keyword when used as a search term. The keywords on the right-hand side were manually categorized as general and can be roughly assessed by the number of associated publications. To narrow the scope of general keywords, we selected "agriculture" as the overarching topic. The final generated query is displayed and updated interactively as values in the transfer lists are adjusted.

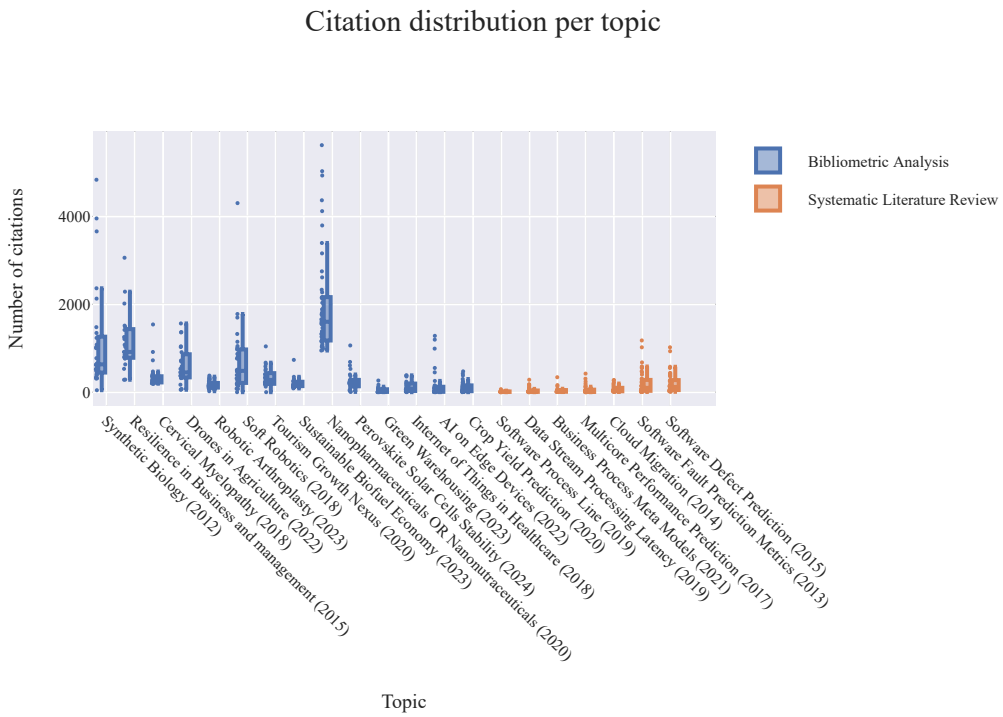

Figure A.2.: The citation ratio per topic, showing the relative citation counts of core publications compared to the average citation frequency within their respective research fields. This illustrates how the prominence of each publication compares to typical citation levels in its field.

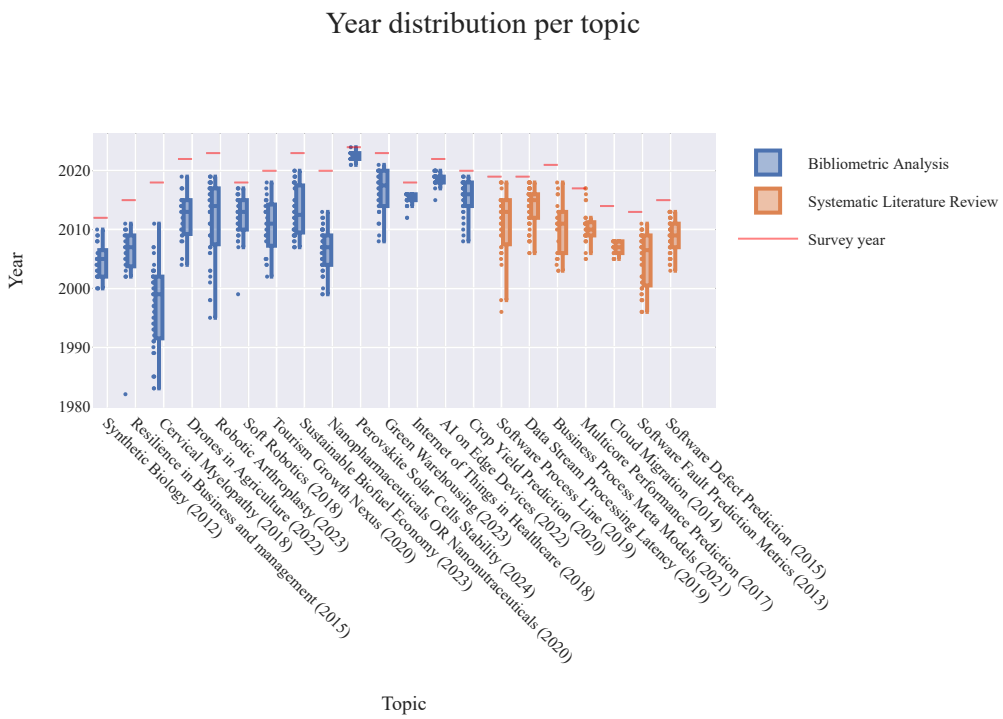

Figure A.3.: The distribution of publication years for core publications across various research topics, highlighting the historical range of studies considered in the bibliometric analyses for each field. Notably, for *Cervical Myelopathy*, the lower bound of publication years was set to 1980 for improved readability, although the actual range goes back to 1953.

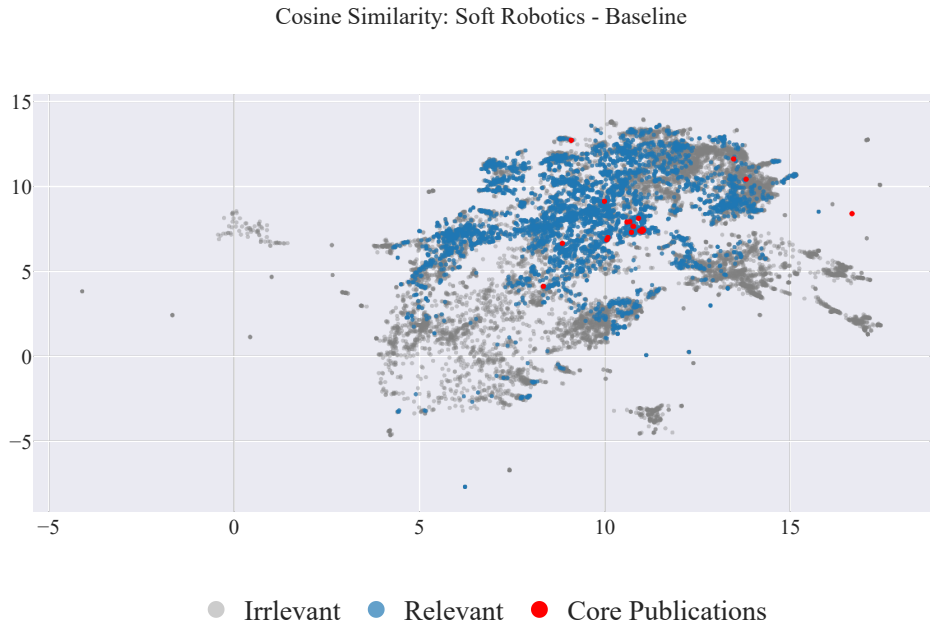

Figure A.4.: This figure illustrates the publications identified as relevant using the cosine similarity measure with the threshold θ defined by Equation 3.1, which in this case was approximately 0.547. The query results included all core publications, as expected, but also classified 75% of the total retrieved publications as relevant.

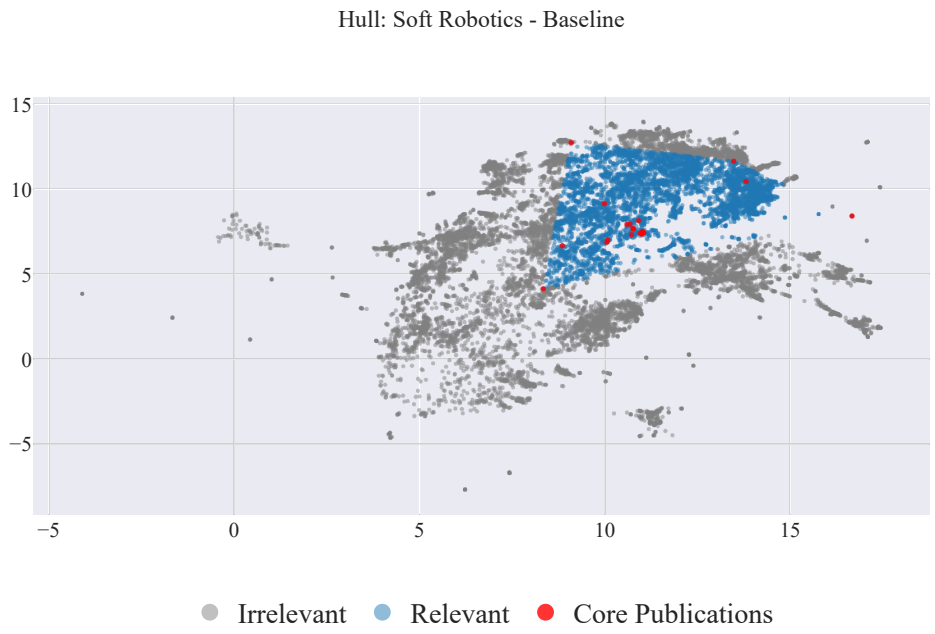

Figure A.5.: This figure shows the relevant publications identified by the Convex Hull

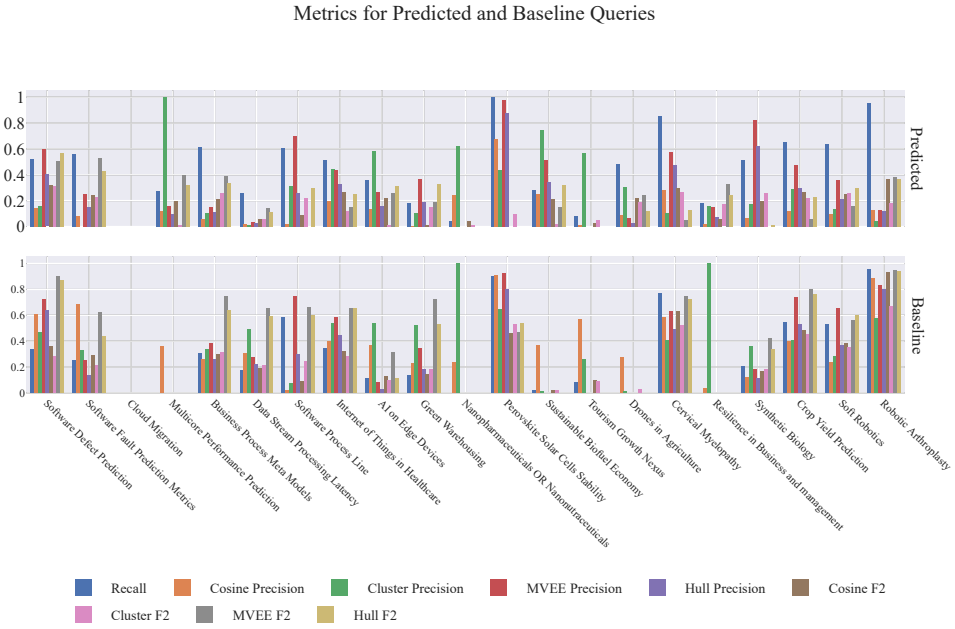

Figure A.6.: This figure shows the results of the first experiment across all the datasets. Initially, the issue with the clustering, MVEE, and Hull precision metrics becomes apparent in cases such as *Cloud Migration* and *Multicore Performance Prediction*, where the value is 0. This occurs due to a recall that is <3 for MVEE and Hull and recall of 0 for the clustering. On the other-hand the impact of the crafted F2 score is particularly evident in cases like *Robotic Arthroplasty*, where the baseline score is very high. Conversely, for the predicted query, which retrieves more publications but maintains the same recall, the score is significantly lower.

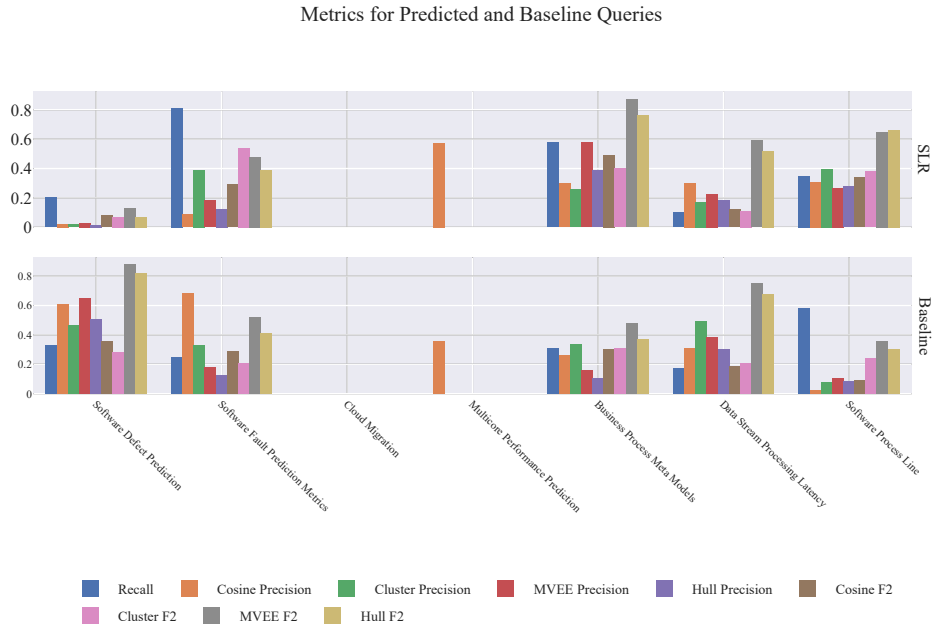

Figure A.7.: This figure shows the results of the second experiment across all the SLR datasets. Surprisingly, we can see that using the SLR query does not achieve outstanding results, which is attributed to its reconstruction and adaptation to fit dimension's search engine. Notably, issues similar to those from the first experiment due to the recall of 0 are also apparent in this case.

Bibliography

[1] M. Badami, B. Benatallah, and M. Baez. "Adaptive search query generation and refinement in systematic literature review". In: *Information Systems* 117 (2023), p. 102231 (cit. on pp. 6–8).

[2] X. Dong, C. Zhang, Y. Ge, Y. Mao, Y. Gao, J. Lin, D. Lou, et al. "C3: Zero-shot text-to-sql with chatgpt". In: *arXiv preprint arXiv:2307.07306* (2023) (cit. on p. 1).

[3] E. Kanoulas, D. Li, L. Azzopardi, and R. Spijker. "CLEF 2017 Technologically Assisted Reviews in Empirical Medicine Overview". In: Jan. 2017 (cit. on pp. 5–7).

[4] E. Kanoulas, D. Li, L. Azzopardi, and R. Spijker. "CLEF 2018 technologically assisted reviews in empirical medicine overview". In: *CEUR workshop proceedings*. Vol. 2125. 2018 (cit. on pp. 5–7).

[5] E. Kanoulas, D. Li, L. Azzopardi, and R. Spijker. "CLEF 2019 technology assisted reviews in empirical medicine overview". In: *CEUR workshop proceedings*. Vol. 2380. 2019, p. 250 (cit. on pp. 5–7).

[6] C. Lu, C. Lu, R. T. Lange, J. Foerster, J. Clune, and D. Ha. *The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery*. 2024. arXiv: 2408.06292 [cs.AI]. URL: https://arxiv.org/abs/2408.06292 (cit. on p. 1).

[7] L. McInnes, J. Healy, and J. Melville. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction". In: (2020). arXiv: 1802.03426 [stat.ML]. URL: https://arxiv.org/abs/1802.03426 (cit. on p. 14).

[8] A. Rejeb, A. Abdollahi, K. Rejeb, and H. Treiblmaier. "Drones in agriculture: A review and bibliometric analysis". In: *Computers and Electronics in Agriculture* 198 (2022). https://doi.org/10.1016/j.compag.2022.107017, p. 107017. DOI: 10.1016/j.compag.2022.107017. URL: https://app.dimensions.ai/details/publication/pub.1147958699 (cit. on p. 8).

[9] L. Van der Maaten and G. Hinton. "Visualizing data using t-SNE." In: *Journal of machine learning research* 9.11 (2008) (cit. on p. 9).

[10] S. Wang, H. Scells, J. Clark, B. Koopman, and G. Zuccon. "From Little Things Big Things Grow: A Collection with Seed Studies for Medical Systematic Review Literature Search". In: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '22. ACM, July 2022, pp. 3176–3186. DOI: 10.1145/3477495.3531748. URL: http://dx.doi.org/10.1145/3477495.3531748 (cit. on pp. 6, 7).

[11] S. Wang, H. Scells, B. Koopman, and G. Zuccon. "Can ChatGPT write a good boolean query for systematic review literature search?" In: *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2023, pp. 1426–1436 (cit. on p. 6).

Declaration

I declare that I have written this work by myself. I have identified as such all passages taken verbatim or in meaning from published or unpublished works by third parties. All sources and aids that I have used for the work are indicated.

(Example formulations follow, which you must adapt to your work for the sake of transparency. Of course, you should have discussed about the acceptability of such aids with your supervisor in advance.) In particular, the following AI systems were also used to create this work:

- ChatGPT in version ... was used for the initial text drafting based on bullet points given by me in the chapters ... / of the entire work.
- ChatGPT was consulted on the following topics: ... / was used to generate ideas regarding ... / for the structuring of ... / for the conception of the system ... .

  The wording of the dialogs and the version used were documented in the appendix of this work. Passages used are marked as such in the text.
- ChatGPT was used to create source code for ... . The wording of the dialogs and the version used were documented in the appendix of this work. The use is indicated in the header of the respective source file / class / method / parts.
- Copilot in version ... was used to create source code / auto-complete for ... . The use is documented in the header of the respective source file / class / method / parts.

I am aware that content generated by AI systems is no substitute for careful scientific work, which is why all such generated content has been critically reviewed and finalized by me.

This work has neither been submitted with the same content nor in essential parts to any other examination authority.

Your City, 2024-07-26

Mohammad Sakinini