

# VITALITY 2: Reviewing Academic Literature Using Large Language Models

Hongye An\*  
University of Nottingham

Arpit Narechania†  
Georgia Institute of Technology

Emily Wall‡  
Emory University

Kai Xu§  
University of Nottingham

## ABSTRACT

Academic literature reviews have traditionally relied on techniques such as keyword searches and accumulation of relevant back-references, using databases like Google Scholar or IEEEExplore. However, both the precision and accuracy of these search techniques is limited by the presence or absence of specific keywords, making literature review akin to searching for needles in a haystack. We present VITALITY 2, a solution that uses a Large Language Model or LLM-based approach to identify semantically relevant literature in a textual embedding space. We include a corpus of 66,692 papers from 1970-2023 which are searchable through text embeddings created by three language models. VITALITY 2 contributes a novel Retrieval Augmented Generation (RAG) architecture and can be interacted with through an LLM with augmented prompts, including summarization of a collection of papers. VITALITY 2 also provides a chat interface that allow users to perform complex queries without learning any new programming language. This also enables users to take advantage of the knowledge captured in the LLM from its enormous training corpus. Finally, we demonstrate the applicability of VITALITY 2 through two usage scenarios. VITALITY 2 is available as open-source software at <https://vitality-vis.github.io>.

**Index Terms:** large language model, retrieval augmented generation, text embedding, vector database, literature review, data visualization

## 1 INTRODUCTION

In recent years, the proliferation of vast digital repositories of academic papers has posed numerous challenges for researchers and scholars alike, including (i) efficiently retrieving pertinent information from this expansive pool, (ii) comprehensively visualizing the relationships within scholarly literature, and (iii) summarizing extensive bodies of literature. Traditional information retrieval methods for literature reviews often fall short in capturing the nuanced connections between academic papers, thereby hindering the seamless extraction of relevant insights; e.g., existing literature review approaches are often ad hoc rather than systematic [35] and even systematic approaches using techniques like keyword search can lead to inadequate coverage of evidence [2, 35].

VITALITY 1 began to address these challenges by introducing a transformer-based approach to exploring academic literature [26]. This system included a web-based visualization interface that used text embeddings to identify semantically similar literature based on input papers or even unpublished paper abstracts using SPECTER [8] and GloVe [31]. It also introduced a corpus of more than 59,000 papers across 38 prominent visualization venues, with summative user feedback demonstrating the utility of the tool.

\*e-mail: psxah15@exmail.nottingham.ac.uk

†e-mail: arpitnarechania@gatech.edu

‡e-mail: emily.wall@emory.edu

§e-mail: kai.xu@nottingham.ac.uk

While prior approaches to visualize academic articles have made significant strides towards finding semantically related articles [1, 16, 42, 26] and exploring citation networks [6, 14, 9, 43], these efforts nonetheless are limited in their ability to support intuitive interaction with a corpus of academic literature or assist individuals in summarizing large bodies of literature. We recognize the past two years have seen an explosive growth and innovation in the capability and range of applications of Large Language Models (LLMs) [51], including use within the visualization community for generating visualizations [39, 32] and authoring data-driven articles [37]. We observe an opportunity to build upon prior literature search visualization approaches and recent developments with LLMs to introduce literature search methods to address these gaps.

We present VITALITY 2, an open-source visualization system for conducting literature searches, that incorporates a novel Retrieval Augmented Generation (RAG) architecture [21]. VITALITY 2 enables users to search for relevant literature using (1) a paper(s) as the seed to find similar work, (2) the abstract of an existing paper or paper to-be-written, (3) traditional keyword-based searching, or (4) a natural language query to an LLM-powered chat interface. Results are queried from a vector database and shown in a rank-ordered table with a similarity score including title, abstract, authors, citation counts, and links to the original sources. Results are also visualized in a projection where papers that appear closer together spatially are more similar in the vector embedding space. Users can also interact with the LLM interface to summarize a corpus of papers and ask contextually relevant questions, e.g., “*what is the grounded theory method?*”

Over the VITALITY 1 predecessor system, VITALITY 2 introduces a number of novel features, including its RAG architecture and integration of prompt chaining (described in Section 3). The system supports natural context-aware conversations through LLMs to summarize and understand collections of papers. The system also augments the dataset of 59,000 papers from VITALITY 1 to cover publications across 38 visualization venues over the past 3 years for a total of 66,692 papers. VITALITY 2 is available as open-source software at <https://vitality-vis.github.io>.

The remainder of this paper is organized as follows: we review relevant background information in Section 2. Next, we discuss the architecture of VITALITY 2 in Section 3 followed by a description of the front-end interface and system capabilities in Section 4. We provide exemplary usage scenarios in Section 5. Finally, we discuss the implications and applications of this work in Section 6.

## 2 RELATED WORK

With the ever growing body of scientific literature, there is a constant need for support to conduct more effective and efficient literature research. This challenge has attracted increasing research attention as more publications become digitally available more widely. Early work comes from both within the visualization community (such as PaperVis [6]) and beyond (such as the work by El-Arini and Guestrin [10]). For this paper, we will focus on the results from visualization and related communities.

Broadly speaking, the visualization research for literature analysis spans a spectrum, from providing an overview of a field to helping find and understand papers related to a specific topic. For the

former, a well-known example is the work by Isenberg et al. [16] that aims to provide an overview of the entire visualization research landscape, using papers spanning two and a half decades. While there are other efforts that utilize a similar paper collection, i.e., all the papers from IEEE VIS, few solely focus on the large picture.

The other end of the spectrum targets literature for a specific topic. This is closer to the goal of VITALITY 2, so we would focus our discussion here. Common approaches employed by these visualization efforts include topic or semantic analysis, network analysis, and supporting the literature review workflow. From early on it has been recognized that keyword matching alone is not an effective way to find relevant papers [10, 42, 44], as a concept or technique can often be expressed in several different ways. Most of the proposed approaches use some form of text analytics or natural language processing methods. Topic modeling is a popular approach among the early work that can help identify relevant papers by grouping similar ones together [10, 42, 44]. This includes techniques that are designed for text analysis in general and can be easily applied to literature analysis such as Serendip [1]. As NLP research progresses, new techniques are being used for literature analysis, such as text embeddings generated by transformer-based models [4], including GloVe [30] and SPECTER [8] used in VITALITY 1 [26]. VITALITY 2 takes advantage of text embeddings created by the latest LLMs, which represent a breakthrough for many NLP tasks, achieving performance levels close to humans.

Network analysis is another common approach used by literature visualization, creating and visualizing co-author and affiliation network [42], citation network [6, 14, 50], and similarity network [6, 16, 4, 26], among others. VITALITY 2 focuses on paper similarity, using the latest LLM-based embeddings to create a more accurate network topology. Finally, there are a series of work targeting the literature review workflow, such as LitSense [36] and Relatedly [29]. While not a focus, the interface and interactions of VITALITY 2 are designed to match and support the workflow of common literature analysis tasks.

Dimensionality reduction is essential for visualizing high-dimensional data, such as the embeddings derived from LLMs. VITALITY 2 leverages state-of-the-art techniques to reduce the dimensionality of embeddings while retaining their semantic relationships. UMAP [3] has been shown to outperform traditional methods like t-SNE [40] and PCA [22] in terms of preserving both local and global structures in the data. This capability makes UMAP particularly suitable for visualizing the complex relationships within the literature embeddings generated by LLMs. Once the embeddings are reduced to a lower-dimensional space, they are visualized using interactive plots. These plots allow users to explore the relationships between different papers intuitively.

Using LLMs for literature visualization is still in its infancy. The closest example we found is SciDaSynth [41], which has not been peer reviewed. While it also uses LLMs to create publication embedding and provides a visual interface for exploration and analysis, it does not come with a collection of papers, and the focus of analysis is on paper comparison. Similar to SciDaSynth, VITALITY 2 introduces LLMs while retaining the original paper visualization and retrieval capabilities of VITALITY 1. VITALITY 2 offers users a novel natural language-based interaction for paper retrieval. The search results are integrated into the existing exploratory and analytical visualization panels of VITALITY 1, thereby combining LLMs technology with visualization.

### 3 ARCHITECTURE AND IMPLEMENTATION

Figure 1 illustrates the architecture, flow of data, and core links of VITALITY 2. All the papers in the corpus are first pre-processed using an LLM to create their embeddings (for similarity search), with the results stored in a vector database. This is shown as the orange part on the top. Similarity search is then performed within

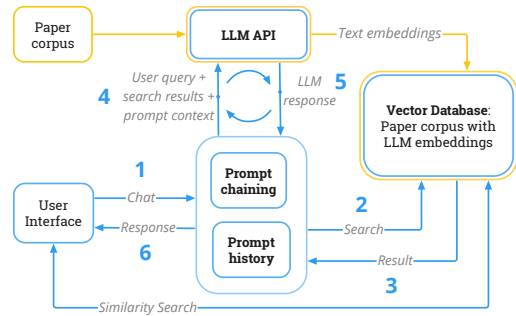


Figure 1: Architecture of VITALITY 2: **(Step 1)** User input to the system. **(Step 2 & 3)** Retrieve data from the vector database. **(Step 4)** Combine the result with user input in the prompt. **(Step 5)** Recall result from LLM. **(Step 6)** Return the final result to the user.

the vector database without further LLM access (the bottom line). User input from the chat interface (Step 1) is broken down into a series of smaller tasks using “Prompt chaining.” For each sub-task, relevant information is retrieved from the vector database (Step 2 & 3) and then combined with user input in the prompt before sending it to the LLM (Step 4). “Prompt history” provides context from previous conversations. Once all the sub-tasks are completed (Step 5), the final results are returned to the user (Step 6).

#### 3.1 Similarity Search using LLM

All the papers in the corpus are pre-processed using an LLM to create their embeddings based on paper metadata, which includes the title, authors, the conference or journal in which the paper is published, publication date, keywords, and abstract. These text embeddings convert textual data into high-dimensional vectors that capture semantic relationships among paper metadata [38], ensuring thorough comprehension of each paper and enabling more effective similarity searches within the vector database. For clarity, the “text embedding” mentioned refers to embeddings created from paper metadata rather than the full paper content. This metadata-focused approach balances detail with computational efficiency and allows for robust semantic matching.

To find similar papers from an academic literature corpus, while VITALITY 1 uses GloVe [30] and SPECTER [8] embeddings, VITALITY 2 uses ADA, the embeddings generated by OpenAI’s *text-embedding-ada-002* model [28], which is a breakthrough model that generates contextually rich embeddings by leveraging advanced attention mechanisms and transformer architectures. Thus, VITALITY 2 now includes ADA, GloVe, and SPECTER embeddings. VITALITY 2 also leverages two *vector databases* (Faiss [17] and ChromaDB [7]) to store and manage these embeddings and also to locate similar papers via approximate nearest neighbor search.

#### 3.2 Chat with Papers using RAG and Prompt Chaining

While text embeddings and vector databases allow finding similar papers, there are several other kinds of analyses a user may want to perform, such as understanding a technical concept, summarizing a single paper, or writing a literature review based on multiple papers. To perform such analyses, users may need to build customized tools and/or learn a new query language, neither of which are readily accessible to nor understandable by many users. VITALITY 2 addresses this issue by leveraging the powerful natural language understanding and generation capabilities of LLMs, allowing users to directly engage with the LLM using natural language.

However, there are a few challenges when applying LLMs for such analyses: (1) *Prompt Size* – This refers to the largest prompt that an LLM can accept and depends on the LLM type and version.

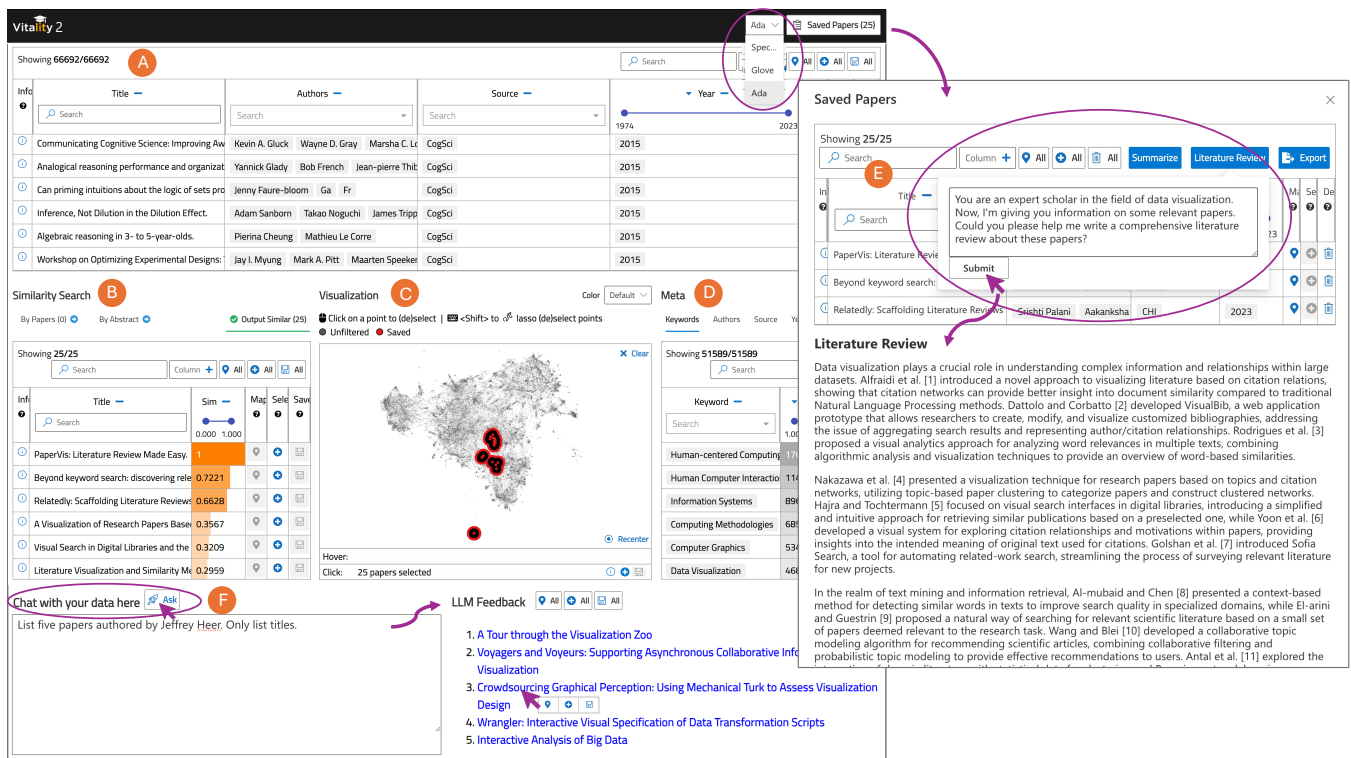


Figure 2: The VITALITY 2 User Interface. (A) **Paper Collection View** shows the entire corpus of publications, (B) **Similarity Search View** shows options to look-up publications that are similar to another list of publications or by a work-in-progress title and abstract, (C) **Visualization Canvas** shows an interactive 2-D UMAP projection of the embedding space of the entire paper collection, (D) **Meta View** shows summaries of certain attributes with respect to the Paper Collection View (A), (E) Opens a **Saved Papers View** from which the saved papers can be exported as JSON. Extending VITALITY 1, we added (F) **Chat with your Data** view to allow users to ask natural-language based questions based on the paper corpus. We also added ADA embeddings (in addition to GloVe and SPECTER embeddings) and enable users to *Summarize* or write a *Literature Review* on the **Saved Papers** using LLMs, including the ability to customize the prompts.

Earlier versions of LLMs accepted a few thousand tokens, wherein each word in a prompt accounted for a few tokens. More recent versions have a larger limit, e.g., up to 16K tokens for GPT 3.5 [11]. However, this can still limit analyses involving a large corpus of papers, which often have thousands of words each. (2) *Hallucination* – it is well known that LLMs can create seemingly plausible information about something that does not exist [47], e.g., suggest papers that never existed, that can be detrimental. (3) *Limited Comprehension Capabilities* – Despite the ability to generate seemingly logical and coherent text, LLMs lack genuine comprehension capabilities. This implies that an LLM may encounter difficulties when handling complex problems or tasks, particularly those requiring a deeper understanding of context or concepts.

To overcome these issues, we employ two popular approaches: Retrieval Augmented Generation (RAG) and Prompt Chaining.

**RAG.** RAG semantically processes the user’s input query to only retrieve relevant information from within the data corpus, thereby reducing the subsequent prompt size and also minimizing the risk of hallucination. This retrieval process is not about finding exact answers but rather fetching documents that contain potentially useful context or information related to the query.

**Prompt Chaining.** Prompt chaining breaks down complex tasks to a series of smaller steps and provides specific prompts that are known to be effective for each of these steps [45]. VITALITY 2 borrows this idea and implements a conversation framework to manage the content and context of user queries and LLM responses. VITALITY 2 uses LangChain [20] a popular open-source library for this purpose. As LLM API calls operate independently of each

other without inherent memorization or management of the context, there is no built-in functionality for retaining conversation history. Sending the entire dialogue history with each LLM API call risks exceeding the maximum token limit. Therefore, we adopt a two-step approach in VITALITY 2: first, a summary of recent conversations is generated by calling the LLM API once to obtain a “condensed conversation history.” Then, in the second API call, the user’s query, conversation history, and retrieved information are concatenated into a prompt and sent to the LLM API. This an example of prompt chaining, that efficiently combines the results of two or more LLM API calls.

## 4 VITALITY 2

We present VITALITY 2, an LLM-powered visual analytics tool to help users write academic literature reviews.

### 4.1 Dataset of Academic Articles

VITALITY 1 [26] provided a dataset of 59,232 academic papers from visualization and HCI literature, along with their metadata such as their title, abstract, author(s), publisher, publication year, citation counts, and n-dimensional and 2-dimensional vector embeddings (GloVe and SPECTER). VITALITY 1 also open-sourced a web-scraping framework to extract the above information from digital repositories such as IEEE Xplore and ACM Digital Library for continued development of the corpus. VITALITY 2 used this scraper to extract more recent papers between 2021-2023, resulting in an augmented dataset of 66,692 papers.

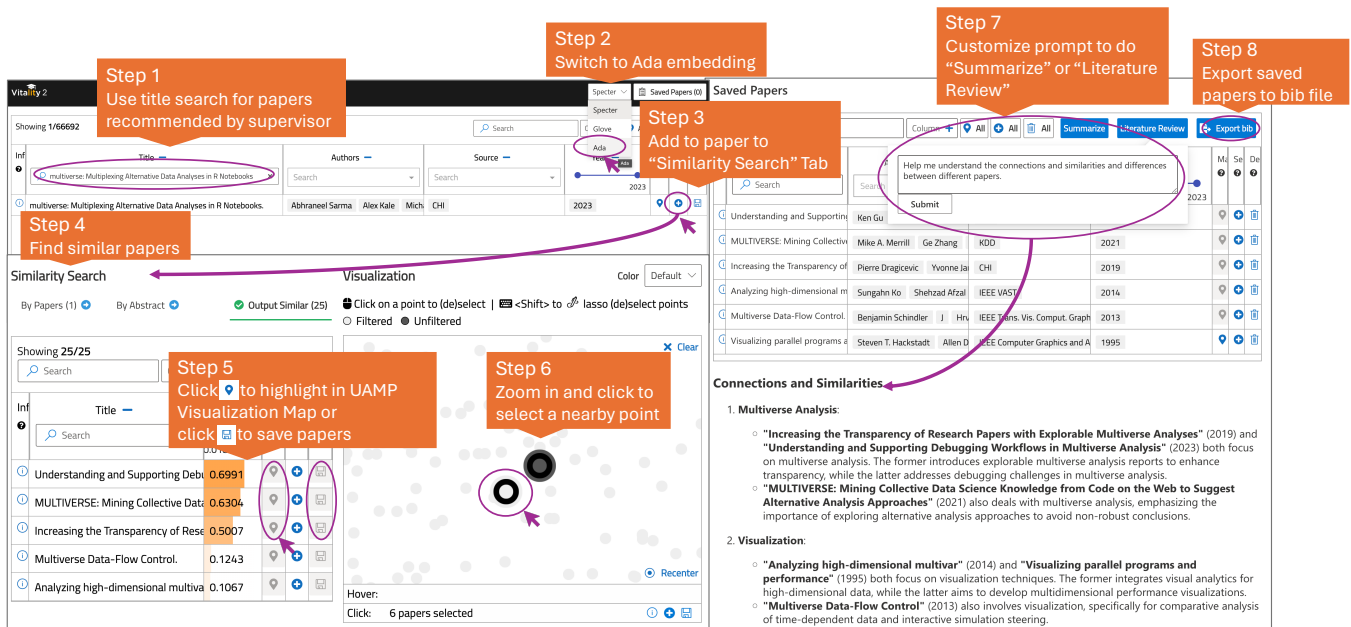


Figure 3: Noori’s process of doing literature review using VITALITY 2. (Step 1) Noori searches the VITALITY 2 database for articles recommended by supervisor. (Step 2) Noori selects Ada Embedding as the embedding option used by VITALITY 2 in “Similarity search”. (Step 3) Noori adds the paper she just searched as a seed for “Similarity Search”. (Step 4) Noori uses “Similarity Search” to find some related papers. (Step 5) Noori saves papers with similarity score of  $> 0.1$  and highlights them in the UMAP Visualization MAP. (Step 6) Noori selects an additional paper that interested her in the UMAP visualization map. (Step 7) Noori tries to modify and use different prompts and does “Summarize” and “Literature Review”. (Step 8) Noori exports the saved papers to a bib file.

## 4.2 User Interface

Figure 2 shows the VITALITY 2 user interface, illustrating the new features built on top of VITALITY 1.

In VITALITY 1, the **Paper Collection View** (A) shows the entire corpus of publications, the **Similarity Search View** (B) shows options to look-up publications that are similar to another list of publications or by a work-in-progress title and abstract, the **Visualization Canvas** (C) shows an interactive 2-D UMAP projection of the embedding space of the entire paper collection, the **Meta View** (D) shows summaries of certain attributes with respect to the Paper Collection View (A), and (E) opens a **Saved Papers View** from which the saved papers can be exported in a JSON and .bibtex format for later use. We added three new capabilities in VITALITY 2.

First, we expanded available embedding options (from GloVe and SPECTER) to add OpenAI’s ADA [27]. Users can view the 2-dimensional ADA embeddings in the UMAP and search for similar papers (by title or abstract). Based on our own testing and consistent with prior benchmarks [25], we found ADA embeddings to perform better than GloVe and SPECTER on VITALITY 2 features.

Second, we added a new **Chat with your Data View** (F) to allow users to ask natural-language questions based on the entire corpus, e.g., “*Help me find some papers related to geographic science visualization.*” For papers mentioned in the LLM’s output, VITALITY 2 allows the user to map (view in the UMAP), select (look up other similar papers), or save them (include it in a literature review or export). This capability holds immense potential for revolutionizing scholarly information access, enabling users to effectively harness the capabilities of RAG and pose a wide spectrum of inquiries.

Third, we also added novel capabilities to summarize papers and conduct literature reviews (in the **Saved Papers** view), building upon the robust semantic understanding capabilities of LLM. Clicking the “Summarize” button outputs a short summary of each of the saved papers, one below the other. Clicking the “Literature Review” button goes a step further, and outputs a comprehensive

literature review based on the saved papers, including descriptions, comparisons, and a bibliography. These enhancements enable users to swiftly grasp the key information contained within their saved papers or draft an early version of their related work section in their ongoing manuscript. Note that VITALITY 2 allows users to customize the base LLM prompts to control the verbosity, writing style, and format of the LLM’s response.

## 5 USAGE SCENARIOS

### 5.1 LLM Summarization of Literature Review

Noori is an undergraduate student who has joined a visualization research lab, working under the supervision of a faculty member and Ph.D. student on an ongoing project conducting controlled experiments on multiverse analyses. Per her supervisor’s suggestion, she uses VITALITY 2 to kickstart her literature review.

She begins using the “Saved Papers” feature in VITALITY 2 to collect papers through various search methods. She starts with a recent paper her supervisor suggested from the CHI conference by Sarma and colleagues [33] and both “Selects” it and “Saves” it. From this selection, she searches for related papers using the ADA embedding. Of the 25 output similar papers, she decides to save any that have a similarity score of  $> 0.1$ , which results in an additional 5 papers (e.g., [12, 19]). She highlights the papers in the UMAP visualization and hovers on nearby papers, selecting an additional relevant paper on parallel program performance [13].

Noori visits the set of saved papers in “Saved Papers” and selects “Literature Review.” VITALITY 2 shares (1) a paper-by-paper summarization, then (2) a comparison and contrast of the set of 7 total papers. Noori adjusts the default prompt to describe a higher-level summary to suit her experience, then reads through the result. The literature review points out the motivation of most of the papers on transparency of data analysis, and reveals a divide where some papers represented earlier relevant ideas (e.g., [19, 34]), while the others were more recent (e.g., [33, 24]). She accordingly uses these

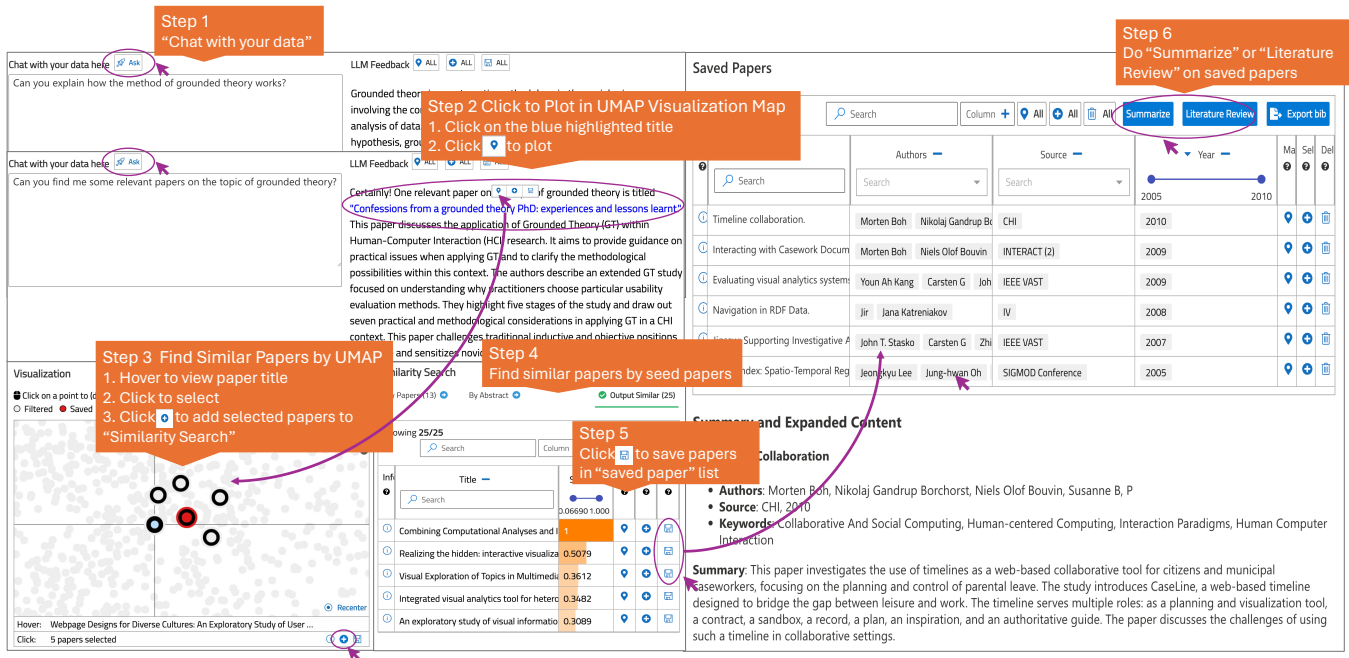


Figure 4: Aaron's process for literature search using VITALITY 2. (Step 1) Aaron utilizes the “Chat with your data” feature to quickly explore the domain of “grounded theory”, an area unfamiliar to him. (Step 2) Aaron plots an paper cited in the LLM feedback onto the UMAP visualization interface. (Step 3) Aaron selects a set of closely related papers from the UMAP visualization and Aaron adds these papers to the “Similarity Search”. (Step 4) Aaron uses “Similarity Search” feature to find some semantically similar papers. (Step 5) Aaron saves a subset of papers of particular interest to his “saved papers” list. (Step 6) Aaron employs the “Summarize” and “Literature Review” feature to review the saved papers.

observations to help her write a first draft of the related work section on multiverse visualization to her team’s Overleaf document. She uses the “Export” feature of VITALITY 2 to add the related papers to the .bib file, and she adds a short paragraph detailing her literature review methodology – including use of VITALITY 2 and LLM summarization in the initial draft.

She reads the papers in more detail, iterates on the writing accordingly, then shares it with her supervisor for feedback. She begins her next literature review task: identifying and summarizing literature on general data analysis workflows with VITALITY 2.

## 5.2 Contextual Conversations about Papers with LLM

Aaron is a new faculty member at a university, having just defended his Ph.D. Having primarily used quantitative methods for his dissertation research, he wants to explore qualitative methods in his next project: conducting in-depth interviews to understand the potential of visualization to address user concerns around misinformation.

Aaron begins using VITALITY 2 to search for a CSCW paper he knows on misinformation in times of crisis [15]. He selects the “info” icon to read the abstract and additional paper metadata. The abstract mentions the use of “*constructivist grounded theory to guide [their] inquiry.*” Aaron has heard of this method before, but is not familiar with how it works. Using the LLM interface in VITALITY 2, he asks “*can you explain how the method of grounded theory works?*” VITALITY 2 responds with an in-depth description (Figure 4). He then asks “*can you find me some relevant papers on the topic of grounded theory?*” The system responds with a description of some relevant papers, which use the grounded theory method. The result is shown in Figure 4.

Aaron continues his literature review accordingly, having a better baseline understanding of the method. Aaron notices that VITALITY 2 highlights the titles of the papers cited in the answer in bold blue text. Aaron clicks on these titles, and VITALITY 2 pops up a function box, including (1) Highlight in the UMAP visualization,

(2) Add the paper to the similarity search, and (3) Save the paper to the “Saved Papers” list. Aaron selects one of the papers he is interested in and uses the similarity search function to find other papers similar to it. Next, Aaron saves all of these articles in the “Saved Paper” list. Finally, Aaron uses the *Summarize* and *Literature Review* features to further review on these saved papers.

## 6 DISCUSSION, LIMITATIONS, & FUTURE WORK

We observe a few limitations to our current approach. First, while using LLMs to summarize a set of papers can be a useful starting point for a literature review, the quality of the output is far from sufficient for being included in a paper as-is. The summarization is based on the meta-data that is contained in the VITALITY 2 paper corpus, which does not include full-text of the articles. A potential solution is to optimize the web crawler to also retrieve the full text of papers, segment these texts into appropriately sized chunks [49] and generate embeddings for these chunks. These embeddings, along with the associated meta-data (title, authors, keywords, abstract, etc.), should then be stored in the vector database. This will ensure more comprehensive and accurate summaries.

Furthermore, some studies also indicate that there is concern within the academic community regarding the development and use of LLMs [18]. The accuracy of response by LLMs depends on the initial training. Potential biases, inaccuracies, or misunderstandings present in the data used for training the models can lead to erroneous outputs. To mitigate this problem, we suggest adding a user prompt in VITALITY 2 to explicitly warn users about the limitations of content generated by VITALITY 2, thereby reminding and cautioning users to use this tool with care.

Another concern is hallucination from LLMs [46]. In one instance, when asked to describe the concept of “grounded theory” in Usage Scenario 2 (Section 5), the LLM described a reasonable high-level summary of the approach, but referenced a paper not

contained in the VITALITY 2 database. Upon searching the title, it referred to a reasonably popular book from Birks and Mills in 2015 with more than 2,000 citations at the time of this writing [5]. However, when asking for additional relevant papers on the topic of grounded theory (implying that they come from the VITALITY 2 corpus), the LLM responded with some paper titles that were neither contained in the VITALITY 2 database, nor obvious paper titles outside the database according to a Google Scholar search. Although completely eliminating hallucinations in LLMs remains challenging [48], various strategies exist to reduce their occurrence. As discussed in this paper, actually the RAG architecture is one such approach. RAG mitigates LLMs hallucinations by providing additional contextual information. In future work, we can further reduce hallucinations in RAG by incorporating external knowledge bases from reliable sources, such as Google Scholar search. Additionally, as research on LLMs advances, several potential methods to address the issue of hallucinations in LLMs have emerged. Recent benchmarks suggest that the incidence of AI hallucination is relatively small for GPT-4 by OpenAI [23].

## 7 CONCLUSION

We introduced a system, VITALITY 2, for conducting literature review using a RAG architecture, a corpus of more than 66,000 papers from visualization-related venues, and novel features for interacting with the paper corpus through Large Language Models (LLMs). We provide the system as an open-sourced code contribution at <https://vitality-vis.github.io>, alongside the paper corpus, and hope to stimulate future work in optimizing literature review methods.

## REFERENCES

- [1] E. Alexander, J. Kohlmann, R. Valenza, M. Witmore, and M. Gleicher. Serendip: Topic model-driven visual exploration of text corpora. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 173–182. IEEE, 2014. 1, 2
- [2] R. F. Baumeister and M. R. Leary. Writing narrative literature reviews. *Review of general psychology*, 1(3):311–320, 1997. 1
- [3] E. Becht, L. McInnes, J. Healy, C.-A. Dutertre, I. W. Kwok, L. G. Ng, F. Ginhoux, and E. W. Newell. Dimensionality reduction for visualizing single-cell data using umap. *Nature biotechnology*, 37(1):38–44, 2019. 2
- [4] A. Benito-Santos and R. Therón. GlassViz: Visualizing Automatically-Extracted Entry Points for Exploring Scientific Corpora in Problem-Driven Visualization Research. In *2020 IEEE Visualization Conference (VIS)*, 2020. 2
- [5] M. Birks and J. Mills. *Grounded theory: A practical guide*. Sage, 2015. 6
- [6] J.-K. Chou and C.-K. Yang. Papervis: Literature review made easy. In *Computer Graphics Forum*, vol. 30, pp. 721–730. Wiley Online Library, 2011. 1, 2
- [7] ChromaDB. The AI-native open-source embedding database. <https://www.trychroma.com>. Accessed: 2024-04-23. 2
- [8] A. Cohan, S. Feldman, I. Beltagy, D. Downey, and D. S. Weld. Specter: Document-level representation learning using citation-informed transformers. In *ACL*, 2020. 1, 2
- [9] A. Dattolo and M. Corbato. Visualbib: narrative views for customized bibliographies. In *2018 22nd International Conference Information Visualisation (IV)*, pp. 133–138. IEEE, 2018. 1
- [10] K. El-Arini and C. Guestrin. Beyond keyword search: discovering relevant scientific literature. In *ACM SIGKDD*, 2011. 1, 2
- [11] Models - OpenAI API. <https://platform.openai.com/docs/models/gpt-3-5-turbo>. Accessed: 2024-04-23. 3
- [12] K. Gu, E. Jun, and T. Althoff. Understanding and supporting debugging workflows in multiverse analysis. In *CHI*, 2023. 4
- [13] S. T. Hackstadt and A. D. Malony. Visualizing parallel programs and performance. *IEEE CGA*, 15(4):12–14, 1995. 4
- [14] F. Heimerl, Q. Han, S. Koch, and T. Ertl. Citerivers: Visual analytics of citation patterns. *IEEE TVCG*, 22(1):190–199, 2015. 1, 2
- [15] Y. L. Huang, K. Starbird, M. Orand, S. A. Stanek, and H. T. Pederesen. Connected through crisis: Emotional proximity and the spread of misinformation online. In *ACM CSCW*, 2015. 5
- [16] P. Isenberg, T. Isenberg, M. Sedlmair, J. Chen, and T. Möller. Visualization as seen through its research paper keywords. *IEEE TVCG*, 2016. 1, 2
- [17] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus, 2019. 2
- [18] J. K. Kim, M. Chua, M. Rickard, and A. Lorenzo. Chatgpt and large language model (llm) chatbots: The current state of acceptability and a proposal for guidelines on utilization in academic medicine. *Journal of Pediatric Urology*, 19(5):598–604, 2023. doi: 10.1016/j.jpuro.2023.05.018 5
- [19] S. Ko, S. Afzal, S. Walton, Y. Yang, J. Chae, A. Malik, Y. Jang, M. Chen, and D. Ebert. Analyzing high-dimensional multivariate network links with integrated anomaly detection, highlighting and exploration. In *2014 IEEE conference on visual analytics science and technology (VAST)*, pp. 83–92. IEEE, 2014. 4
- [20] LangChain. <https://www.langchain.com/>. Accessed: 2024-04-23. 3
- [21] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474. Curran Associates, Inc., 2020. 1
- [22] A. Maćkiewicz and W. Ratajczak. Principal components analysis (pca). *Computers & Geosciences*, 19(3):303–342, 1993. 2
- [23] T. R. McIntosh, T. Liu, T. Susnjak, P. Watters, A. Ng, and M. N. Haggamuge. A Culturally Sensitive Test to Evaluate Nuanced GPT Hallucination. *IEEE Transactions on Artificial Intelligence*, pp. 1–13, 2023. Conference Name: IEEE Transactions on Artificial Intelligence. doi: 10.1109/TAI.2023.3332837 6
- [24] M. A. Merrill, G. Zhang, and T. Althoff. Multiverse: mining collective data science knowledge from code on the web to suggest alternative analysis approaches. In *ACM SIGKDD*, 2021. 4
- [25] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022. 4
- [26] A. Narechania, A. Karduni, R. Wesslen, and E. Wall. VitaLITY: Promoting Serendipitous Discovery of Academic Literature with Transformers & Visual Analytics. *IEEE TVCG*, Jan. 2022. 1, 2, 3
- [27] A. Neelakantan, T. Xu, R. Puri, A. Radford, J. M. Han, J. Tworek, Q. Yuan, N. Tezak, J. W. Kim, C. Hallacy, et al. Text and code embeddings by contrastive pre-training, 2022. 4
- [28] OpenAI. New and improved embedding model. <https://openai.com/blog/new-and-improved-embedding-model>, 2022. Accessed: 2024-04-23. 2
- [29] S. Palani, A. Naik, D. Downey, A. X. Zhang, J. Bragg, and J. C. Chang. Relatedly: Scaffolding Literature Reviews with Existing Related Work Sections. In *ACM CHI*, 2023. 2
- [30] J. Pennington, R. Socher, and C. Manning. GloVe: Global Vectors for Word Representation. In *ACM EMNLP*, 2014. 2
- [31] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *ACM EMNLP*, 2014. 1
- [32] S. Sah, R. Mitra, A. Narechania, A. Endert, J. Stasko, and W. Dou. Generating Analytic Specifications for Data Visualization from Natural Language Queries using Large Language Models. *NLVIZ Workshop (IEEE VIS)*, 2024. 1
- [33] A. Sarma, A. Kale, M. J. Moon, N. Taback, F. Chevalier, J. Hullman, and M. Kay. multiverse: Multiplexing alternative data analyses in r notebooks. In *ACM CHI*, 2023. 4
- [34] B. Schindler, J. Waser, H. Ribičić, R. Fuchs, and R. Peikert. Multiverse data-flow control. *IEEE TVCG*, 19(6):1005–1019, 2012. 4
- [35] H. Snyder. Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, 104:333–339, 2019. 1
- [36] N. Sultanum, C. Murad, and D. Wigdor. Understanding and Supporting Academic Literature Review Workflows with LitSense. In *Proceedings of the International Conference on Advanced Visual Interfaces*, 2020. 2
- [37] N. Sultanum and A. Srinivasan. Datatales: Investigating the use of

- large language models for authoring data-driven articles. In *2023 IEEE Visualization and Visual Analytics (VIS)*, 2023. 1
- [38] J. Tang, M. Qu, and Q. Mei. PTE: Predictive Text Embedding through Large-scale Heterogeneous Text Networks. In *ACM SIGKDD*, 2015. 2
- [39] Y. Tian, W. Cui, D. Deng, X. Yi, Y. Yang, H. Zhang, and Y. Wu. Chartgpt: Leveraging LLMs to generate charts from abstract natural language. *IEEE TVCG*, 2024. 1
- [40] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 2
- [41] X. Wang, S. L. Huey, R. Sheng, S. Mehta, and F. Wang. SciDaSynth: Interactive Structured Knowledge Extraction and Synthesis from Scientific Literature with Large Language Model. <https://doi.org/10.48550/arXiv.2404.13765>. 2
- [42] Y. Wang, M. Yu, G. Shan, H.-W. Shen, and Z. Lu. Vispubcompas: a comparative analytical system for visualization publication data. *Journal of Visualization*, 22(5):941–953, 2019. 1, 2
- [43] J. Wilkins, J. Järvi, A. Jain, G. Kejriwal, A. Kerne, and V. Gumudavelly. Evolutionworks. In *IFIP Conference on Human-Computer Interaction*, pp. 213–230. Springer, 2015. 1
- [44] S. Wu, Y. Zhao, F. Parvinzamid, N. Ersotelos, S. Wei, and F. Dong. Literature Explorer: effective retrieval of scientific documents through nonparametric thematic topic detection. *The Visual Computer*, 36, July 2020. doi: 10.1007/s00371-019-01721-7 2
- [45] T. Wu, M. Terry, and C. J. Cai. AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts. In *ACM CHI*, 2022. 3
- [46] Z. Xu, S. Jain, and M. Kankanhalli. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*, 2024. 5
- [47] J.-Y. Yao, K.-P. Ning, Z.-H. Liu, M.-N. Ning, and L. Yuan. LLM Lies: Hallucinations are not Bugs, but Features as Adversarial Examples, Oct. 2023. doi: 10.48550/arXiv.2310.01469 3
- [48] J.-Y. Yao, K.-P. Ning, Z.-H. Liu, M.-N. Ning, and L. Yuan. Llm lies: Hallucinations are not bugs, but features as adversarial examples. *arXiv preprint arXiv:2310.01469*, 2023. 6
- [49] A. J. Yepes, Y. You, J. Milczek, S. Laverde, and L. Li. Financial report chunking for effective retrieval augmented generation. *arXiv preprint arXiv:2402.05131*, 2024. 5
- [50] T. Yoon, H. Han, H. Ha, J. Hong, and K. Lee. A Conference Paper Exploring System Based on Citing Motivation and Topic. In *IEEE PacificVis*, 2020. 2
- [51] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023. 1