ClusterChat: Multi-Feature Search for Corpus Exploration

Ashish Chouhan
chouhan@informatik.uni-heidelberg.de
Heidelberg University
Heidelberg, Germany

Saifeldin Mandour
saifeldin.mandour@stud.uni-heidelberg.de
Heidelberg University
Heidelberg, Germany

Michael Gertz
gertz@informatik.uni-heidelberg.de
Heidelberg University
Heidelberg, Germany

Abstract

Exploring large-scale text corpora presents a significant challenge in biomedical, finance, and legal domains, where vast amounts of documents are continuously published. Traditional search methods, such as keyword-based search, often retrieve documents in isolation, limiting the user's ability to easily inspect corpus-wide trends and relationships. We present *ClusterChat*[1], an open-source system for corpus exploration that integrates cluster-based organization of documents using textual embeddings with lexical and semantic search, timeline-driven exploration, and corpus and document-level question answering (QA) as multi-feature search capabilities. We validate the system with two case studies on a four million abstract PubMed dataset, demonstrating that *ClusterChat* enhances corpus exploration by delivering context-aware insights while maintaining scalability and responsiveness on large-scale document collections.

CCS Concepts

• **Information systems** → *Clustering*; **Search interfaces**; *Question answering*.

Keywords

corpus exploration, clustering, federated learning, retrieval augmented generation

ACM Reference Format:
Ashish Chouhan, Saifeldin Mandour, and Michael Gertz. 2025. ClusterChat: Multi-Feature Search for Corpus Exploration. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3726302.3730137

1 Motivation and Background

The increase in publications in domains such as biomedicine, finance, or legal presents researchers with exceptional opportunities but also with significant challenges. For example, in biomedical research, a resource such as PubMed[2] contains over 36 million articles of biomedical literature, with more than 1 million added annually [5]. As this volume of information grows, processes of retrieving documents to satisfy specific information needs and extracting meaningful insights, has become a complex and time-consuming task. Traditional search methods, such as keyword-based search, often retrieve documents in isolation, limiting the ability of users to uncover semantic relationships among documents. For example, both PubMed [5] and legal platforms like EUR-Lex[3] return a list of articles in response to user queries without further analysis of the retrieved articles.

Corpus exploration systems are developed to address these challenges, enabling users to browse and discover insights from large text corpora. Systems such as *Carrot2*[4] and Knowledge Navigator [6] offer structured approaches to corpus exploration by organizing results into thematic clusters or topics. While these systems represent significant advances, they are often limited in scope. For example, *Carrot2* excels at cluster-based organization but is not open-source and lacks corpus and document-level question-answering (QA), semantic search, and timeline-centric exploration (temporal filtering). Knowledge Navigator organizes and structures retrieved documents into navigable hierarchical topics using clustering and lexical search but does not allow for dynamic interaction with results. Similarly, Zheng et al. [13] introduce OpenResearcher, which leverages Retrieval-Augmented Generation (RAG) [7] to provide corpus and document-level answers but does not integrate cluster organization and timeline-driven exploration. Reveal[5] provides cluster organization and search for concepts representing a cluster; however, it is not open-source and lacks a QA feature. González-Márquez et al. [2] present Nomic Atlas, an interactive web version of the 2D atlas of the PubMed database having features similar to *ClusterChat*. However, it is not open-source, lacks timeline-driven exploration, and does not support corpus-level QA for question-answering on the entire corpus. To better illustrate the capabilities and limitations of existing systems, we present a comparison of the aforementioned features in Table 1. In addition to corpus exploration systems, embedding visualization tools like Embedding Projector [10] and WizMap [12] have been developed to facilitate the interpretation of high-dimensional embeddings. These open-source tools provide clustering, temporal filtering, and lexical search capabilities but lack support for semantic search and QA.

Recognizing these limitations of existing systems and tools, we introduce *ClusterChat*, an open-source system designed to provide multi-feature search capabilities for corpus exploration. *ClusterChat* integrates the key features listed in Table 1 and enables users to interactively filter documents and explore a corpus based on topics. For example, a researcher investigating targeted therapies for non-small cell lung cancer (NSCLC) in recent articles can first explore a

[1] The demo video and source code are available at: https://github.com/achouhan93/ClusterChat

[2] https://pubmed.ncbi.nlm.nih.gov (accessed on 11th April 2025)

[3] https://eur-lex.europa.eu (accessed on 11th April 2025)

[4] https://search.carrot2.org (accessed on 11th Feb 2025)

[5] https://www.revealdata.com/ (accessed on 11th Feb 2025)

SIGIR '25, July 13–18, 2025, Padua, Italy.

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730137

Table 1: Comparison of feature support across different corpus exploration systems. The table highlights seven key features: clustering for topics, temporal filtering, keyword-based (lexical) search, semantic search, corpus-level QA, document-level QA on filtered documents, and open-source availability.

Feature	**ClusterChat (ours)**	Nomic Atlas	*Carrot2*	Reveal	Knowledge Navigator	OpenResearcher
Clustering	✓	✓	✓	✓	✓	✗
Temporal Filtering	✓	✗	✗	✗	✗	✗
Lexical Search	✓	✓	✗	✓	✓	✓
Semantic Search	✓	✓	✗	✗	✗	✓
Corpus-level QA	✓	✗	✗	✗	✓	✓
Document-level QA	✓	✓	✗	✗	✓	✓
Open Source	✓	✗	✗	✗	✓	✓

high-level view of the corpus through thematic clusters and then apply temporal filters to focus on research published in the past two years. This multi-feature search approach makes it easier to narrow down large document collections while preserving the broader context of the collection. Thus, *ClusterChat* offers a scalable and interactive solution for large-scale text exploration by integrating multiple exploratory features into a single platform.

Section 2 provides a detailed overview of the system architecture of *ClusterChat*, followed by the description of real-world scenarios in Section 3, and finally, Section 4 discusses the practical impact on corpus exploration and future directions.

2 ClusterChat Architecture

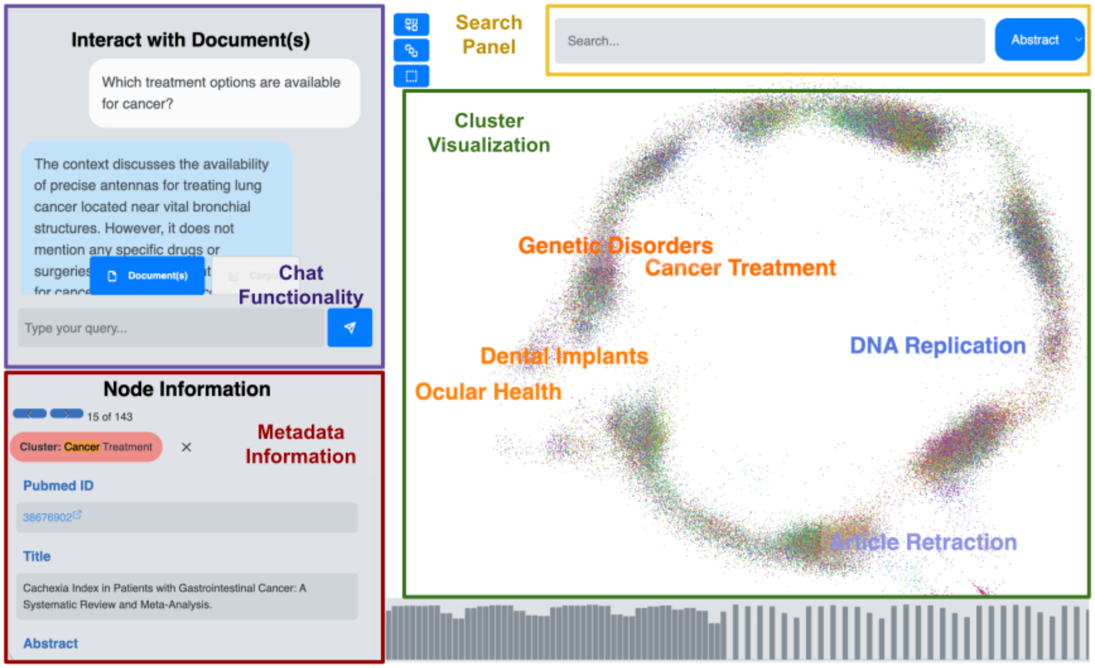
The architecture of *ClusterChat* is designed to offer scalable, multi-feature corpus exploration by integrating backend components with an intuitive and interactive frontend. The system combines topical document clustering, timeline-centric exploration, lexical and semantic search, and corpus and document-level QA, making it an end-to-end exploratory search platform. It uses BERTopic [3] and LangChain[6] in the backend, and the frontend is built on Cosmograph[7]. It involves visualizing clusters to provide users with an overview of a corpus and search functionality for interactive filtering of documents. Finally, users can ask queries on filtered documents or explore the corpus using natural language queries.

ClusterChat Backend. In order to demonstrate the functionalities of the *ClusterChat* system, we collected PubMed abstracts and their metadata (e.g., publication date, journal, authors) by leveraging the Entrez PubMed API[8] for the English language and stored the data in OpenSearch[9]. Approximately four million PubMed abstracts for the four-year time frame 2020-2024 have been collected. Each abstract is indexed in OpenSearch, allowing for lexical search (using BM25 [9]) and semantic search. This retrieval mechanism ensures that users can explore the corpus through traditional keyword searches and semantic exploration powered by embeddings.

PubMed abstracts are processed in two ways to compute embedding vectors. First, each abstract is converted into a 768-dimensional vector using the pre-trained language model PubMedBERT[10] [4], resulting in approximately 4 million embeddings. Second, each sentence of an abstract is converted into a 768-dimensional vector to enable QA, which will be discussed later in this section. González-Márquez et al. [2] performed pilot experiments to compare the performance of eight BERT variants and determined PubMedBERT as the best-performing model. Abstract embeddings are reduced using UMAP [8] to preserve local and global structures essential for meaningful clustering. Subsequently, the reduced embeddings are clustered using HDBSCAN [1], a density-based algorithm that identifies clusters of varying densities and effectively handles noise. Documents present in a cluster are analyzed using tf-idf (term frequency-inverse document frequency) to identify the cluster's keywords [11]. Finally, the GPT-4o-mini[11] model is prompted with keywords associated with each cluster to generate a label for a cluster. Cluster labels, centroid embeddings, and clusters associated with each PubMed abstract are stored in OpenSearch for efficient retrieval.

Given the scale of the dataset (~4M PubMed abstracts), it was computationally expensive to apply BERTopic to all documents simultaneously due to hardware limitations. To address the computational challenges, we adopted a federated learning-inspired segmentation strategy, which enabled scalable topic modeling without sacrificing semantic coherence. The dataset is segmented into non-overlapping 15-day intervals based on publication date, and BERTopic models are trained for each subset. This interval length is primarily determined by computational considerations, i.e., shorter intervals would have resulted in an increase in the number of models (for each timeframe) and thus storage overhead, while longer intervals would have led to excessively large subsets that exceeded hardware capabilities. Each model produced date range-specific topics. In order to construct a cohesive topic landscape, these models are merged into a unified topic structure through a multi-step process. This merging involved aligning and combining topic embeddings using a centroid-based strategy and dimensionality reduction

[6] https://www.langchain.com (accessed on 11th Apr 2025)
[7] https://cosmograph.app (accessed on 11th Apr 2025)
[8] https://www.ncbi.nlm.nih.gov/books/NBK25501 (accessed on 16th Oct 2024)
[9] https://opensearch.org (accessed on 23rd Oct 2024)
[10] https://huggingface.co/NeuML/pubmedbert-base-embeddings (accessed on 15th Oct 2024)
[11] https://platform.openai.com/docs/models/gpt-4o-mini (accessed on 12th Feb 2025)



Interact with Document(s)

Which treatment options are available for cancer?

The context discusses the availability of precise antennas for treating lung cancer located near vital bronchial structures. However, it does not mention any specific drugs or surgeries

Type your query...

Node Information

15 of 143

Cluster: Cancer Treatment

Pubmed ID

38676902

Title

Cachexia Index in Patients with Gastrointestinal Cancer: A Systematic Review and Meta-Analysis.

Abstract

Figure 1: Overview of the web-based *ClusterChat* interface. It includes four main features: 1) a chat panel on the top-left for corpus and document-level QA; 2) a metadata information panel on the bottom-left for displaying metadata information of the selected documents; 3) a cluster visualization map showing research topics like "Cancer Treatment" and "Genetic Disorders"; 4) a search panel at the top to perform a lexical and semantic search on "Abstract" text and a keyword search on the "Title" text.

via UMAP for a proper visualization. Tf-idf was employed to extract representative keywords, and concise labels and descriptions are generated using GPT-4o-mini for interpretability. The resulting topics are then organized into a hierarchical structure, where cosine similarity between cluster embeddings defines relationships across levels. Finally, the merged topics and their UMAP-based coordinates are indexed in OpenSearch for seamless integration with efficient question answering and corpus-level exploration. This iterative approach ensures that *ClusterChat* remains scalable while preserving the local and global coherence of topics across different date ranges.

For enabling QA on the corpus and document level, we implemented an Retrieval-Augmented Generation (RAG) pipeline. Each PubMed abstract is segmented into sentences to create granular chunks of text and is embedded using the PubMedBERT model. Approximately 46 million sentence embeddings and corresponding texts are indexed in OpenSearch. This allows for the retrieval of relevant sentences during the QA process. The system allows users to explicitly select between corpus-level and document-level QA modes via a toggle in the frontend interface. Users may ask questions about the entire corpus or a specific cluster when the corpus-level mode is selected. Corpus-level queries are passed to the Mixtral–8x7B model[12] to determine relevant cluster labels, which are then used to construct OpenSearch queries to retrieve cluster-specific information, i.e., topic words in cluster and cluster description, which are then summarized into an answer. However, when the document-level mode is selected, the system performs a filtering on the metadata to retrieve the top-10 most relevant sentences based on embedding similarity. On average, the query latency is about 2 seconds on 46 million embeddings. Retrieved sentences combined with the query are prompted to the Mixtral–8x7B model for answer generation. Based on the performance on biomedical texts, Mixtral–8x7B is the default model for answer generation in both QA modes. While the frontend design does not allow for selecting different LLMs at runtime, the backend architecture is modular and easily extensible to experiment with alternative LLMs such as GPT-4 or Claude.

ClusterChat Frontend. Figure 1 shows the *ClusterChat* interface designed to provide users with an interactive experience for exploring the corpus and obtaining answers to queries.

Users are presented with a visualization of the topic clusters derived from the abstracts. The embedding dimensionality is reduced to two dimensions using UMAP and displayed through Cosmograph. Each cluster reflects a grouping of semantically related abstracts, with proximity indicating semantic similarity. This high-level view allows researchers to explore specific areas and the broader corpus context. In addition to the cluster visualization, *ClusterChat* integrates lexical and semantic search as well as temporal filtering to refine the exploration process. Users can filter documents by publication date, keyword or semantic search, or cluster selection to narrow down results based on their research focus. For instance, a researcher can apply a publication date filter to analyze recent

[12] https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1 (accessed on 12th April 2025)

developments within a specific cluster or use keyword search to retrieve documents related to a particular treatment or biomarker. Timeline-centric exploration enhances this experience by visualizing document distributions across date ranges, allowing users to track research trends and shifts in focus. This timeline-based analysis helps researchers detect bursts of publication activity, such as the rapid growth of COVID-19 research between 2020 and 2022.

A distinctive feature of *ClusterChat* is its QA capability, powered by a RAG pipeline. Users can pose natural language questions at both the corpus and document level, enabling targeted information retrieval. For example, a corpus-level query on the entire corpus or selected cluster/s like "Which topics are covered in the corpus?" or "Which topics are covered in the Cancer Treatment cluster?" provide an overview of related themes across the corpus/cluster. In contrast, a document-level query such as "What targeted therapies are discussed for non-small cell lung cancer (NSCLC)?" retrieves semantically relevant sentences from the filtered documents, and an answer is generated based on these retrieved sentences. The generated answer is attributed to the relevant documents, highlighting the specific sources from which the relevant sentences are obtained, ensuring precision and relevance while reducing the need for manual document review.

3 Case Studies

To showcase the utility and effectiveness of *ClusterChat*, we conducted case studies using a large biomedical corpus of four million PubMed abstracts (2020–2024). Instead of relying on traditional evaluation metrics, which may not fully capture the exploratory capabilities of the system, we present two real-world usage scenarios. These scenarios demonstrate how *ClusterChat* facilitates interactive and flexible corpus exploration, enabling researchers to extract meaningful insights from large-scale text collections.

3.1 Scenario 1: Exploring Emerging Research Trends in the Cancer Treatment

Identifying emerging trends and recent advances is crucial in rapidly evolving fields such as cancer research. Traditional search engines are limited in providing a comprehensive overview of the landscape. *ClusterChat* addresses this gap by enabling researchers to navigate the corpus.

A researcher exploring advancements in cancer treatment, particularly for non-small cell lung cancer (NSCLC), begins by examining the high-level cluster map generated from the PubMed dataset. Within the "Cancer Treatment" cluster, they identify significant topics, such as immunotherapy, advancements in chemotherapy, and targeted drug therapies. To ensure they review the most recent research, the researcher applies a date filter to focus on studies published between 2023 and 2024. Using *ClusterChat*'s temporal filtering feature, they observe a notable spike in mid-2023, which may indicate key advancements, like novel immunotherapy approaches or breakthroughs in combination therapies. Intrigued by these trends, the researcher further refines her search with keyword filters, explicitly focusing on "immunotherapy in lung cancer", which results in a curated list of relevant documents. Rather than manually reviewing each article, the researcher takes advantage of the QA feature by querying: "What are the advancements in immunotherapy for non-small cell lung cancer (NSCLC)?" The system generates an answer, highlighting key findings from the retrieved documents and attributing them to their source documents. Through this iterative process of moving from high-level exploration to focused querying, the researcher gains a broad understanding of trends and detailed insights from the literature. This approach saves time and facilitates a more strategic and informed investigation into the latest developments in cancer treatment.

3.2 Scenario 2: Question Answering for Biomedical Queries

Integrating a RAG pipeline in *ClusterChat* provides researchers with a powerful tool for answering specific questions directly from the corpus. This feature bridges the gap between exploratory search and precise information retrieval.

Consider a scenario where a medical researcher specializing in genetic disorders seeks to answer the question: "What are the therapeutic approaches for managing cystic fibrosis?". A RAG system first retrieves semantically relevant PubMed abstract sentences, and then an LLM (Mistral) generates a concise, evidence-backed answer. The answer highlights key therapeutic approaches with attribution, including the PubMed IDs for source verification. This efficient QA capability accelerates the researcher's workflow by efficiently generating context-aware answers and reducing the time required for manual literature review. Additionally, the ability to attribute answers within *ClusterChat* allows researchers to validate findings and seamlessly investigate related concepts.

4 Discussion and Ongoing Work

ClusterChat addresses several limitations of existing corpus exploration systems by combining multiple features, such as clustering, lexical and semantic search, timeline-centric exploration, and QA. This integration allows researchers to explore large-scale text corpora more efficiently and interactively, enhancing decision-making and enabling more profound insights into their datasets. For instance, medical researchers can use *ClusterChat* to discover and explore emerging trends in specific areas, for example, non-small cell lung cancer (NSCLC), by navigating the 'Cancer Treatment' cluster and filtering about publication dates and keywords. Additionally, the QA feature enables users to obtain specific information from selected documents. While our case study focuses on biomedical literature, *ClusterChat* system architecture is domain-agnostic and can be applied to other large-scale text corpora, such as legal or financial documents. This adaptability makes it an ideal tool for researchers across multiple disciplines.

Several enhancements are planned to improve *ClusterChat* adaptability and user experience. Currently, clustering is performed at the backend and is static. Enhancing this functionality to dynamically cluster retrieved documents similar to $Carrot2$ and Knowledge Navigator will result in an even more intuitive and insightful experience.

Acknowledgments

We thank the Federal Ministry of Education and Research (BMBF) for funding this research within the FrameIntell project, https://frameintell.de.

References

[1] Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. 2013. Density-Based Clustering Based on Hierarchical Density Estimates. In *Advances in Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, 160–172.

[2] Rita González-Márquez, Luca Schmidt, Benjamin M Schmidt, Philipp Berens, and Dmitry Kobak. 2024. The landscape of biomedical research. *Patterns* 5, 6 (2024), 100968.

[3] Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint arXiv:2203.05794* (2022).

[4] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Trans. Comput. Healthcare* 3, 1 (2021), 1–23.

[5] Qiao Jin, Robert Leaman, and Zhiyong Lu. 2024. PubMed and beyond: biomedical literature search in the age of artificial intelligence. *eBioMedicine* 100 (2024), 104988.

[6] Uri Katz, Mosh Levy, and Yoav Goldberg. 2024. Knowledge Navigator: LLM-guided Browsing Framework for Exploratory Search in Scientific Literature. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics, 8838–8855.

[7] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems* 33 (2020), 9459–9474.

[8] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software* 3, 29 (2018), 861.

[9] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (2009), 333–389.

[10] Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. 2016. Embedding projector: Interactive visualization and interpretation of embeddings. *arXiv preprint arXiv:1611.05469* (2016).

[11] Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28, 1 (1972), 11–21.

[12] Zijie J. Wang, Fred Hohman, and Duen Horng Chau. 2023. WizMap: Scalable Interactive Visualization for Exploring Large Machine Learning Embeddings. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*. Association for Computational Linguistics, 516–523.

[13] Yuxiang Zheng, Shichao Sun, Lin Qiu, Dongyu Ru, Cheng Jiayang, Xuefeng Li, Jifan Lin, Binjie Wang, Yun Luo, Renjie Pan, Yang Xu, Qingkai Min, Zizhao Zhang, Yiwen Wang, Wenjie Li, and Pengfei Liu. 2024. OpenResearcher: Unleashing AI for Accelerated Scientific Research. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 209–218.