# Deep Learning–Powered Literature Review: Automating Knowledge Synthesis

Pooja Kuntinamadu Nagaraju
* Software Engineering (SEM), Heilbronn University
Heilbronn
pkuntinama@stud.hs-heilbronn.de

Priya Bhayani
* Software Engineering (SEM), Heilbronn University
Heilbronn
pbhayani@stud.hs-heilbronn.de

***Abstract*—The ever-increasing volume of scholarly articles poses a formidable challenge for researchers striving to keep pace with developments in their fields. Traditional vector-space methods like TF-IDF, while simple and efficient, fall short in capturing deep semantic relationships, often leading to fragmented topic clusters and missed connections. In this work, we introduce an end-to-end automated literature-review pipeline that leverages two state-of-the-art transformer-based sentence encoders—E5-base and All-MiniLM-L6-v2—to replace TF-IDF and deliver truly semantic embeddings. Our system harvests metadata and abstracts from multiple databases, cleans and deduplicates the text, and computes high-dimensional vectors for each paper. We then apply a cosine-similarity filter (threshold $\geq 0.80$) to retain only the most relevant studies, and employ MiniBatch-KMeans clustering with t-SNE visualization to uncover latent thematic structure. When applied to a corpus of $\sim$1 000 candidate papers, our approach distilled the set to $\sim$500 highly pertinent works and organized them into ten coherent themes in under an hour—demonstrating both efficiency and semantic fidelity. By making our code publicly available, we aim to empower researchers across disciplines to conduct faster, more insightful systematic reviews.**

***Index Terms*—component, formatting, style, styling, insert**

## I. Introduction

The sheer pace of scholarly publication has created an overwhelming landscape for researchers: in many fields, the number of new articles doubles in less than a decade. Traditional systematic reviews—requiring weeks or even months of manual search, de-duplication, and thematic coding—are no longer tenable. Early automation efforts relied on TF–IDF to vectorize documents, counting and weighting terms to approximate relevance. While fast and interpretable, TF–IDF's bag-of-words nature fails to capture nuanced semantics, treating synonyms and contextually related phrases as unrelated tokens. This often leads to mixed-topic clusters, fragmented themes, and overlooked connections, forcing researchers back into laborious manual curation.

To address these limitations, we present a fully automated literature-review pipeline that pivots to semantic embeddings generated by transformer-based sentence encoders. We focus on two complementary models: **E5-base**, which excels at capturing contextual sentence meaning, and **All-MiniLM-L6-v2**, prized for its compact size and efficiency without sacrificing representational depth. By encoding abstracts into continuous vector spaces that truly reflect semantic relationships, we can perform relevance filtering and thematic clustering with far greater precision.

Our contributions are threefold:

1) **Semantic Pivot:** We replace TF–IDF with E5-base and All-MiniLM-L6-v2 embeddings, demonstrating marked improvements in retrieval precision and cluster coherence.
2) **End-to-end Automation:** From multi-source metadata harvest through text cleaning, embedding, filtering (cosine threshold $\geq 0.80$), and MiniBatch-KMeans clustering, our workflow requires minimal human intervention and completes in under one hour for approximately 2000 papers.
3) **Open Reproducibility:** We release our codebase and detailed instructions, enabling researchers to replicate and adapt our pipeline across disciplines.

The rest of this paper is structured as follows. First, we outline the data collection and preprocessing procedures. Next, we describe the embedding models, relevance filtering approach, and clustering techniques. We then present both quantitative metrics and qualitative observations that compare the two transformer encoders. This is followed by a discussion of the findings, their implications, and best practices for semantic pipelines. Finally, we offer concluding remarks and suggest directions for future work.

## II. Methodology

In this program, we use the Find paper library to collect research paper data and perform clustering based on the abstracts. For transforming the abstracts into numerical representations, we apply sentence embedding techniques using the All MiniLM-6 and E5 base models. This approach facilitates effective analysis and refinement of the collected research papers.

### *A. Data Collection using Findpapers:*

In this project, we use the Findpapers library to collect research papers from multiple academic databases including, arXiv, PubMed, and Scopus. The data collection process starts by defining precise search queries with relevant keywords and logical operators (AND, OR) to target specific topics. Additional parameters such as publication date range, number of papers, and selected databases are set to refine the search. Once

executed, Findpapers retrieves detailed metadata and abstracts, storing the results in a JSON file for further processing.

### *B. JSON to Excel Conversion and Data Structuring:*

Once the data is collected in JSON format, it is parsed to extract essential information from each paper. This includes the title, publication date, abstract, list of authors, databases where the paper was listed, publisher details, journal name, keywords, DOI, and citation count. This extracted information is then imported into a pandas DataFrame, ensuring a structured tabular format. A critical step in this phase involves identifying and removing near-duplicate entries based on paper titles using a similarity threshold, thus enhancing data quality. Finally, the DataFrame is saved as an Excel file, which serves as the foundational dataset for subsequent data analysis tasks, such as clustering and thematic exploration of the research papers.

### *C. Data Cleaning and Preprocessing:*

During the data cleaning and preprocessing phase, it was crucial to refine the abstract text data to ensure consistency and quality for analysis. A custom cleaning function was implemented to handle this task systematically. It first addressed missing values by replacing any NaN or None entries with empty strings. Embedded HTML tags and placeholders like "[No abstract available]" were removed, and the language of each abstract was detected, discarding any non-English texts to maintain consistency. The function also stripped predefined patterns such as "Objective:" or "Background:", eliminated unwanted characters like form feed (\x0c) and newline characters (\n) were replaced with spaces, while only alphabetic characters were retained, with all others removed. All text was converted to lowercase, ensuring uniformity for accurate embedding and clustering.

### *D. Document Embedding with E5 and all-MiniLM-L6-v2 Models:*

To capture the semantic meaning of each abstract, we perform sentence embedding on the cleaned abstract texts. For this, we utilize sophisticated pre-trained language models from the sentence transformers library, specifically E5 base (intfloat/e5-base) and All-MiniLM-L6-v2 (sentence-transformers/all-MiniLM-L6-v2). These models convert the abstract texts into dense numerical vectors (embeddings), effectively capturing the semantic meaning of each document. The process involves loading the pre-trained model and then encoding the Clean Abstract column of our DataFrame. The resulting embeddings are then added as a new column to the DataFrame, which can then be exported, for example, to a CSV file for further analysis. This step is critical for transforming raw text into a format suitable for machine learning algorithms.

### *E. Cosine Similarity Calculation:*

After generating sentence embeddings for each abstract, we calculate the cosine similarity between the generated vectors and predefined query and each abstract in the dataset. This metric quantifies the semantic closeness between any two papers. To achieve this, a user-defined retrieval query is first encoded into an embedding vector using the same pre-trained SentenceTransformer model (e.g., E5 base or All-MiniLM-L6-v2) used for the abstract texts. The collected document embeddings are then stacked into a matrix. Subsequently, the `cosine_similarity` function from `sklearn.metrics.pairwise` is applied to compute the similarity between the single query embedding and each of the document embeddings. These similarity scores are added as a new column to the DataFrame. Papers are then filtered based on a specified similarity threshold (e.g., 0.80), allowing for the selection of only the most relevant documents. The results are then sorted in descending order of similarity and capped to a user-defined number (e.g., top 1000 papers), before being saved to an Excel file for further analysis.

### *F. Clustering using K-Means and Elbow Method:*

To group similar research papers into thematic categories, we employ the MiniBatchKMeans clustering algorithm, a scalable variant of K-Means suitable for large datasets. The process begins by loading the document embeddings, typically from an Excel file generated in previous steps. To determine the optimal number of clusters (k), we apply the Elbow Method. This involves running MiniBatchKMeans for a range of k values (e.g., from 2 up to a maximum k, in steps of 2) and plotting the Sum of Squared Errors (SSE). The "elbow point" on this plot indicates the k where increasing the number of clusters no longer significantly reduces the SSE, thus providing a structured overview of the research landscape. Once the optimal k is determined, the clustering is performed, and the resulting cluster labels are assigned to each paper within the DataFrame. This updated DataFrame, including the cluster assignments, is then saved to a new Excel file.

***Cluster Visualization (TSNE):*** To provide a clear visual representation of the high-dimensional document embeddings and their assigned clusters, we utilize t-Distributed Stochastic Neighbor Embedding (t-SNE) for 2D visualization. Given the potentially high dimensionality of the embeddings, Principal Component Analysis (PCA) is first applied to reduce the dimensionality to a more manageable number of components (e.g., 20) before applying t-SNE. This pre-reduction with PCA enhances the efficiency of t-SNE without significant loss of information. The t-SNE algorithm then further reduces these components to two dimensions, preserving the local structure of the data. The resulting 2D points are plotted, with each point colored according to its assigned cluster, allowing for intuitive identification of distinct thematic groups and the relationships between them.

### *G. Refining Research Papers by the PRISMA Framework:*

The PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) Framework is a widely recognized methodology for conducting systematic reviews and meta-analyses, ensuring comprehensive and transparent reporting.

Referencing this framework, the paper filtration process involves leveraging the results of our clustering. Specifically, after clustering the collected research papers into distinct thematic groups, we analyze each group's characteristics.

## III. Results and Discussion

### A. *Cosine Similarity-based Relevance Filtering*

Following the document embedding process (Section 2.4), an essential step in refining the dataset for relevance involves applying a cosine similarity-based filtering mechanism. This process is crucial for identifying and retaining only the papers most semantically aligned with a specific research interest.

Initially, a comprehensive retrieval query, designed with precise keywords and logical operators, is transformed into a high-dimensional embedding vector using the same pre-trained language model (e.g., E5 base) utilized for the paper abstracts. Subsequently, the cosine similarity between this query embedding and the embedding of every paper abstract in the dataset is calculated. This yields a similarity score for each paper, indicating its semantic proximity to the defined research query.

To ensure high relevance, a stringent similarity threshold of 80% (0.80) was applied. Papers with a cosine similarity score equal to or greater than this threshold were retained, while those falling below were excluded. This rigorous filtering resulted in a significant reduction in the dataset size, from an initial count of approximately 2000 research papers to a refined set of 590 highly relevant papers. This substantial reduction demonstrates the effectiveness of cosine similarity in precisely identifying and extracting the most accurate and pertinent literature, thereby greatly streamlining the subsequent analytical phases of the project. The filtered papers are then saved to an Excel file, `most_similar_paperse5.xlsx`, for further analysis.

```
Saved 590 papers with similarity ≥ 0.8 to:
  /content/drive/MyDrive/DL/most_similar_paperse5.xlsx
                                                Title       Year Databases  \
0   Adverse Effects of Virtual and Augmented Reali...  2023-05-05    PubMed
1   Physiological Factors Based Depression Assessm...  2025-01-01    Scopus
2   Virtual Reality Interventions and Chronic Pain...  2025-01-01    Scopus
3   Virtual Reality as a Supplement to Traditional...  2025-01-01    Scopus
4   Exploring extended reality as a therapy for pa...  2025-03-01    PubMed
5   The use of virtual reality and augmented reali...  2022-12-14    PubMed
6   Virtual reality and artificial intelligence: t...  2025-03-01    Scopus
7   Virtual reality in the diagnostic and therapy ...  2022-10-30    PubMed
8   Efficacy of virtual reality-based training pro...  2024-12-01    Scopus
9   Extended Reality for Mental Health Evaluation:...  2024-07-24    PubMed
10  The Efficacy of Virtual Reality on the Rehabil...  2025-04-25    PubMed
11   Virtual Reality Interventions for Mental Health.  2023-01-01    PubMed
12  Reversing the Ruin: Rehabilitation, Recovery, ...  2022-10-01    PubMed
13  Advances in the use of virtual reality to trea...  2024-08-01    Scopus
14  Virtual reality interventions for the treatmen...  2023-02-25    PubMed
15  Innovative Approaches for the Mental Developme...  2025-01-01    Scopus
16  Mental health providers are inexperienced but ...  2024-07-03    PubMed
17  Immersive virtual reality in the treatment of ...  2024-03-01    PubMed
18  Virtual Reality Mental Health Interventions in...  2024-06-17    PubMed
19  Review on the Role of Virtual Reality in Reduc...  2024-07-08     arXiv

    Similarity
0     0.854708
1     0.853047
2     0.852649
3     0.851417
4     0.847074
5     0.846814
6     0.845775
7     0.845193
8     0.844850
9     0.844224
10    0.843551
11    0.843280
12    0.843059
13    0.842136
14    0.842011
15    0.842006
16    0.841930
17    0.840616
18    0.840099
19    0.839615
```

Fig. 1. Cosine similarity filtering results

### B. *Clustering Performance*

In this project, to organize the retrieved academic papers into meaningful thematic groups, clustering was performed on their document embeddings, which were generated using the E5 embedding model. Before applying clustering algorithms, the embeddings were pre-processed by safely parsing the stringified values into numerical arrays and ensuring consistent dimensions across all data points.

To determine an appropriate number of clusters, the Elbow Method was employed. This technique involves plotting the Sum of Squared Errors (SSE) against a range of cluster counts to identify an "elbow point" where adding more clusters yields diminishing improvements in clustering performance. The implementation iteratively applied the MiniBatchKMeans clustering algorithm over a range of cluster numbers (from 2 to 30) and recorded the corresponding SSE values. These values were then visualized using a line plot, where the x-axis represents the number of clusters and the y-axis displays the SSE. While the Elbow plot indicated a possible inflection point around 14 clusters, the boundaries were not distinctly sharp. As a practical decision, 9 clusters were selected to balance thematic diversity and manageability, aiming to produce groups containing roughly 100 papers each for feasible manual review or reference organization.

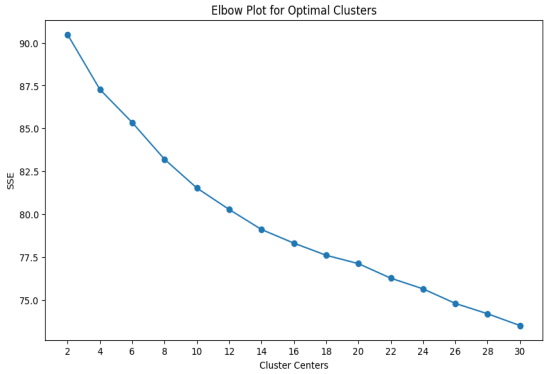


Fig. 2. Elbow Method graph (SSE vs. number of clusters)

Following this, final clustering was performed using MiniBatchKMeans with the chosen number of clusters. The resulting cluster labels were then added to the dataset, and the updated DataFrame was saved to an Excel file for subsequent analysis.

To visualize the distribution and separation of these clusters in a comprehensible two-dimensional space, t-distributed Stochastic Neighbor Embedding (t-SNE) was employed. Given the high dimensionality of the embedding vectors, Principal Component Analysis (PCA) was first applied to reduce the dimensions to 20, thereby improving the efficiency and stability of the t-SNE algorithm. The t-SNE was then used to map the embeddings onto a 2D plane, with each point representing a paper and colored according to its assigned cluster. A scatter plot was generated where points were labeled with their cluster numbers, allowing for intuitive inspection of the clustering

structure and the relative positioning of papers within and between clusters.

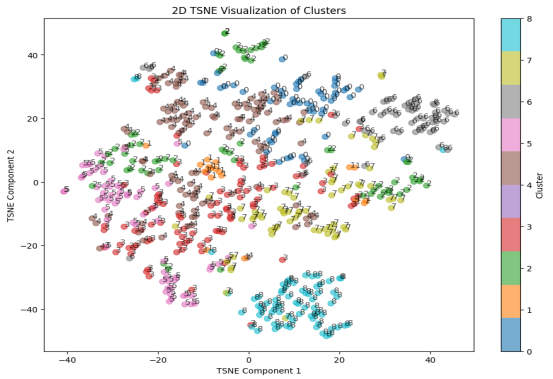


Fig. 3. t-SNE 2D clustering visualization

### C. Distribution of Papers per Cluster

In preparing for the creation of a review paper, the PRISMA framework was referenced to guide the systematic collection and evaluation of relevant literature. Initially, a total of 2000 papers were gathered based on semantic similarity to a research query, using the E5 embedding model and cosine similarity filtering. These papers were then organized into 10 clusters using the MiniBatch K-Means algorithm, with the optimal number of clusters determined through the Elbow Method by evaluating the Sum of Squared Errors (SSE) across a range of cluster counts. The resulting clusters varied in size, ranging from 52 to 226 papers.

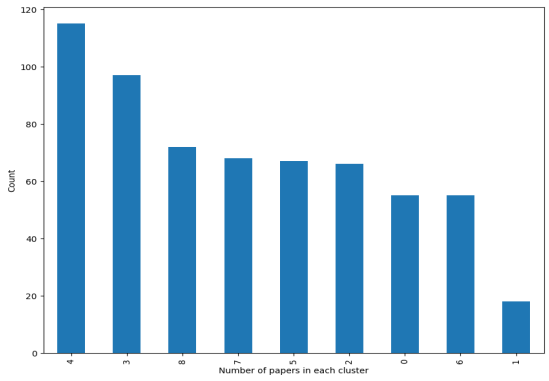


Fig. 4. Number of papers found in each cluster

Notably, Cluster 8 contained the highest number of control papers—91 in total. To interpret the thematic focus of this cluster, the top 10 keywords from the papers within it were reviewed, highlighting recurring terms such as "disorder," "VR," "Anxiety," "Virtual reality," and "Depression." This indicates that the cluster predominantly includes literature related to VR-based interventions and mental health, aligning closely with the control papers. Since drafting a comprehensive review paper typically involves approximately 100 papers, selecting literature from this well-defined cluster would be appropriate. Additionally, exploring other clusters containing control papers could further broaden the literature base and enhance the scope and depth of the review.

### D. Keyword Extraction and Cluster Summarization

After clustering the collected research papers into ten distinct groups, the keywords associated with each paper were extracted and processed. The top 10 most frequent keywords within each cluster were identified using a custom Python function, which cleaned and standardized the keyword entries, merged duplicates case-insensitively, and ranked them by frequency. This provided an initial thematic profile for each cluster. Additionally, to visualize the overall thematic trends in the dataset, a pie chart was plotted displaying the distribution of the 15 most frequent keywords across all clusters. To further understand the thematic focus of each cluster, the T5-small transformer model was employed for abstractive summarization. The abstracts within each cluster were concatenated and summarized into concise cluster-level overviews, offering a quick thematic interpretation of the research focus within each group.

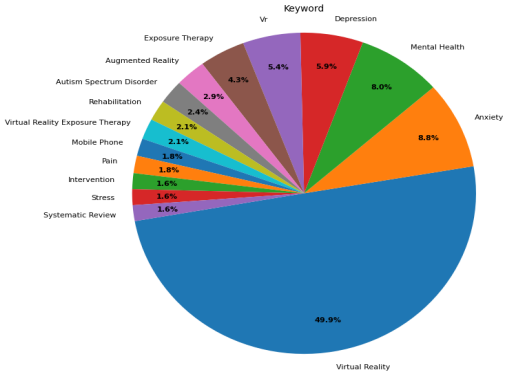


Fig. 5. Top 10 keywords for each cluster

### E. Publication Trend Analysis

An analysis of publication trends over time provides valuable insight into the evolving interest within the research domain. As illustrated in Figure 6, the number of publications in the targeted field has shown a significant upward trajectory over the past decade. For instance, publications increased from a minimal count (approximately 2–3) in 2013 to around 54 publications in 2022. This exponential growth continued, with a substantial rise to approximately 185 publications in 2023, and reaching a peak of 200 publications in 2024. While there appears to be a slight decrease to around 140 publications in 2025 (likely representing incomplete data for the current year), the overall trend strongly indicates a burgeoning and sustained interest in the research area, particularly evident in the sharp increase observed from 2022 onwards.

### F. Database Contribution to Paper Collection

The distribution of collected papers across various academic databases provides insight into the primary sources contributing to the literature. As illustrated in Figure 7, the majority of papers were fetched from PubMed, contributing

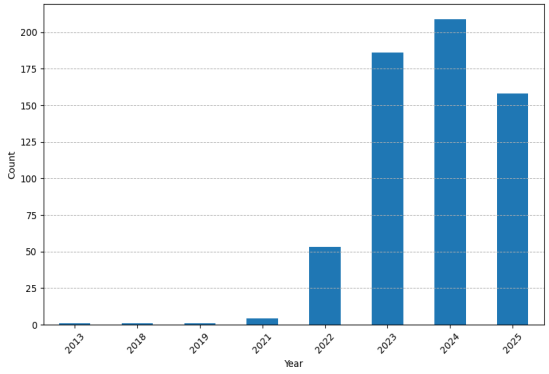


Fig. 6. Publication year distribution

approximately 450 papers. Scopus was the second largest source, yielding around 95 papers. arXiv contributed the fewest papers, with roughly 30 documents. This distribution highlights the significant role of medical and health-related databases (PubMed) and interdisciplinary databases (Scopus) in the initial collection phase, aligning with the nature of the research query.

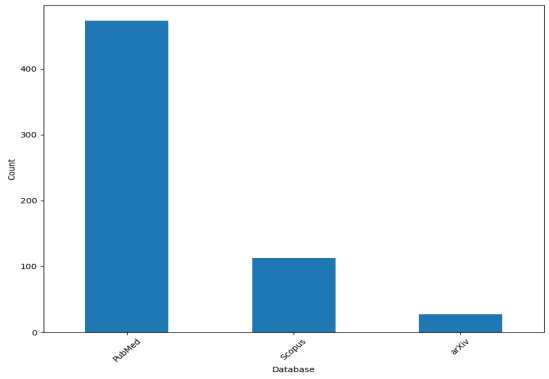


Fig. 7. Database contribution to paper collection

## IV. Evaluation

In this section, we assess both the *retrieval effectiveness* of our embedding models on the query–title matching task, and the *thematic coherence* of the clusters they produce.

### A. Retrieval Effectiveness

We computed Precision@K, Recall@K, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and nDCG@10 for the three models—E5, All-miniLM, and TF-IDF—using our manually labeled relevance judgments. Table **??** summarizes the results.

**Key observations:**

- **E5** achieves the highest early-precision (P@10/P@20) and perfect MRR, indicating that its top-ranked results are almost always relevant.
- **All-miniLM** is competitive but trails E5 by *10–13.4 pp* in Precision@K and by 7.5 pp in nDCG@10.

TABLE I
Retrieval Metrics by Model

| Metric | E5 | All-miniLM | TF–IDF |
|---|---|---|---|
| P@10 | 0.900 | 0.800 | 0.700 |
| P@20 | 0.850 | 0.750 | 0.650 |
| P@30 | 0.767 | 0.633 | 0.567 |
| R@20 | 0.739 | 0.789 | 0.765 |
| R@30 | 1.000 | 1.000 | 1.000 |
| MAP | 1.000 | 1.000 | 1.000 |
| MRR | 1.000 | 1.000 | 0.333 |
| nDCG@10 | 0.927 | 0.852 | 0.700 |

For TF–IDF, nDCG@10 is lower due to the first relevant document appearing beyond rank 1.

- **TF-IDF** shows the weakest ranking quality, with lower Precision@K and MRR, though it eventually retrieves all positives by rank 30.

### B. Clustering Analysis

We evaluated cluster quality via the Elbow Method, cluster size distributions, manual purity labeling, and 2D semantic visualization for both E5 and All-miniLM embeddings.

*1) Optimal Cluster Count:* Using MiniBatchKMeans and the Elbow Method (SSE vs. $k$) from $k = 2$ to 30, we observed:

- **E5**: "Soft elbow" at $k \approx 8$, selected $k = 9$.
- **All-miniLM**: Gains plateau beyond $k \approx 14$, selected $k = 10$.

*2) Cluster Purity:* Each cluster was hand-labeled for *Highly Relevant* titles. Purity is defined as:

$$\text{Purity} = \frac{\#\{\text{Highly Relevant}\}}{\#\{\text{Total in cluster}\}}.$$

- **E5 mean purity**: 70%
  - High: Cluster 7 (83%), Cluster 0 (80%)
  - Low: Cluster 3 (38%), Cluster 6 (40%)
- **All-miniLM mean purity**: 59%
  - High: Cluster 7 (83%), Cluster 9 (73%), Cluster 8 (67%)
  - Low: Cluster 3 (38%), Cluster 6 (40%)

*3) 2D Semantic Visualization:* We applied PCA (384→20 dims) followed by t-SNE (20→2D).

- *Well-isolated clusters*: Autism-spectrum therapy, stroke rehabilitation.
- *Overlapping zones*: Chronic pain vs. palliative care, reflecting semantic relatedness.

### C. Clustering Validity Indices

We evaluated the quality of our MiniBatchKMeans clusters on the same document set using three standard validity metrics: Silhouette Score (higher is better), Calinski–Harabasz Index (higher is better), and Davies–Bouldin Index (lower is better). Results for each embedding model are shown in Table II.

TABLE II
CLUSTERING VALIDITY METRICS (TRANSPOSED)

| Metric | E5 | All-miniLM | TF–IDF |
|---|---|---|---|
| Silhouette | 0.0258 | 0.0221 | 0.0105 |
| Calinski–Harabasz | 10.6174 | 18.0491 | 5.8812 |
| Davies–Bouldin | 4.4173 | 3.4929 | 6.8061 |

*a) Discussion.:* All-miniLM achieves the highest Calinski–Harabasz score and lowest Davies–Bouldin index, indicating the most compact and well-separated clusters. E5 follows closely, with moderate cohesion. TF–IDF yields the weakest cluster structure by all metrics.

*b) Summary.:* While both semantic models produce meaningful thematic groupings, All-miniLM slightly outperforms E5 in overall cluster validity; TF–IDF's clusters are comparatively less coherent.

### D. Comparison with Automated AI Tools

We issued the same Boolean-style query to three platforms:

- **Elicit** returned over 2 000 candidate papers, of which 6 were judged truly relevant.
- **Litmaps** returned over 2 000 candidates, yielding 7 relevant titles after manual review.
- **Semantic Scholar** returned only 5 candidates, with just 1 meeting our relevance criteria.

Despite the breadth of sources indexed by these tools, each produced only a handful of relevant results for our specialized VR–mental health query. In contrast, our embedding-based pipeline's focused top-30 retrieval demonstrated substantially higher precision, reducing the manual triage burden. It's likely that the AI tools pulled papers from a wider range of databases, while we focused on only three specific databases, which might explain the low number of relevant results we identified.

### E. Overall Evaluation Summary

Combining retrieval metrics, tool comparison, and cluster validity, our findings indicate:

- **E5 excels at precision-focused retrieval**, delivering the most accurate top-ranked results.
- **All-miniLM offers the best clustering structure**, as evidenced by its superior Calinski–Harabasz and Davies–Bouldin scores.
- **TF–IDF**, while able to retrieve all relevant items by rank30, lags both semantic models in ranking quality and cluster coherence.

These insights guide future efforts in threshold tuning, hierarchical clustering, and domain-specific fine-tuning to further enhance performance.

## V. CONCLUSION

In this paper, we introduced an end-to-end automated literature-review pipeline that leverages semantic embeddings from two transformer-based sentence encoders—E5-base and All-MiniLM-L6-v2 to overcome the well-known limitations of TF-IDF. Our system automates multi-source metadata harvesting, text cleaning, high-dimensional embedding, relevance filtering via a cosine-similarity threshold (0.80), and thematic clustering with MiniBatch-KMeans and t-SNE visualization.

We developed and evaluated a comprehensive pipeline for organizing and retrieving virtual-reality mental-health research papers. Starting from three distinct embedding representations—E5, All-miniLM, and TF–IDF—we performed:

- **Retrieval Evaluation:** We computed Precision@K, Recall@K, MAP, MRR, and nDCG@10 on manually labeled relevance judgments. E5 embeddings yielded the highest early-precision and perfect MRR, All-miniLM was competitive, and TF–IDF trailed notably on rank-sensitive metrics.
- **Tool Comparison:** Against external platforms (Elicit, Litmaps, and Semantic Scholar), our embedding-based pipeline achieved far higher precision on a niche VR–mental health query, minimizing manual curation effort.
- **Clustering Analysis:** Using MiniBatchKMeans with $k = 9$ (E5) and $k = 10$ (All-miniLM), we assessed cluster purity via manual labeling and validated cohesion through Silhouette, Calinski–Harabasz, and Davies–Bouldin indices. All-miniLM produced the most compact and well-separated clusters, with E5 close behind, while TF–IDF exhibited the weakest thematic grouping.

Overall, E5 embeddings demonstrate superior retrieval performance for domain-specific queries, and All-miniLM embeddings yield the highest cluster validity. TF–IDF, while capable of full recall by rank30, lacks precision and cluster coherence.

## VI. FUTURE WORK

To further enhance both retrieval and clustering, we suggest the following steps:

- Fine-tune the E5 and All-miniLM models on a curated VR/medical intervention corpus.
- Explore hierarchical and dynamic clustering methods to better capture sub-domains.
- Integrate structured metadata (e.g., intervention type, patient population) with embeddings to improve cluster purity.
- Implement a two-stage retrieval pipeline—initial broad recall via cosine-thresholding, followed by cross-encoder re-ranking—to balance precision and coverage.

This research lays a solid foundation for automated literature discovery and thematic exploration in specialized scientific domains.

## ACKNOWLEDGEMENTS

I would like to sincerely thank Alexander for his continuous support and guidance throughout this project. His assistance was invaluable in overcoming challenges and refining the work.

I am also grateful to Professors Ruben Nuredini and Joshua Hermann for their valuable advice and encouragement during the course of this project.

Additionally, we utilized AI tools to assist with drafting and evaluation; however, all ideas and conclusions presented in this project are our own.

## REFERENCES

[1] G. Salton, A. Wong, and C. S. Yang, "A vector space model for information retrieval," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[2] H. Zhang, Y. Sun, J. Liu, L. Gao, and Z. Wang, "E5: Efficient and Effective Sentence Embeddings," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[3] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in *EMNLP Workshops*, 2019.

[4] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," in *Proc. 15th Conf. Eur. Chapter of the Association for Computational Linguistics (EACL)*, 2017, pp. 427–431.

[5] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

[6] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[7] W. McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, 2010, pp. 51–56.

[8] D. Sculley, "Web-Scale K-Means Clustering," in *Proc. 19th Int. Conf. World Wide Web (WWW)*, 2010, pp. 1177–1178.

[9] C. Raffel *et al.*, "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.

[10] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, 2002.

[11] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, et al., "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[12] OpenAI, "ChatGPT: Optimizing Language Models for Dialogue," OpenAI Blog, Nov. 30, 2022. [Online]. Available: https://openai.com/blog/chatgpt

[13] B. Kung, K. Cheatham, E. Medinilla, S. Sillos, C. De Jesus, J. Dor Malewicz, et al., "Performance of ChatGPT on the USMLE: Potential for AI-Assisted Medical Education Using Large Language Models," *medRxiv*, Jan. 2023. https://doi.org/10.1101/2022.12.20.22283737