Scientific Concept Evolution Tracker

Nic Bolton
University of Toronto
Toronto, Canada
nic@cs.toronto.edu

Abstract

With the sheer volume of research being published, the history and context of how scientific ideas evolve are often difficult to visualize through the noise. Terminology in science can be dynamic—the semantic meaning of terms such as "neural networks", "entropy", or "plasma" shift significantly over decades as new research subfields emerge. Traditional information retrieval systems and static vector databases index semantic meanings as fixed points in high-dimensional space, which flattens the temporal dimension and hides the evolutionary history of these concepts. This report introduces the Scientific Concept Evolution Tracker (SCET), a comprehensive system designed to ingest, index, and analyze large-scale scientific corpora to quantify this semantic drift. SCET is built for scale with PostgreSQL for storing metadata and Milvus for embeddings. We introduce a methodology that combines unsupervised clustering (K-Means) with temporal segmentation (Decision Tree Regression) to automatically identify distinct "eras" of a concept's life cycle. We demonstrate the system's capabilities through case studies, such as the divergence of "Transformer" from electrical engineering to natural language processing, and provide a quantitative analysis of system performance on a dataset sourced from arXiv.

1 Introduction

Science is a cumulative endeavor, yet the language of science is fluid. A core challenge in bibliometrics and the "Science of Science" is understanding how scientific consensus and terminology evolve. For a researcher entering a new field, understanding the historical context of a term is as critical as understanding its current definition. For example, a query for "Attention" in 2005 would yield results dominated by cognitive psychology and neurobiology. The same query in the late 2010s and early 2020s is overwhelmingly dominated by field of machine learning. This is the phenomenon that we are interested in, that is, being able to quantify how a concept's relevance or association with specific fields change over time.

The introduction and combination of Large Language Models (LLMs) and vector databases have revolutionized semantic search. By representing text as dense vectors in a high-dimensional space, we can capture semantic similarity beyond simple keyword matching. However, most vector search implementations treat the document corpus as a static snapshot. They are designed to answer the question, "What is semantically similar to this query now?" rather than "How has the meaning of this query changed over time?".

This project addresses the following question. How can we design a scalable vector database system that can quantify the semantic evolution of scientific concepts and identify pivotal publications that drive these shifts?

SCET was developed to answer this question. It consists of a pipeline that:

(1) Ingests and indexes scientific abstracts using a hybrid embedding strategy
(2) Retrieves context-aware results using a weighted hybrid search
(3) Clusters results into sub-concepts associated with a given query
(4) Produces a set of time periods that represent "stable" eras of a sub-concepts association/relevancy
(5) Identifies papers that are "pivotal" to the shift into these eras

The remainder of this paper is organized as follows. We first review the background and related work in NLP and bibliometrics. We then discuss the details of the methodology and system architecture, followed by a presentation of the experimental results and case studies. We conclude by discussing limitations and future work.

2 Background and Related Work

The problem to solve (tracking the evolution of scientific concepts) sits at the intersection of Natural Language Processing (NLP), Information Retrieval (IR), and the "Science of Science". This section reviews the historical progression of these fields and the specific technologies that enable SCET.

2.1 The Evolution of Information Retrieval

The field of Information Retrieval has evolved through several distinct paradigms. An overview of the history is listed below. It is important to note that although these systems perform well within their own goals, they all lack the ability to capture semantic and contextual meaning within terms.

2.1.1 Boolean Logic. The earliest IR systems relied on semantics used within set theory, where queries were built with operators such as AND, OR, NOT. These systems could make retrievals at a high level of precision by using Inverted Indices. These work as a key-value store, that is, each term is a key that is associated with a list of pages that contain the term. This results in a very fast and precise system [12].

2.1.2 TF-IDF. Term Frequency Inverse Document Frequency (TF-IDF) introduced the concept of weighting the importance of words. This system involves calculating a score for a term, derived by balancing how frequently a term appears in a specific document (TF) against how rarely the term appears across the entire collection of documents (IDF). Consequently, these scores often favour the terms that best characterize the topics involved in a collection of documents. [6].

2.1.3 Vector Space Model. The Vector Space Model (VSM) generalized the idea of weighting terms by representing documents as

vectors in a multi-dimensional space, where each dimension corresponds to a distinct term in the corpus. In this model, the relevance of a document to a query is measured by the similarity between their respective vectors. The similarity measurement is often done using Cosine Similarity

$$\cos(\boldsymbol{q}, \boldsymbol{d}) = \frac{\boldsymbol{q} \cdot \boldsymbol{d}}{\|\boldsymbol{q}\|\|\boldsymbol{d}\|} \tag{1}$$

for TF-IDF weights of the query $\boldsymbol{q}$ and the document $\boldsymbol{d}$.

This allows for ranked retrieval results based on the angle between vectors, rather than a binary inclusion/exclusion. However, traditional VSMs treat terms as orthogonal dimensions, meaning they cannot capture semantic relationships between synonyms (e.g., car and automobile) without explicit expansion [6].

2.1.4 BM25. Okapi Best Matching 25 (BM25) represents a significant evolution in probabilistic information retrieval. While sharing similarities with TF-IDF, BM25 introduces two critical improvements: term saturation and document length normalization. Unlike TF-IDF, where the score increases linearly with term frequency, BM25 applies a saturation function so that the value of a term diminishes as it is repeated (thus, preventing long repetitions of keywords from dominating results). Additionally, it penalizes long documents (which naturally contain more terms) to prevent them from unfairly dominating search results. BM25 remains a strong baseline for lexical search tasks today.[11].

2.2 Evolution of NLP Representations

The representation of text is fundamental to our ability to measure drift.

2.2.1 Static Embeddings. Word2Vec [8] and GloVe [9] introduced distributed representations. They rely on the distributional hypothesis: "a word is characterized by the company it keeps". These models map discrete tokens to dense vectors in a continuous space, capturing syntactic and semantic regularities. The vectors that are assigned to these tokens (words) are derived so that they are in close proximity to other similar words (for example, the vectors for "dog" and "puppy" are very close). However, these models are static, meaning they assign a fixed vector to each word type regardless of context. For example, a vector would be generated for the word "bank", but would be indifferent whether it was used in the context of river banks or piggy banks. This imposes a significant limitation for scientific text, where acronyms and terms often have distinct or field-specific definitions that cannot be derived via static representation.

2.2.2 Contextual Embeddings. ELMo [10] and BERT [4] introduced dynamic embeddings to address the limitations of static models. The Transformer [13] architecture's self-attention mechanism allows the representation of a token to be a function of its surrounding context. BERT (Bidirectional Encoder Representations from Transformers) pre-trains on a masked language modeling objective, allowing it to learn deep bidirectional representations. Unlike static embeddings, BERT generates a representation for a token that is conditioned on the entire input sequence. This allows a word such as "bank" to become associated with the context it was used in—if it sees "river" nearby, then it can identify river bank as the likely association. This concept is useful for our problem, as it allows for distinguishing "Attention" (psychology) from "Attention" (machine learning) in a given corpus.

2.2.3 Domain-Specific Models. Although contextual embedding models such as BERT are great drivers for advancing the field of NLP, they often underperform on domain-specific corpora such as scientific text. This is because they are trained on corpora such as Wikipedia, which differs significantly in vocabulary and syntax from scientific literature. SciBERT [2] addresses this by pre-training BERT on a large corpus of scientific papers. SPECTER [3] took this further by leveraging a unique feature of research: citations. They used citation graphs as a signal for semantic similarity, which groups papers based on how related they are as opposed to just textual overlap. It uses a triplet loss objective:

$$\mathcal{L} = \max\left\{d(q, p^+) - d(q, p^-) + m, 0\right\}$$

where d is a distance function, q is a query paper, p^+ is a cited paper, p^- is a non-cited paper (but may be cited by p^+), and m is the loss margin hyperparameter. By training the model to pull cited papers closer in vector space and push unrelated papers apart, 2 learns embeddings that reflect the functional relationships between papers.

2.3 Vector Database Indexing

Searching a dataset of millions of high-dimensional vectors is computationally prohibitive using brute force ($O(N)$). SCET relies on Approximate Nearest Neighbor (ANN) algorithms.

2.3.1 Inverted File Index. The Inverted File Index (IVF) works by partitioning the vector space into Voronoi cells, where every document vector is then assigned to its nearest centroid. Now upon search, the system can extract a subset of the vectors by comparing the query vector to the centroids. A benefit to IVF is its low memory footprint [6].

2.3.2 Hierarchical Navigable Small World. While partition-based methods like IVF offer memory efficiency, graph-based approaches currently provide the superior trade-off between latency and recall. Hierarchical Navigable Small World (HNSW) structures data into a multi-layered graph hierarchy inspired by Skip Lists and the "small world" phenomenon [14]. The upper layers consist of sparse, long-range links that allow the search algorithm to traverse the vector space rapidly, effectively zooming in on the target region. Once the coarse location is identified, the search descends to lower, denser layers for fine-grained greedy traversal to locate the nearest neighbors. Although HNSW requires higher memory overhead to store the graph connectivity compared to quantization methods, it is robust against the curse of dimensionality and does not require the training phases of clustering approaches [5].

2.4 Semantic Drift Analysis

Semantic drift (or semantic change) is the study of change with respect to the meaning of words, specifically the evolution of how a word is used. For example, the word *awful* originally meant to inspire wonder or fear, and hence impressive. Today, it is used to describe something that is regarded as very bad.

2.4.1 Alignment. Since embedding models are initialized randomly, the vector spaces for different time periods often end up rotated or flipped relative to each other. Orthogonal Procrustes Analysis is used to fix this, by mathematically rotating the vector space of one time period to align it with the next. By locking the two maps together, it ensures that if a word's position changes, it represents a genuine shift in meaning rather than a side effect of the model's random initialization [7] .

2.4.2 Dynamic Word Embeddings. While alignment-based methods rely on independent training of temporal slices, Dynamic Probabilistic Models treat the evolution of semantic meaning as a continuous latent variable process. First introduced by Balmer and Mandt [1], their process involves training a single global model where the embedding of a term at time t is conditioned on its embedding at time $t-1$, modeled as a Gaussian Random Walk. This process ensures that meanings shift gradually rather than abruptly. By using data from surrounding time periods, these models work much better for rare terms. This eliminates the random noise often seen in year-by-year training, resulting in a clearer, smoother path of how a concept has changed.

3 Methodology

3.1 System Architecture and Data Pipeline

3.1.1 Data Ingestion Layer.

3.1.2 Vector Storage Layer (Milvus).

3.2 Hybrid Embedding Strategy

3.2.1 Dense Embedding (Semantic).

3.2.2 Sparse Embedding (Lexical).

3.2.3 Hybrid Scoring.

3.3 Concept Clustering Algorithm

3.4 Era Detection

3.5 Identifying Pivotal Papers

4 Experimental Results

4.1 System Performance Benchmarks

4.1.1 Ingestion Scalability.

4.1.2 Query Latency.

4.2 Case Studies

4.2.1 Transformer.

4.2.2 Corona.

4.3 Ablation Study: Hybrid vs. Dense-Only

5 Discussion

5.1 The Time Machine Effect

5.2 Scalability vs. Depth Trade-off

5.3 Limitations

6 Conclusion

Acknowledgments

Acknowledgements go here. Delete enclosing begin/end markers if there are no acknowledgements.

References

[1] Robert Bamler and Stephan Mandt. 2017. Dynamic Word Embeddings. arXiv:1702.08359 [stat.ML] https://arxiv.org/abs/1702.08359

[2] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: Pretrained Language Model for Scientific Text. In *EMNLP*. arXiv:arXiv:1903.10676

[3] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. 2020. SPECTER: Document-level Representation Learning using Citation-informed Transformers. arXiv:2004.07180 [cs.CL] https://arxiv.org/abs/2004.07180

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs.CL] https://arxiv.org/abs/1810.04805

[5] Yu. A. Malkov and D. A. Yashunin. 2018. Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs. arXiv:1603.09320 [cs.DS] https://arxiv.org/abs/1603.09320

[6] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, USA.

[7] Lucas Maystre, Alvaro Ortega Gonzalez, Charles Park, Rares Dolga, Tudor Berariu, Yu Zhao, and Kamil Ciosek. 2025. When Embedding Models Meet: Procrustes Bounds and Applications. arXiv:2510.13406 [cs.LG] https://arxiv.org/abs/2510.13406

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [cs.CL] https://arxiv.org/abs/1301.3781

[9] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Alessandro Moschitti, Bo Pang, and Walter Daelemans (Eds.). Association for Computational Linguistics, Doha, Qatar, 1532–1543. doi:10.3115/v1/D14-1162

[10] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. arXiv:1802.05365 [cs.CL] https://arxiv.org/abs/1802.05365

[11] Stephen Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. 0–.

[12] Gerard Salton, Edward A. Fox, and Harry Wu. 1983. Extended Boolean information retrieval. *Commun. ACM* 26, 11 (Nov. 1983), 1022–1036. doi:10.1145/182.358466

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. arXiv:1706.03762 [cs.CL] https://arxiv.org/abs/1706.03762

[14] Wikipedia contributors. 2025. Small-world experiment — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Small-world_experiment&oldid=1320552337 [Online; accessed 12-December-2025].