

International Journal of Information Sciences and Engineering | ***ISSN 1694-4496***
Published by AIR JOURNALS | *https://airjournal.org/ijise*
16/18 Avenue des Longaniers, Quatre Bornes, Mauritius
airjournals@gmail.com; enquiry@airjournal.org

SCHOLARLY ARTICLE

Reliability-Aware Hybrid Phrase-Semantic Clustering for Document Retrieval: A Reproducible Benchmark, Ablation Study, and Stress-Test Framework

Bob Chile-Agada [1], Paul Ndudiri Ohia [2], Ihim Kingsley Chukwunazom [3], Sunday Adebayo Alabi [4], & Ohia, Obi [5]

[1 3]Alvan Ikoku Federal University of Education Owerri
[2] Department of Information Technology University of Port Harcourt
[4] National Open University of Nigeria
[5] Department of Computer & Robotics Education, School of Secondary Education (Science), Federal College of Education (Technical) Omoku, Nigeria

Citations - APA

Chile-Agada, B., Ohia, P. N., Chukwunazom, I. K., Alabi, S. A., & Ohia, O. (2026). Reliability-Aware Hybrid Phrase-Semantic Clustering for Document Retrieval: A Reproducible Benchmark, Ablation Study, and Stress-Test Framework. *International Journal of Information Science and Engineering*, *10*(2), 1-16
https://doi.org/10.5281/zenodo.24304940

ABSTRACT

Document clustering can make search results easier to explore, but phrase and semantic representations have different failure modes. Phrase n-grams produce inspectable labels yet can miss paraphrases, whereas latent or transformer representations can connect vocabulary variants but may be harder to explain. This study develops a reproducible hybrid clustering prototype that fuses phrase TF-IDF, a semantic representation, and lexical TF-IDF. The package includes K-means and agglomerative baselines, three channel ablations, a fusion-weight sweep, five controlled synthetic conditions, machine-readable assignments, metadata, visualisations, and tests. The bundled corpus contains 400 synthetic documents across four topical domains, with generated paraphrase and noise variation. In the primary run, several methods attain near-perfect partition scores; this is interpreted as a ceiling effect caused by the deliberately structured synthetic corpus, not evidence of broad superiority. The default semantic channel uses TF-IDF with truncated singular value decomposition (SVD); an optional SentenceTransformers mode is evaluated separately. The phrase component uses contiguous n-grams and is not a faithful compressed suffix-tree STC implementation. The contribution is a transparent, reproducible benchmark scaffold for evaluating complementarity, sensitivity, and failure modes. Validation on public real-world corpora, multiple independent seeds, transformer-specific runs, and human retrieval tasks remains necessary before drawing general conclusions.

Keywords: *Document Clustering; Phrase-Based Retrieval; Semantic Similarity; TF-IDF; Latent Semantic Representation; Sentence Embeddings; Ablation Study; Reproducibility; Information Retrieval*

This article is licensed under the creative commons license 4.0. Copyright© The Authors

Introduction

Search interfaces often return a ranked list that requires users to inspect documents sequentially. Clustering offers an alternative exploratory view: it can group related results and expose multiple topical paths through a collection. Early systems such as Scatter/Gather and web document clustering engines demonstrated the utility of grouping search results into navigable themes (Cutting et al., 1992; Zamir & Etzioni, 1998, 1999). However, no single representation is universally adequate. Lexical methods are transparent and efficient but can be brittle when related documents use different words. Semantic representations can encode contextual relationships, yet their clusters can be harder to label and can obscure the concrete textual evidence supporting a grouping.

The distinction matters for document retrieval because the system must do more than assign documents to groups. It should produce groups that are coherent enough to help exploration, provide labels grounded in document content, and remain computationally manageable. Phrase-based methods can offer labels directly from repeated word sequences, while semantic representations can help connect paraphrases. A hybrid method is therefore plausible, but the mere combination of feature channels does not guarantee improvement. Similarity channels may be redundant, poorly calibrated, or harmful when their weights are unsuitable.

Modern topic and clustering systems provide useful context. BERTopic combines transformer-based document embeddings, clustering, and class-based TF-IDF representations to generate interpretable topics (Grootendorst, 2022). Recent studies continue to investigate combinations of embeddings, topic representations, and network structure for document clustering (Ortu, 2025; Amiri & Karimianghadim, 2024). Work on clustering long texts also highlights that representation strategy can materially affect which information survives model input constraints (Ortakci & Borhan, 2025). These lines of work motivate an empirical rather than purely architectural question: under what conditions do phrase, lexical, and semantic channels complement one another?

Problem Statement

The problem addressed is the lack of a clearly reproducible, controlled comparison that isolates the contribution of phrase, semantic, and lexical similarity within one document-clustering pipeline. A hybrid score may perform well because one channel dominates, because the channels provide complementary information, or because the synthetic corpus happens to align with the representation. Without ablation and sensitivity analysis, these explanations cannot be separated.

Aim and Objectives

i. Develop a transparent hybrid phrase-semantic similarity pipeline for document clustering.
ii. Compare the hybrid method with lexical K-means, semantic K-means, phrase-only agglomerative clustering, and single-channel agglomerative baselines.
iii. Quantify the effect of removing individual similarity channels through ablation experiments.
iv. Evaluate sensitivity to fusion weights and controlled corpus conditions involving paraphrase and noise.
v. Package data generation, code, outputs, metadata, plots, and tests so that the experiments can be reproduced.
vi. Define a defensible follow-up protocol for external datasets and human evaluation.

Research Questions

i. RQ1: How does the hybrid method behave relative to lexical, semantic, and phrase-only baselines on the included synthetic corpus?
ii. RQ2: How sensitive are cluster outcomes to the relative weights assigned to phrase, semantic, and lexical similarity?
iii. RQ3: Does removing one channel materially change cluster quality across controlled paraphrase and noise conditions?

iv. RQ4: What evidence would be required to establish that a hybrid approach is useful on real search-result collections?

Contributions and Boundaries

The contributions are (a) a modular reference implementation, (b) a multi-method benchmark runner, (c) channel ablations and weight sensitivity, (d) five controlled synthetic conditions, (e) reproducibility metadata and exported assignments, and (f) a limitations-first reporting structure. This paper does not claim a new transformer architecture, a faithful classical STC implementation, or demonstrated real-world superiority. The phrase channel uses contiguous word n-grams; the default semantic channel uses TF-IDF followed by SVD. Optional transformer embeddings are an extension and require a separate experiment.

Background and Related Work

Document Clustering and Exploratory Retrieval

Document clustering groups texts without requiring a fixed category label for every incoming document. The cluster hypothesis in information retrieval suggests that documents similar to one another are more likely to be relevant to the same information need (Hearst & Pedersen, 1996). Scatter/Gather operationalised clustering as an interactive browsing process, while web clustering systems sought to organise result sets into meaningful categories (Cutting et al., 1992; Carpineto et al., 2009; Zamir & Etzioni, 1999). These systems motivate evaluating both cluster structure and the usefulness of the labels presented to users.

Lexical and Phrase Representations

TF-IDF weights terms by their importance within documents relative to a collection and remains a standard baseline for information retrieval (Manning et al., 2008). N-gram features extend lexical representations to contiguous phrases, allowing the model to preserve local word order. Such features are interpretable because high-weight phrases can be shown directly to users. Their limitation is vocabulary mismatch: “solar power generation” and “electricity produced from sunlight” may be semantically related without sharing a discriminative phrase. Phrase evidence can also overemphasise boilerplate or generic expressions unless preprocessing and frequency thresholds are carefully chosen.

Classical Suffix Tree Clustering (STC) is a phrase-based method that identifies shared phrases and combines base clusters according to overlap. Suffix trees are a well-studied data structure for substring processing (McCreight, 1976; Ukkonen, 1995). The implementation in this project does not construct a suffix tree and should not be described as classical STC. It uses n-gram TF-IDF as a transparent phrase proxy. This terminology is important for reproducibility and fair comparison.

Semantic Representations

Latent semantic indexing projects a term-document matrix into a lower-dimensional space, seeking latent associations beyond exact term matching (Deerwester et al., 1990). Truncated SVD is a common computational approach to this low-rank approximation. Transformer-derived sentence embeddings offer another route: texts are encoded into dense vectors whose geometric proximity is intended to reflect semantic similarity. Reimers and Gurevych (2019) introduced Sentence-BERT to make semantically meaningful sentence embeddings practical for similarity comparison. Yet embeddings depend on model choice, domain, input length, and the training objectives of the underlying model. A result obtained with an SVD fallback must not be reported as a transformer result.

Hybrid Clustering and Topic Representations

BERTopic provides a prominent pipeline that clusters transformer embeddings and then describes clusters with class-based TF-IDF (Grootendorst, 2022). Other work combines topic modelling, embeddings, and graph or network structure to address limitations of sparse bag-of-words representations (Amiri & Karimianghadim, 2024). Ortu (2025) presents a multi-dimensional similarity-network approach that combines embeddings with other document-level dimensions. SUMEX addresses semantic similarity and explanation generation (Saeed et al., 2024). These approaches indicate that combining multiple signals is an active area, so a credible contribution must be specific: a transparent fusion rule alone is not sufficient evidence of novelty.

Evaluation of Clustering

External metrics compare cluster assignments with known labels, while internal metrics assess structure without using ground-truth labels. Adjusted Rand Index (ARI) compares pairwise co-membership while adjusting for chance (Hubert & Arabie, 1985). Normalised mutual information and related measures quantify agreement between partitions. Silhouette scores assess whether observations are closer to their own cluster than to alternatives (Rousseeuw, 1987). No single metric fully captures retrieval usefulness, cluster label quality, or computational cost. This study reports multiple external metrics and runtime, but synthetic labels remain an artificial reference and cannot replace external benchmarks or user studies. Recent work on use-oriented evaluation shows how human judgements and carefully validated automated proxies can complement partition metrics (Hoyle et al., 2025).

Methodology

Research Design

The study is an experimental software-engineering investigation using a generated, labelled corpus. The corpus generator deliberately controls four topical domains, paraphrase frequency, generic noise, and a small proportion of cross-topic distractor sentences. Labels are available because the data are generated from topic-specific templates. This supports controlled debugging and stress tests but creates a known limitation: the corpus does not represent the diversity of natural scientific abstracts or search results. The evaluation is therefore exploratory and reproducible, not a final claim about generalisation.

Corpus Construction

The bundled primary corpus contains 400 documents, with 100 documents per topic: renewable energy, machine learning, public health, and agriculture. Each document combines multiple topic-specific sentences. Some are selected from paraphrase pools to reduce direct lexical overlap, while others use core topic sentences. Generic cross-domain statements are injected at controlled rates. The stress suite contains five additional conditions: low noise and low paraphrase, moderate noise and paraphrase, high paraphrase, high noise, and high noise combined with high paraphrase. The generator uses fixed seeds and exports the corpus as CSV.

The labels are synthetic by construction. Because documents are generated from topic-specific phrase pools, models that use similar features can achieve unusually high scores. This is a deliberate test fixture rather than an unbiased benchmark. A future study should use at least two public real-world corpora with documented licensing, label structure, preprocessing, and train/test or validation protocols.

Text Representations

Three representations are constructed. First, phrase TF-IDF uses contiguous word n-grams of length two through four. Second, lexical TF-IDF uses unigrams and bigrams. Third, the default semantic representation applies truncated SVD to the lexical TF-IDF matrix and normalises the resulting vectors. An optional SentenceTransformers mode uses the all-MiniLM-L6-v2 model and normalised transformer embeddings. The SVD representation is a latent lexical

representation and must not be treated as equivalent to contextual transformer embeddings. All modes are recorded in metadata.

Hybrid Similarity

For documents d_i and d_j, let S_p be cosine similarity over phrase TF-IDF, S_s semantic similarity, and S_l lexical similarity. The fused similarity is:

$$S_h(d_i, d_j) = \alpha S_p(d_i, d_j) + \beta S_s(d_i, d_j) + \gamma S_l(d_i, d_j) \quad (1)$$

$$\alpha \geq 0, \ \beta \geq 0, \ \gamma \geq 0; \quad \alpha + \beta + \gamma = 1 \quad (2)$$

The fusion weights are constrained to be nonnegative and sum to one, as formalised in Equation (2). The reference configuration uses $\alpha = 0.45$, $\beta = 0.35$, and $\gamma = 0.20$. These are illustrative initial settings, not empirically optimised parameters.

For each representation $r \in \{p, s, l\}$, the channel-specific similarity between documents d_i and d_j is the cosine similarity of their representation vectors, as expressed in Equation (3).

$$S_r(d_i, d_j) = \cos(x_i, x_j) \quad (3)$$

The fused similarity is converted to dissimilarity for average-linkage clustering using Equation (4). Similarity values are clipped to [0, 1], and each matrix diagonal is set to 1 before the distance is formed.

$$D_h(d_i, d_j) = 1 - S_h(d_i, d_j) \quad (4)$$

The requested cluster count is fixed at four to align with the synthetic reference labels. This controlled setting does not test automatic cluster-count selection.

Baselines and Ablation

The benchmark includes word TF-IDF K-means, semantic K-means, phrase-only agglomerative clustering, semantic-only agglomerative clustering, lexical-only agglomerative clustering, and the full hybrid method. Three ablations remove phrase, semantic, or lexical similarity and renormalise the remaining weights. The design is intended to expose whether the full method's performance is sensitive to any single channel. Because the synthetic data are constructed from topic templates, baseline performance should not be generalised to arbitrary corpora.

Weight Sensitivity

A grid sweep varies phrase and semantic weights over coarse increments while treating the lexical weight as the remaining nonnegative mass. For each valid combination, the program exports ARI, NMI, and the resulting cluster count. This is a descriptive sensitivity analysis, not a hyperparameter optimisation on held-out data. Selecting the best weight from the same labelled synthetic corpus would produce optimistic estimates. A stronger protocol would tune weights on a validation set and report final scores on a separate test set.

Metrics

i. ARI: chance-adjusted agreement between predicted and reference partitions.
ii. AMI: adjusted mutual information between predicted and reference partitions.
iii. NMI: normalised mutual information; interpreted alongside ARI because metrics respond differently to partition structure.
iv. Homogeneity: whether each predicted cluster contains documents from one reference class.
v. Completeness: whether documents from each reference class are assigned to the same predicted cluster.

vi. V-measure: harmonic mean of homogeneity and completeness.
vii. Silhouette: internal separation score computed from precomputed distance; interpretation depends on the selected distance representation.
viii. Runtime: measured wall-clock time for the clustering operation; it is machine- and load-dependent and excludes some representation construction in the per-method rows.

Reproducibility and Implementation

The code is organised as a Python module with a command-line interface. It exports benchmark_metrics.csv, cluster_assignments.csv, cluster_phrase_labels.json, weight_sensitivity.csv, and run_metadata.json. Metadata include the random seed, corpus size, semantic representation, software versions, and SHA-256 digest of the input file. Automated tests cover weight normalisation, invalid weights, similarity bounds, and expected cluster count. The package also includes controlled condition results and plots. Exact runtime comparisons should only be made on the same machine and execution environment.

Computational Complexity

The reference implementation materialises dense pairwise similarity matrices, requiring $O(n^2)$ memory and pairwise computation for n documents, in addition to representation and clustering costs. This is suitable for moderate research corpora but not millions of documents. A scalable successor should construct sparse k-nearest-neighbour graphs or use approximate nearest-neighbour search, and should report peak memory, indexing cost, query/update latency, and quality trade-offs.

Experimental Results

Results are generated by the included code. They should be read as controlled synthetic-corpus observations, not as external benchmark evidence. The metrics table below is inserted directly from the execution output of the packaged experiment. Timing values are environment-specific.

Table 1: ***Primary Benchmark and Ablation Results on the Synthetic Corpus***

Method	ARI	NMI	Homogeneity	Completeness	Clusters	Runtime (s)
TF-IDF K-means	1.000	1.000	1.000	1.000	4	0.1760
Semantic K-means	1.000	1.000	1.000	1.000	4	0.0239
Phrase-only agglomerative	1.000	1.000	1.000	1.000	4	0.0189
Semantic-only agglomerative	0.993	0.990	0.990	0.990	4	0.0023
Lexical-only agglomerative	0.993	0.990	0.990	0.990	4	0.0022
Hybrid full	1.000	1.000	1.000	1.000	4	0.0085
Ablation no phrase	0.993	0.990	0.990	0.990	4	0.0045
Ablation no semantic	1.000	1.000	1.000	1.000	4	0.0021
Ablation no lexical	1.000	1.000	1.000	1.000	4	0.0027

Note. ARI = adjusted Rand index; NMI = normalized mutual information. Runtime values are the recorded clustering-stage wall times from this environment, excluding representation construction and some preparation steps; they are not controlled hardware comparisons.

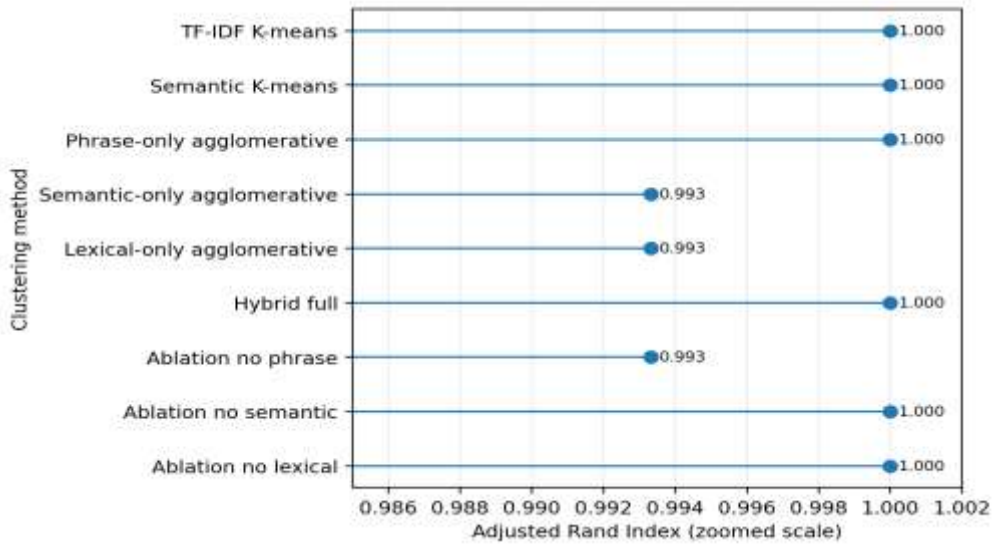


Figure 1: *Adjusted Rand Index for baseline methods and ablations in the primary synthetic corpus*
Note. The horizontal axis is intentionally zoomed to show small differences near 1.0; the scale does not start at zero.

Interpreting the Primary Benchmark

Figure 1 visualises the Adjusted Rand Index (ARI) values reported in Table 1. The primary benchmark provides a controlled check that the pipeline can run all specified representations, clustering methods, and metrics. Because the four classes are generated from distinct vocabularies and repeated topic templates, near-perfect scores are possible and should not be interpreted as evidence that the hybrid method solves lexical mismatch in natural text. The comparison is most useful as a software validation and a way to inspect differences among feature channels under known conditions. The paper intentionally avoids a claim of superiority unless the results show a robust and independently validated advantage.

Ablation Analysis

The ablation rows in Table 1 remove one similarity channel at a time. A channel can be redundant when the synthetic corpus strongly exposes its topic words; removing it may have little impact. Conversely, a performance decline suggests that the channel contributes information in that condition. A convincing ablation conclusion requires replication over independent corpora and seeds, plus uncertainty estimates. One single synthetic corpus cannot establish a universal contribution.

Weight Sensitivity

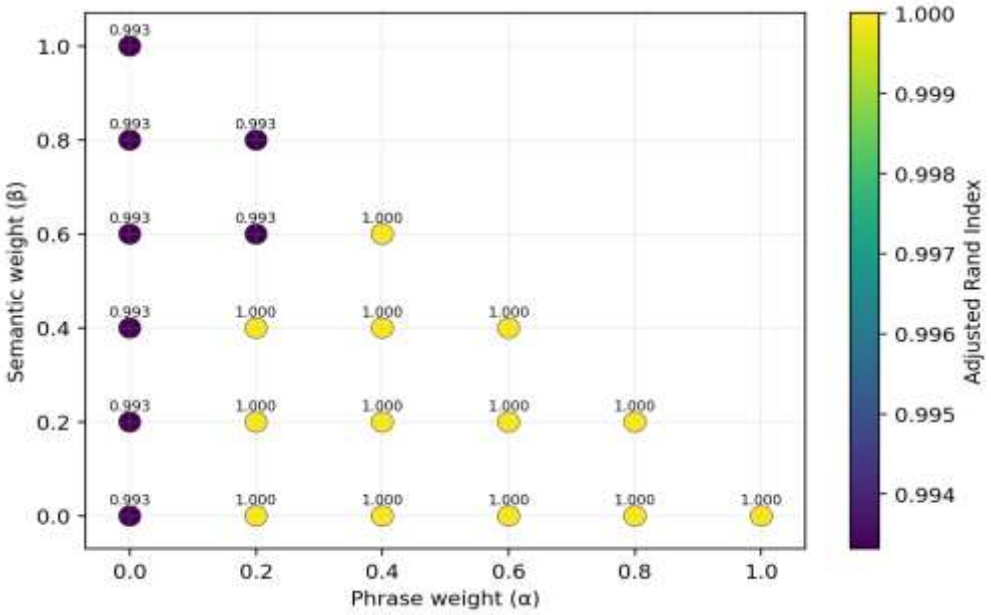


Figure 2.: *Sensitivity of clustering agreement to phrase and semantic fusion weights*

Note. The lexical weight is the remaining nonnegative mass, γ = 1 − α − β. Only valid weight combinations are plotted.

Figure 2 plots the grid of phrase and semantic weights, with lexical weight defined as the remainder. The purpose is to show whether the observed outcome is stable across plausible settings or depends on a narrow configuration. A flat response surface would indicate redundancy under the present corpus; sharp changes would indicate sensitivity. Neither pattern alone establishes real-world performance. Weights should be selected using validation data, not chosen from the test results.

Controlled Stress Conditions

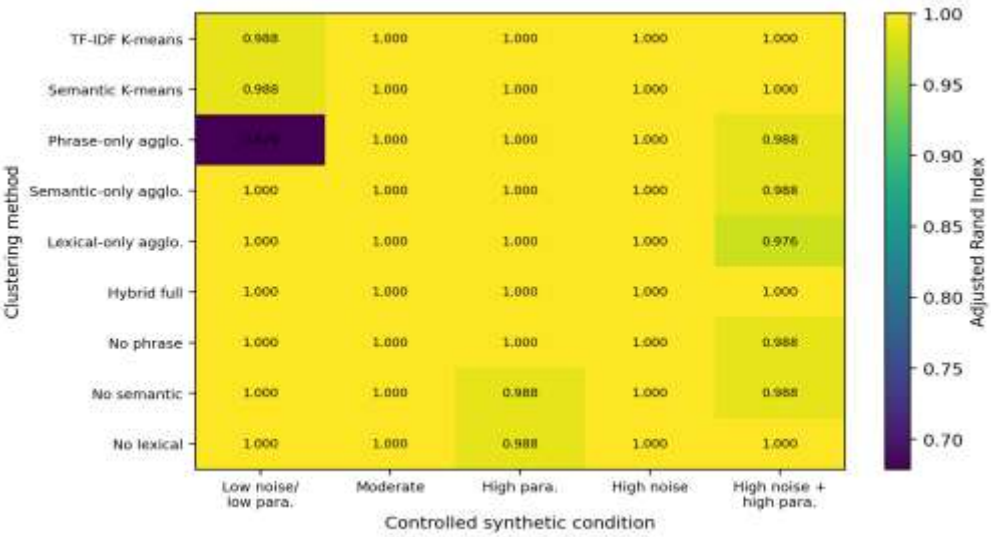


Figure 3.: *Adjusted Rand Index across five controlled synthetic stress conditions.*

Note. Conditions vary the generator's paraphrase and noise rates. Axis labels abbreviate "paraphrase" as "para."; "low noise/low para." and "high noise + high para." name the corresponding joint conditions.

Figure 3 summarises the Adjusted Rand Index across the five controlled conditions. The stress suite varies paraphrase and noise rates to examine whether methods respond differently as surface overlap decreases or distractor text increases. The conditions are generated from shared templates and are therefore not independent samples of natural language. They should be treated as controlled diagnostic scenarios. A more rigorous study would repeat each condition across multiple seeds, report means and confidence intervals, and include external datasets with established labels.

Cluster Labels and Traceability

The program extracts the highest-scoring phrase n-grams for each predicted cluster and exports them in JSON. These labels are inspectable and can be traced to the phrase representation. However, the current label generator does not perform a human-rated coherence assessment and does not guarantee that a phrase is representative of every document in a cluster. Phrase alignment should therefore be regarded as a descriptive aid rather than a validated human-interpretability measure.

Discussion

What the study Establishes

The work establishes a reproducible experimental scaffold for combining phrase, latent-semantic, and lexical similarity in a single clustering pipeline. It shows how to run baselines, isolate channels, inspect weight sensitivity, generate cluster labels, and preserve outputs and metadata. These engineering contributions matter because clustering papers can otherwise present a single aggregate score without exposing how features or parameters affect the result.

What the Study does not Establish

The study does not establish that the proposed hybrid method is novel relative to all existing systems, that it outperforms BERTopic or current embedding-based approaches, or that it improves human search effectiveness. The bundled experiment uses synthetic labels and a fixed number of clusters. It does not include a public external corpus, a human user study, statistical significance testing over independent corpora, or a memory-scalability benchmark. Those are limitations of the current evidence, not details that should be concealed in a submission.

Why Fusion may Help

Phrase features provide direct lexical evidence and interpretable labels. Semantic vectors may connect paraphrases, while lexical TF-IDF preserves exact topical terms that a compressed representation could dilute. Fusion is most likely to help when the channels have complementary errors. It may fail when the semantic model is domain-mismatched, the phrase vocabulary is dominated by boilerplate, or the lexical channel overwhelms semantic evidence. The value of fusion must therefore be measured through ablations and stress tests rather than assumed from the design.

Relationship to Earlier Project Work

This manuscript differs from a system-design paper about phrase-based retrieval and from a paper focused primarily on explainable evaluation. Its proposed focus is the complementarity of three similarity channels, their ablations, and their sensitivity to controlled changes in text. However, overlap in code, generated corpora, or methodological descriptions should be disclosed when papers are submitted. The current paper should not reuse another manuscript's results as though they were independently obtained. If the authors develop a faithful classical STC implementation or a real-corpus evaluation, that would materially strengthen the distinction and contribution.

Threats to Validity

i. Construct validity: synthetic topic labels measure recovery of generator-defined categories, not necessarily useful search-result groups.
ii. Internal validity: different algorithms have different computational pathways; runtime is affected by caching, representation construction, and environment load.
iii. External validity: four synthetic topics cannot represent the range of scientific, web, or institutional documents.
iv. Conclusion validity: a single corpus and coarse weight grid do not support broad claims of superiority.
v. Implementation validity: n-gram phrase extraction is not a suffix-tree algorithm; SVD is not a transformer embedding.
vi. Human utility: no controlled user study tests whether cluster labels improve navigation or task completion.

Operationalising a Reliability-Aware Extension

The present implementation uses a fixed weighted sum because it is transparent and easy to reproduce. A future reliability-aware extension could estimate how trustworthy each channel is for a particular document pair, rather than assuming a single global weight is appropriate for every pair. For example, phrase evidence may be unreliable when a document is extremely short or contains boilerplate, while an embedding channel may be unreliable for a highly specialised domain outside the embedding model's training distribution. Any adaptive mechanism must be defined before evaluation and compared against the fixed-weight reference; it should not be retrofitted after observing the test labels.

A possible reliability-aware extension is a pair-dependent mixture, given in Equation (5), rather than a fixed global weighting rule. The weights might depend on observable, label-free diagnostics such as phrase coverage, document length, vector norm, out-of-vocabulary rate, or agreement between channels. This design is a research hypothesis rather than a result of the current experiments. It must not use reference topic labels at inference time, and its weight functions would need to be documented and unit-tested.

$$S^*(i, j) = w_p(i, j)S_p(i, j) + w_s(i, j)S_s(i, j) + w_l(i, j)S_l(i, j) \quad (5)$$

A practical reliability signal is channel disagreement. If phrase similarity is high while semantic similarity is low, the pair may share boilerplate or an ambiguous phrase. If semantic similarity is high while phrase overlap is low, the documents may be paraphrases or the embedding model may be overgeneralising. Disagreement should not automatically imply that one channel is wrong; instead it can be exposed as an uncertainty flag for downstream inspection. Future work can compare (a) fixed fusion, (b) validation-tuned global fusion, and (c) a constrained pair-adaptive fusion rule under the same held-out protocol.

A Stronger Evaluation Protocol for a follow-up Study

A publication-grade evaluation should separate development, tuning, and final reporting. First, select external corpora with clear provenance and licences. The corpus selection should represent distinct conditions, such as short web snippets, scientific abstracts, and longer technical documents. Second, pre-register the preprocessing decisions, baseline methods, clustering-count policy, primary metric, and statistical comparisons. Third, tune weights only on development data. Fourth, evaluate the frozen system on held-out data, with no repeated tuning based on the held-out scores. Fifth, repeat stochastic algorithms across several seeds and report distributions rather than the single best run.

Where reference labels are available, report ARI and AMI as primary external measures and NMI, homogeneity, completeness, and V-measure as supporting measures. If labels are hierarchical or overlapping, a flat partition metric may be insufficient and should be supplemented with an appropriate evaluation. Internal metrics such as silhouette should not be treated as substitutes for external evaluation: a geometrically separated partition can still be semantically unhelpful. For retrieval-oriented claims, evaluate whether cluster navigation helps users locate relevant

items, using a controlled task and measures such as time to target, success rate, number of inspected documents, and perceived usefulness.

Uncertainty estimates should respect the unit of sampling. Resampling individual documents from a single synthetic generator can underestimate uncertainty because documents share templates. Better evidence comes from independent corpora, independently generated topic vocabularies, or domain-level replications. When comparing methods on the same corpus, use paired comparisons and report effect sizes with confidence intervals. Correct for multiple comparisons if many methods, metrics, and parameter configurations are tested. Publish all planned comparisons, not only the strongest result.

Ablation Interpretation and Common Analytical Traps

Ablation experiments are most informative when the feature channels are removed without inadvertently changing unrelated parts of the pipeline. This implementation renormalises the remaining fusion weights when a channel is removed, which is preferable to leaving the total similarity scale arbitrary. Even so, removing a channel changes the distribution of pairwise similarities and can alter hierarchical merge order. The observed difference is therefore an effect of the full representation-and-clustering pipeline, not a pure causal estimate of one feature in isolation.

Three analytical traps deserve attention. First, a known cluster count favours algorithms that are instructed to produce that count; the follow-up should separately test automatic cluster-number or density-based approaches. Second, perfect scores on a templated synthetic corpus can indicate that the task is too easy rather than that the algorithm is excellent. Third, the best weight combination selected from the same dataset used for reporting will be optimistically biased. The included weight sweep is descriptive and should not be used to claim an optimal configuration. These constraints should appear in the abstract or limitations section whenever the study is submitted.

Software Architecture and Reproducibility Checklist

i. Input contract: require document identifier, text, and a reference label for labelled evaluation; reject or explicitly log missing fields.
ii. Preprocessing record: preserve the raw text and log normalisation, tokenisation, stop-word, and n-gram settings.
iii. Model record: record semantic model name, version or revision, embedding dimension, pooling behaviour, and normalisation.
iv. Experiment record: record random seed, cluster count, fusion weights, distance metric, linkage rule, package versions, and execution command.
v. Output contract: export per-document assignments, cluster labels, all metric values, and machine-readable metadata.
vi. Failure handling: distinguish an algorithm failure from a metric that is undefined for a particular partition.
vii. Performance record: state whether runtime includes vectorisation, embedding generation, matrix construction, clustering, and label extraction.
viii. Integrity check: hash the input corpus and preserve a copy of the configuration used for the reported result.

Scalability and Deployment Considerations

A production system should avoid materialising every pairwise similarity for a large corpus. If n is the number of documents, a dense float32 matrix alone uses approximately $4n^2$ bytes, excluding temporary arrays and library overhead. This grows rapidly: at 100,000 documents, a single dense matrix would require roughly 40 GB before accounting for additional matrices. A scalable implementation should instead build a sparse similarity graph by retrieving a small set of approximate nearest neighbours per document. Clustering can then operate on that graph, and phrase labels can be calculated from cluster members after assignment. The sparse design introduces its own trade-offs: approximate neighbours can omit useful links, graph thresholds affect connectivity, and memory usage depends on the chosen degree and index.

Operational evaluation should include end-to-end latency, peak resident memory, indexing time, incremental update cost, and the time required to produce explanations. If embeddings are precomputed, report both embedding time and clustering-only time. If the model is used in a search engine, also measure query-to-cluster latency and how often clusters change as new documents arrive. These are different performance questions and should not be collapsed into one runtime figure.

Relevance to Scholarly Publishing and Research Discovery

A hybrid phrase-semantic approach could be investigated for organising scientific abstracts, journal articles, theses, conference papers, and institutional repository records. Phrase labels may expose domain-specific terminology, while semantic representations may connect papers that use different names for related methods. Such an application would need additional safeguards: abstracts may omit methods described in full text, metadata quality varies, citations create relationships not represented by textual similarity, and disciplinary vocabulary can shift over time. The present synthetic study does not demonstrate effectiveness in scholarly publishing; it only provides a prototype that could be evaluated in that setting.

For scholarly discovery, a future benchmark should compare text-only clustering with a citation-aware graph baseline, while avoiding leakage from citation or category labels into the text representation. It should evaluate whether researchers can find relevant articles faster and whether cluster labels are sufficiently precise to support browsing. Human evaluation should use blinded conditions, balanced tasks, and a transparent rubric. If participants are involved, the study should follow the relevant ethics and consent requirements.

Research Integrity and Novelty Positioning

The hybrid combination of phrase features and embeddings is an active and populated research area. Therefore, the novelty claim should be narrow and evidence-based. The present deliverable contributes an implementation and controlled evaluation scaffold, not a proven state-of-the-art method. A future paper should articulate its distinction from BERTopic, embedding-plus-graph clustering, and recent hybrid similarity frameworks by specifying the exact algorithmic innovation, not merely listing the components. Any shared code, generated data, figures, or wording across related manuscripts should be disclosed and cited. Independent results must be regenerated from the frozen final code before submission.

Algorithmic Workflow and Reproducible Decision Points

The executable pipeline can be described as a sequence of explicit decisions. First, validate the input schema and preserve document identifiers. Second, normalise text only through the documented vectoriser settings, while retaining the original text for audit. Third, construct the phrase, lexical, and semantic representations independently. Fourth, compute cosine similarity within each representation and combine the channels using a normalised weight vector. Fifth, transform similarity into a nonnegative distance matrix and apply average-linkage agglomerative clustering. Sixth, calculate external and internal metrics, extract representative phrase labels, and export assignments. Finally, save the run configuration and input hash. Each stage is independently inspectable, which helps separate a representation defect from a clustering or evaluation defect.

The fixed four-cluster setting is a deliberate experimental control, not a claim that four is the correct number of clusters for arbitrary data. In deployment, the cluster count could be estimated through a validation-based selection rule, stability analysis, or a density-based method that permits noise points. These alternatives have different assumptions and should not be substituted without documenting their consequences. A comparison should include both a known-k setting, which isolates representation quality, and an unknown-k setting, which evaluates the broader operational problem.

Failure Cases that should be Examined in External Validation

A realistic error analysis should sample both correct and incorrect assignments rather than inspect only the best clusters. Useful failure categories include: (a) paraphrase miss, where related documents share little lexical material; (b) generic phrase attraction, where common phrases cause unrelated texts to merge; (c) domain collision, where the same term has different meanings across fields; (d) topic mixture, where a document genuinely belongs to multiple subjects; (e) long-document dilution, where a single dominant topic hides secondary topics; and (f) cluster fragmentation, where a coherent topic is split into several groups. Each sampled error should be accompanied by the top phrases, channel-specific similarity values, and the final fused similarity. This trace makes it possible to understand why a result occurred rather than merely record that a metric fell.

For a user-facing retrieval tool, a cluster should also communicate uncertainty. A cluster label generated from only one repeated phrase may be less reliable than a label supported by several distinctive phrases and a consistent set of member documents. A future confidence indicator could combine label frequency, label distinctiveness, within-cluster cohesion, and separation from the nearest competing cluster. Such a score would need calibration against human judgements; it should not be called a probability unless its probabilistic interpretation is validated.

Statistical Reporting and Reproducibility beyond a Single Run

A robust final experiment should run each stochastic baseline over a predeclared seed set and report mean and standard deviation, as well as confidence intervals where appropriate. When several methods are tested on the same independent corpus, comparisons should be paired at the corpus level. Treating every document as an independent experimental replicate would be inappropriate when documents are generated from shared templates or grouped within a single corpus. A stronger design would use multiple independent corpora or independently generated vocabularies and treat corpus as the unit of replication. Where the number of corpora is small, the manuscript should present descriptive results and avoid overstated significance claims.

Reproducibility also requires documenting unsuccessful runs and configuration changes. If a transformer model cannot be downloaded, the experiment should be reported as not run rather than silently replaced with SVD while retaining a transformer label. If a metric is undefined because all documents fall into one cluster, the output should preserve that failure as a missing value with an explanation. If the package version changes clustering behaviour, the exact environment should be archived. These details may appear operational, but they determine whether another researcher can verify the reported result.

Conclusion and Future Work

This paper presents a reproducible hybrid phrase-semantic clustering prototype with baselines, ablations, weight sensitivity, controlled stress conditions, exported assignments, cluster phrase labels, metadata, visualisations, and automated tests. The central research premise is that phrase, semantic, and lexical evidence may complement one another, but the usefulness of their combination is an empirical question. The included synthetic experiments provide a controlled demonstration of the pipeline and its limitations. They do not justify a claim that the method is superior on real collections.

The next stage should evaluate the method on public document datasets, include multiple independent random seeds, compare against established embedding-clustering pipelines such as BERTopic where appropriate, and use a held-out evaluation protocol. The semantic mode should be run with a named transformer model and its model revision recorded. The study should also measure peak memory and runtime at increasing corpus sizes, and conduct a human evaluation of cluster labels and search tasks. A more advanced algorithm could estimate channel reliability on a validation set or use uncertainty-aware fusion, but any such mechanism would require a separately specified method and rigorous comparison.

Declarations

Funding: No funding statement supplied; author to complete accurately.
Conflict of interest: none
Data availability: The synthetic data generator and CSV files are included in the companion research package.
Code availability: The Python implementation, requirements, tests, outputs, and execution instructions are included in the companion ZIP.
Ethics: The bundled experiments use synthetic text and do not involve human participants. Any future user study requires appropriate consent and institutional review where applicable.

References

Amiri, B., & Karimianghadim, R. (2024). A novel text clustering model based on topic modelling and social network analysis. *Chaos, Solitons & Fractals, 181*, Article 114633. https://doi.org/10.1016/j.chaos.2024.114633

Carpineto, C., Osiński, S., Romano, G., & Weiss, D. (2009). A survey of Web clustering engines. *ACM Computing Surveys, 41*(3), Article 17. https://doi.org/10.1145/1541880.1541884

Cutting, D. R., Karger, D. R., Pedersen, J. O., & Tukey, J. W. (1992). Scatter/Gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 318–329). ACM. https://doi.org/10.1145/133160.133214

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 41*(6), 391–407. https://doi.org/10.1002/(SICI)1097-4571(199009)41:6%3C391::AID-ASI1%3E3.0.CO;2-9

Grootendorst, M. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv*. https://doi.org/10.48550/arXiv.2203.05794

Hearst, M. A., & Pedersen, J. O. (1996). Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 76–84). ACM. https://doi.org/10.1145/243199.243216

Hoyle, A., Calvo-Bartolomé, L., Boyd-Graber, J., & Resnik, P. (2025). ProxAnn: Use-oriented evaluations of topic models and document clustering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 15872–15897). Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.acl-long.772

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*, 193–218. https://doi.org/10.1007/BF01908075

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press. https://nlp.stanford.edu/IR-book/

McCreight, E. M. (1976). A space-economical suffix tree construction algorithm. *Journal of the ACM, 23*(2), 262–272. https://doi.org/10.1145/321941.321946

Ortakci, Y., & Borhan, B. (2025). Optimizing SBERT for long text clustering: Two novel approaches with empirical insights. *The Journal of Supercomputing, 81*, Article 950. https://doi.org/10.1007/s11227-025-07414-4

Ortu, M. (2025). Topic-sentiment hybrid networks for explainable document clustering: A probabilistic multi-dimensional similarity analysis. *Applied Stochastic Models in Business and Industry, 41*(4), Article e70024. https://doi.org/10.1002/asmb.70024

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (pp. 3982–3992). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1410

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Saeed, S., Rajput, Q., & Haider, S. (2024). SUMEX: A hybrid framework for semantic textual similarity and explanation generation. *Information Processing & Management, 61*(5), Article 103771. https://doi.org/10.1016/j.ipm.2024.103771

Ukkonen, E. (1995). On-line construction of suffix trees. *Algorithmica, 14*, 249–260. https://doi.org/10.1007/BF01206331

Zamir, O., & Etzioni, O. (1998). Web document clustering: A feasibility demonstration. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 46–54). ACM. https://doi.org/10.1145/290941.290956

Zamir, O., & Etzioni, O. (1999). Grouper: A dynamic clustering interface to Web search results. *Computer Networks, 31*(11–16), 1361–1374. https://doi.org/10.1016/S1389-1286(99)00054-7

Appendix A. Reproduction commands

Use the commands below from the root of the extracted project directory. Run the experiments in a clean environment and retain the generated metadata and output files.

Install dependencies

python -m pip install -r requirements.txt

Run the primary benchmark

python src/hybrid_clustering.py --input data/synthetic_documents.csv --outdir results --seed 20261018 --clusters 4

Run automated tests

pytest -q

Rerun without weight sensitivity sweep

python src/hybrid_clustering.py --input data/synthetic_documents.csv --outdir results_fast --seed 20261018 --clusters 4 --no-sweep

Optional transformer mode

python src/hybrid_clustering.py --input data/synthetic_documents.csv --outdir results_transformer --sentence-transformers