Drift Inspector: Exploring and Measuring Scientific Drift with Atomic Contribution Claims

Vsevolod Karimov[1,2], **Stepan Ostarkov**[3,4], **Anastasia Poroshina**[1], **Anatoly Frolov**[1,5], and **Alexander Panchenko**[1,5]
[1]Skoltech, [2]HSE University, [3]Lomonosov Moscow State University, [4]ITMO University, [5]AIRI
Correspondence: {hamyrappy, panchenko.alexander}@gmail.com

Abstract

Scientific abstracts mix contributions with background, motivation, and meta-language, so tools that read them as-is cannot separate what a field *produces* from what it *discusses*. We present *Drift Inspector*, an open-source system for measuring and exploring how a research field changes over time at the level of *Atomic Contribution Claims (ACCs)*: decontextualized, contribution-bearing propositions an LLM extracts from each abstract before analysis. The system clusters these claims across years into an interactive map where every trend traces back to the claims and papers behind it. Applied to six years of EMNLP, it shows the field shifting away from classic NLP tasks toward LLM-era capabilities such as reasoning and multimodality — a movement that keyword or whole-abstract counts blur. The released data extend beyond EMNLP: the same pipeline has processed the full ACL Anthology (346k claims, 80k abstracts, 423 venues). Extraction is human-validated and clustering checked against an external manually constructed taxonomy.[1]

1 Introduction

Major NLP venues, such as ACL or EMNLP, publish thousands of papers released every few months — far beyond than anyone can read. This is due to recent acceleration in AI development, firstly attracting more interest to the field, but also because the research itself, including paper writing, now can be automated with LLMs and agents. As a result, the number of published papers has a trend on sharp increase every year. Similar situation is observed in most large AI-related conferences beyond NLP, such as NeurIPS, ICLR, CVPR, etc.

This urges development of tools for automatic summarization and effective and efficient exploration of this tsunami of research contributions. Besides, to understand how a research field evolves, one needs to know what its papers contribute, not only what they talk about. In rapidly moving fields like NLP the two diverge: an old thread may still saturate the motivation of papers that now advance other topics, while a new direction may appear only in contribution statements.

Existing literature exploration systems do not make this distinction. For instance, NLP Scholar (Mohammad, 2020), NLPExplorer (Parmar et al., 2020), and CL Scholar (Singh et al., 2018) index papers, venues, authors, and term frequencies; topic-model dashboards built on LDA or BERTopic (Grootendorst, 2022) operate on raw abstract text. So in existing systems, usually, the word "parsing" would count the same whether it appears in the motivation of an LLM reasoning paper or in the actual contribution of a dependency parsing paper.

Drift Inspector, system presented in this paper, aim to address this gap. Its pipeline first maps each abstract into a small set of **Atomic Contribution Claims (ACCs)** — self-contained, falsifiable propositions about what the paper contributes, stripped of motivation, prior-work framing, and meta-language. Extraction acts as an analytic lens: it fixes what kind of information enters every downstream statistic. We then embed the claims with SPECTER2, cluster them jointly across years, and aggregate them into per-cluster prevalence trajectories. The result is served as a web application with linked views: an interactive claim map (year filters, drift coloring), cluster trend pages, side-by-side comparison of cohorts — conferences, authors, keyword subsamples (Figure 3); every aggregate traces back to the claims and source papers.

The system is aimed at three audiences: (i) *researchers* positioning new work or writing surveys, who need to see how a topic evolved and what replaced it; (ii) *conference organizers* or area chairs, who need evidence of topical shifts across conference years; and (iii) *science-of-science researchers*,

[1]A live demo, code and data are available at: https://hamyrappy.github.io/drift-inspector.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 138–148
October 24-29, 2026 ©2026 Association for Computational Linguistics

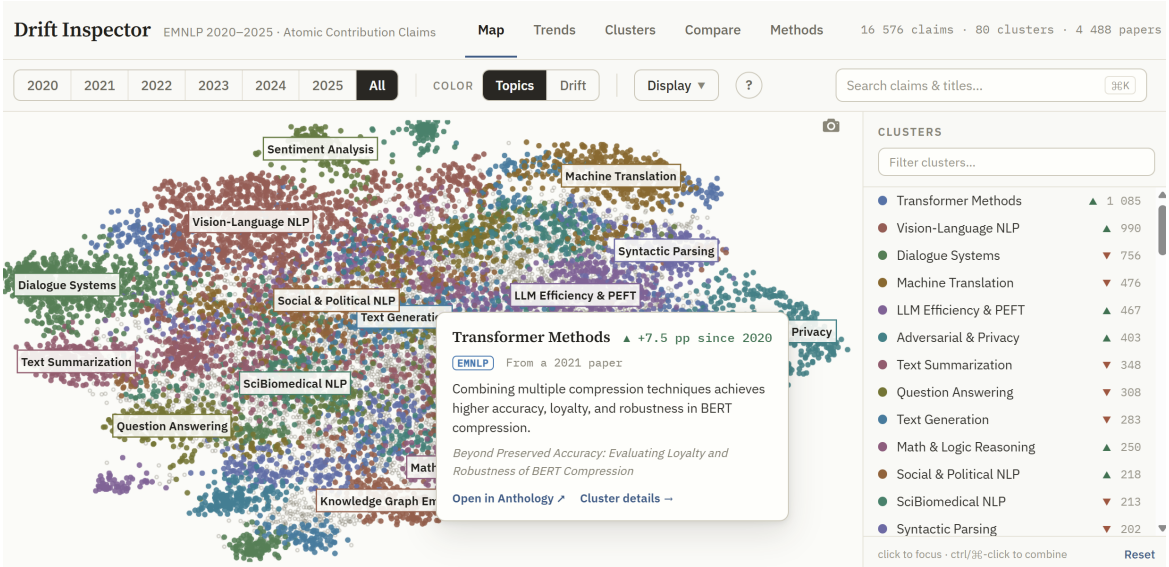

Figure 1: **Drift Inspector frontend** in Map view representing an interactive WebGL map of 16,576 atomic claims from EMNLP 2020–2025. Top bar tabs allow to switch between views (Map, Trends, Clusters, Compare or Methods). A user can adjust year filter, color mode (Cluster or Drift Trend), or perform a claim search with a per-cluster match ranking. On the right are filterable list of clusters. Hovering a point shows its claim and the cluster's prevalence trajectory; clicking pins the card with a link to the source paper.

who get a validated, reusable extraction pipeline.

Our contributions are three-fold:

- A drift-measurement method based on Atomic Contribution Claim (ACC) extraction.
- Pre-extracted results for EMNLP and ACL Anthology corpora. The extracted claims were validated by human annotators and against gold reference topics.
- A visualization and exploration tool of the ACC datasets enabling a drift analysis across years, authors, conferences, etc. (cf. Figure 1).

2 Related Systems and Work

2.1 Literature exploration systems

NLP Scholar (Mohammad, 2020) provides interactive dashboards over ACL Anthology metadata and term frequencies; NLPExplorer (Parmar et al., 2020) and CL Scholar (Singh et al., 2018) index papers, fields, and citation structure. These systems operate at the topic or keyword level. More recent systems include NLP-KG (Schopf and Matthes, 2024), GenGo (Takeshita et al., 2024), and GenGo Ultra (Takeshita et al., 2025). Also various surveys of NLP as a field has been conducted, e.g. (Schopf et al., 2023)

Drift Inspector differs in its unit of analysis, extracting *atomic contribution claims*, distinguishing what papers contribute from what they mention. As a tool it also adds a drift-color overlay that turns the claim map into a field-level trend heatmap, claim-to-paper traceability from every aggregate, side-by-side cohort comparison (venues, authors, keyword-defined subsamples), and a corpus-agnostic single-file deploy that these corpus-tied dashboards do not offer.

2.2 Diachronic analyses of NLP

Prior work tracks task/method prevalence (Uban et al., 2021), idea co-occurrence (Tan et al., 2017), rhetorical framing (Prabhakaran et al., 2016), and causal paradigm shifts (Pramanick et al., 2023). Closest is Pramanick et al. (2025), which classifies contribution statements into a fixed taxonomy. Our system instead lets contribution clusters emerge bottom-up and packages the whole measurement loop — extraction, validation, clustering, statistics, visualization — into a reusable tool.

2.3 Atomic semantic units

Claim-level decomposition follows the atomization principle of FActScore (Min et al., 2023) and Atomic Content Units (Liu et al., 2023), used there for factuality and summarization evaluation. Idea-atom decompositions have also been used to sample novel research directions (Artiles et al., 2026). We repurpose atomization for scientometric measurement. ACCs are units of field-level counting, so we check them against their own abstracts and leave their truth to the source paper.

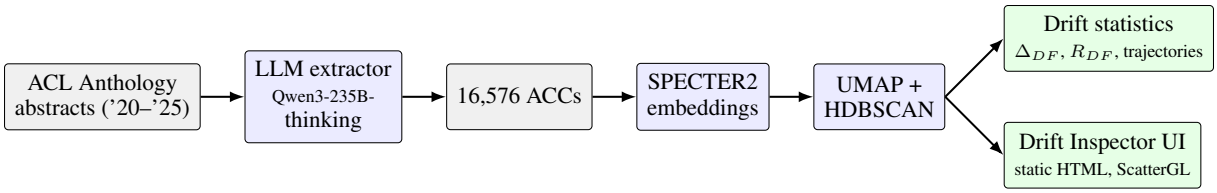

Figure 2: **System workflow.** An LLM maps each abstract to Atomic Contribution Claims; claims are embedded with SPECTER2 and clustered jointly across years; per-cluster document-frequency trajectories and the interactive map are generated from the same canonical clustering.

3 System Description

Figure 2 shows an overall workflow of Drift Inspector featuring offline part (extraction, embedding, clustering, drift statistics) and the online part, which was illustrated earlier in detail in Figure 1.

3.1 Atomic Contribution Claims

Raw abstracts conflate multiple contributions, background context, and meta-language ("In this paper, we propose..."). Given a document d, an LLM agent maps its abstract into a set of ACCs $A_d = \{a_1, \ldots, a_n\}$, each satisfying three constraints: **atomicity** (exactly one contribution-bearing proposition), **decontextualization** (pronouns resolved to named entities, meta-language removed), and **falsifiability** (a verifiable assertion). A typical ACC looks like "*TheoremLlama uses curriculum learning and block training techniques to train large language models for formal theorem proving*" (Wang et al., 2024). We deliberately operate on abstracts: they are the author-curated contribution summary, are uniformly available across all six years, and keep the pipeline to one cheap LLM pass per paper. The extractor is `qwen3-235b-a22b-thinking-2507` (Team, 2025) (via OpenRouter, temperature 0.2) with a few-shot prompt that excludes background, motivation, and raw metric claims; the condensed prompt is in Appendix A and the full prompt in the released code. On EMNLP 2020–2025 the extractor yields 3.58 claims per abstract in 2020 and 4.01 in 2025; abstracts with no extractable contribution (3 of 751 in 2020) are excluded.

3.2 Semantic Topology

We embed all claims with SPECTER2 (Cohan et al., 2020; Singh et al., 2023), a scientific encoder pretrained on citation relatedness, so claims about similar mechanisms land close together. We then reduce the vectors with UMAP (McInnes et al., 2018) and cluster them with HDBSCAN (McInnes et al., 2017), as in BERTopic (Grootendorst, 2022), except that our unit is the claim rather than the abstract. Each cluster gets class-based TF-IDF descriptors — short (top-3, e.g., *agents, action, web*) and extended (top-5) — and all 80 clusters additionally carry LLM-generated, author-reviewed readable names: a short name for map labels (e.g., *Math & Logic Reasoning*) and a full name on the cluster page (e.g., *Mathematical and Logical Reasoning in LLMs*), with the c-TF-IDF descriptor preserved in the claim card and cluster page.

About 36% of claims remain unclustered "noise" — paper-specific contributions that have not (yet) become recurring field-level patterns; the share rises from 31–34% in 2020–2022 to 42% in 2025, consistent with the newest contributions having had the least time to consolidate. These are kept in the interface as a background layer rather than discarded. Retaining them is why the pipeline clusters by density (HDBSCAN): a method that must place every claim would put those with no field-level counterpart into the nearest cluster and dilute it. All hyperparameters are listed in Appendix B.

3.3 Drift Quantification

We measure prevalence as paper-level **document frequency**: the fraction of papers in year y with at least one claim from cluster c,

$$P_y(c) = |\{d \in D_y \mid A_d \cap c \neq \emptyset\}| \,/\, |D_y|$$

where D_y is the set of papers in year y and A_d the claims extracted from paper d. This prevents prolific claim lists from inflating a topic: a paper with ten RAG claims counts once. Because $P_y(c)$ is computed independently per cluster, the unclustered noise does not renormalize the other clusters' shares. The system reports the absolute endpoint shift $\Delta_{DF}(c) = P_{2025}(c) - P_{2020}(c)$, the relative change $R_{DF}(c)$, full per-year trajectories, and a relative *drift* score — the base-2 log-ratio of the 2025 to 2020 share — that drives the map's drift-color overlay (saturating at an $8\times$ change).

3.4 The Drift Inspector Frontend

The frontend (Figure 1) is a web application with five linked views. Every number it shows opens down to the claims and papers behind it.

- **Map.** A WebGL scatter (Plotly ScatterGL) of all 16,576 claims over a shared 2D UMAP projection. Year-filter buttons show any single year (2020–2025) or all at once; a color switch recolors every cluster by its prevalence trend (red = declining, green = growing), turning the map into a field-level drift heatmap; a Display menu toggles cluster names, the noise layer, and auto-fit. A search box highlights all claims whose text or source title contains a query substring (e.g., `retrieval`) and ranks the matching clusters. Hovering a point shows the claim, source paper, year, cluster name, and prevalence trajectory; clicking pins this card with a link to the ACL Anthology. The legend isolates clusters (ctrl/cmd-click to combine).
- **Trends.** A butterfly chart of the 2020→2025 document-frequency shift, per-year trajectories for any clusters the user selects (by clicking bars or sparklines), and a sparkline overview of all clusters.
- **Clusters.** A per-cluster profile: its document frequency by year, its location in claim space, and all of its claims with links to source papers, exportable as CSV.
- **Compare.** Side-by-side thematic profiles of two user-chosen cohorts (a conference, an author, or a keyword-defined subsample), ranking clusters by their gap in paper share (Figure 3), with a paper/claim-share toggle, per-cohort year ranges, and an *over time* mode (Figure 5).
- **Methods.** A self-contained account of the pipeline, corpus, and views, generated from the corpus metadata so the build re-labels itself for a new dataset.

The noise layer renders as a faint background on the Map and can be inspected like any other point. A light/dark theme toggle and the Display options reconfigure the view for figure capture.

3.5 Implementation and Availability

The pipeline is Python (transformers, adapters, umap-learn, hdbscan, Plotly); every stage caches its artifact (claims CSV, embedding matrix, projection, cluster assignments), and all stochastic steps are seeded, so from the released claims the published clustering is exactly reproducible from configuration. Re-running the extraction reproduces aggregate rates but not individual claim strings, because LLM decoding is not reproducible run to run. Anybody can rerun the whole process end to end for a small API cost and negligible compute (§3.6). The frontend has no dependencies beyond a browser and ships as a self-contained file in our GitHub repository. Code is released under the MIT license; the claim dataset under CC BY 4.0 (derived from openly licensed ACL Anthology abstracts, CC BY 4.0). Appendix B gives the extractor and its decoding parameters, the embedding checkpoint, the clustering hyperparameters and the random seeds.

The same pipeline runs unchanged at corpus scale. On a research inference cluster we extracted the full ACL Anthology — 80,144 abstracts, 423 venues, 346,010 claims — with the cheaper `gpt-oss-120b` extractor (∼$40 at median commercial API prices; §3.6), and we run a second live Drift Inspector over six *ACL main conferences (70k claims, 2018–2026).[2] There, cross-venue and cross-author questions become one-click cohort comparisons: Figure 3 contrasts ACL and COLING directly. On the EMNLP corpus this extractor swap preserves cluster validity and all 15 headline drift directions (§4.3). The human-aligned judge (§4.1) rates 300-claim samples from the scaled corpora at 96.7–99.7% Good (full EMNLP main track 98.7%, full ACL 99.7%, whole anthology 96.7%) against a 98.3% same-judge control on the validated corpus — overlapping 95% intervals throughout — and per-corpus judge reports ship with the data. All analyses in this paper are computed on the human-validated EMNLP corpus.

3.6 Scalability and Cost

The pipeline makes one LLM call per abstract, so extraction dominates the cost and grows linearly with the corpus (Table 1). Budgeting a new corpus is one multiplication: about **$0.50 per thousand abstracts** with `gpt-oss-120b`, so the full ACL Anthology — 80,144 abstracts — costs ∼$40 and runs overnight. A reasoning extractor such as Qwen3-235B-Thinking costs ∼$11 per thousand instead, because most of what you pay for is hidden reasoning, not the claims it returns. The cheap extractor preserves every headline drift direction (§4.3), so you do not need to pay it. Everything after extraction is two orders of magnitude cheaper and needs

[2] https://hamyrappy.github.io/drift-inspector-acl/

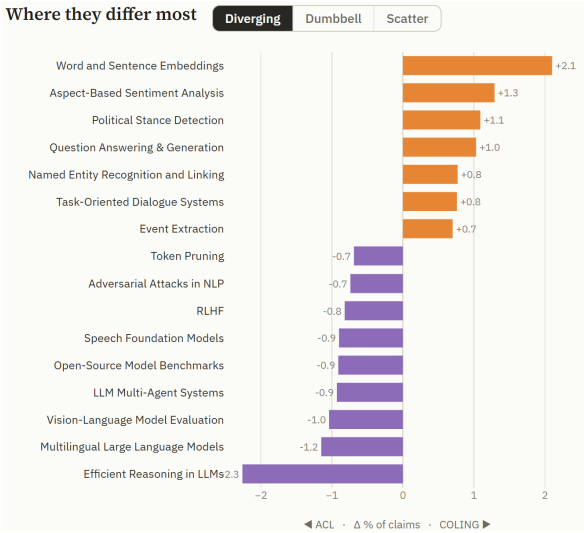

Figure 3: **Compare view, venue cohorts** (live six-venue instance): where ACL and COLING differ most, as the gap in cluster claim shares. COLING keeps a markedly stronger classic-NLP profile (embeddings, aspect-based sentiment, NER, event extraction), while ACL leans toward LLM-era clusters (efficient reasoning, multilingual LLMs, vision–language evaluation). Chips switch the diverging / dumbbell / scatter renderings; the same view compares authors and keyword-defined cohorts.

no accelerator: embedding, projection and clustering finish in well under an hour of laptop CPU at either scale, and the frontend needs no server.

4 Evaluation

We evaluate the two stages that could corrupt the analysis: claim extraction quality and cluster validity. Both use the released EMNLP 2020–2025 ACC corpus: extraction quality on a stratified sample of extracted claims (with curated negative controls), and cluster validity on the 653 papers where our corpus overlaps the SToP gold taxonomy (Rohatgi et al., 2023). Robustness of the headline findings is evaluated separately (§4.3).

4.1 Human Validation of Claim Quality

Three of the authors labeled 180 items (136 claims sampled from the extraction output, stratified by year, plus 44 curated invalid claims, which the annotators did not know about, as negative controls) as Good, Bad, or Unsure; a claim is Good if it is atomic, self-contained, contribution-bearing, and faithful to the abstract. Inter-annotator agreement is high and robust to the Unsure convention (Fleiss' $\kappa = 0.844$ dropping Unsure, 0.76 folding it into Bad, 0.73 across three categories). On the 136 sampled claims the annotators rate 97.8%, 91.8%, and 90.5% as Good, and flag 44/44 planted negatives by 2-of-3 majority, so the labels are discriminating, not lenient. An independent LLM judge from a different vendor (Gemini vs. Qwen), separately prompted and blind to the human labels, matches the majority consensus with 94.5% accuracy ($\kappa = 0.863$) and flags 43 of 44 negatives, supporting its use for monitoring extraction on new corpora. The dominant failure modes are unsupported details and background-context extraction; the full protocol is released with the data.

	EMNLP '20–'25	ACL Anthology
Abstracts	4,960	80,144
Claims	18,293	346,010
Extractor	Qwen3-235B-T	gpt-oss-120b
Extraction cost	$54	∼$40
per 1k abstracts	$10.87	$0.51
Extraction wall-clock	1.9 h	6.8 h
Embed + cluster*	8 min	34 min
Frontend, one file*	8 MB	18.9 MB

Table 1: Cost of the two released extractions. The EMNLP figure is the amount actually invoiced; the anthology pass ran on donated compute, so its cost is the equivalent at 2026 commercial rates. *measured on the deployed instance: the 16,576-claim EMNLP corpus and the 70k-claim six-venue instance built from the anthology extraction.

4.2 Cluster Validity Against an External Taxonomy

We benchmark the clustering against the SToP gold taxonomy (Rohatgi et al., 2023) on the 2020–2021 overlap (653 papers), comparing eight configurations: bag-of-words topic models (LDA, NMF), abstract-level neural clustering (BERTopic with SPECTER2 and MPNet encoders), sentence-level clustering (SentSPECTER), and claim-level clustering (ACC, ours), each optionally restricted to its 30 largest clusters. Table 2 summarizes the SToP-alignment results.

Coverage differs by construction: LDA/NMF/BERTopic are scored with a soft document–topic distribution (BERTopic via `approximate_distribution`) that assigns mass even to abstracts HDBSCAN treats as outliers (∼100% coverage), whereas ACC keeps a hard HDBSCAN partition with noise excluded, leaving all-noise papers uncovered (83.9%). The coverage-robust V-measure, LRAP, nDCG@1, and top-30 rows neutralize this, and ACC leads or ties throughout. The soft step was required for BERTopic, because under equally hard assignment

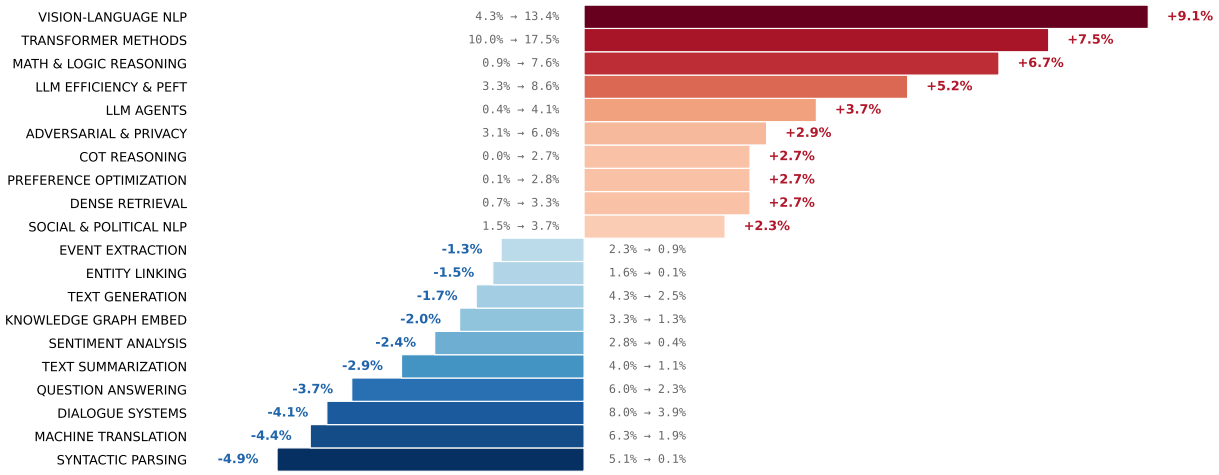

Figure 4: **Scientific drift in EMNLP (2020–2025)** as measured by the system. Diverging bars show the change in paper-level document frequency (percentage points) for the top emerging and declining ACC clusters, annotated with 2020 and 2025 shares.

Method	Pur.	LRAP	nDCG@1	V-mes	PairF1
LDA	0.245	0.508	0.329	0.278	0.170
NMF	0.294	0.585	0.445	0.324	0.186
$\text{BERTopic}_{\text{SPECTER2}}$	0.324	0.651	0.515	0.469	0.289
$\text{BERTopic}_{\text{MPNet}}$	0.327	0.651	0.509	0.457	0.234
SentSPECTER	0.654	0.668	0.540	0.532	0.268
$\text{SentSPECTER}_{\text{top30}}$	0.472	0.637	0.512	0.471	0.365
ACC *(ours)*	**0.689**	0.674	0.580	**0.571**	0.361
$\text{ACC}_{\text{top30}}$	0.618	**0.694**	**0.596**	0.552	**0.403**

Table 2: Alignment with the SToP taxonomy (2020–2021 subset, 653 papers). Purity and V-measure are computed on covered documents; LRAP and nDCG@1 via 5×5 repeated cross-validation, the topic-to-label map learned per fold on the training split. The step from abstract-level ($\text{BERTopic}_{\text{SPECTER2}}$, 0.324) to sentence-level (0.654) to claim-level (0.689) purity shows input granularity matters more than encoder choice. Full table with intrinsic metrics and coverage in the released artifacts.

it leaves 22–26% of abstracts as outliers vs. 16% for ACC. The hard partition is therefore not what costs coverage: the noise fraction follows from the input granularity. A soft ACC assignment remains a natural extension.

4.3 Robustness of Headline Findings

Our headline finding is the set of 15 cluster-level prevalence shifts in Figure 4. Six reasons explain the pattern without the field having changed: sampling noise, a two-snapshot artifact, the encoder, the cluster-matching rule, the extractor, and the clustering hyperparameters. Each test below tracks the same quantity, the sign of the 2020→2025 shift.

Sampling. All 15 shifts are significant under a two-proportion z-test (13 at $p < 0.001$), and their bootstrap 95% CIs on Δ_{DF} (2,000 paper-level resamples) exclude zero.

Two snapshots. Trajectories are largely monotone across the six years (parsing runs $5.1 \rightarrow 3.2 \rightarrow 1.6 \rightarrow 1.6 \rightarrow 0.8 \rightarrow 0.1\%$ of papers; Figure 6), so the shifts are not an artifact of comparing the two endpoints.

Encoder. We re-embed and re-cluster the same claims with nine alternative representations under an identical UMAP/HDBSCAN pipeline: eight other sentence and document encoders (SciNCL, SPECTER1, MPNet, MiniLM, BGE-large/small, E5-large, Nomic) and a non-neural TF-IDF + SVD baseline. Cluster counts and noise fractions stay in the same range (55–116 clusters, 31–53% noise), and every representation preserves the sign for at least 11 of the 15 headline clusters (median 14/15; TF-IDF + SVD 12/15); a cluster in an alternative clustering is identified with a headline cluster by best-Jaccard overlap of their claim sets. Per-encoder results are in Appendix C.

Matching rule. Recomputing all signs under two sign-blind alternatives to best-Jaccard matching, union overlap and weighted claim image, never lowers a per-encoder count.

Extractor. Re-extracting with `gpt-oss-120b`, a different model family that yields 4.6 claims per paper against 3.7, preserves all 15 signs and keeps claim-level clustering ahead of every abstract-level baseline on SToP (purity 0.638, V-measure 0.541).

Clustering hyperparameters. A 36-cell sweep (UMAP *n_neighbors* $\in \{30, 40, 50\}$, *n_components* $\in \{5, 10\}$; HDBSCAN *min_cluster_size* $\in \{20, 25, 30\}$, *min_samples* $\in \{5, 10\}$) yields clusterings that agree with the reported one at ARI 0.79–0.95 (AMI 0.93–0.97), with between 55 and 116 clusters. The sign is preserved for all 15 headline clusters in all 36 cells.

5 Case Study: EMNLP 2020–2025

Running the system on 4,488 EMNLP papers (748 per year across 2020–2025) yields the field-level picture in Figure 4. Classic task clusters lose document-frequency share: parsing falls most steeply (-4.9 pp, -97% relative), while machine translation, dialogue, and question answering each shrink by 3.7–4.4 pp. LLM-era clusters grow in their place: multimodal and vision-language modeling gains $+9.1$ pp ($3.1\times$ its 2020 prevalence) and agents rise more than tenfold ($10.3\times$), with mathematical reasoning, large-model tuning and inference, and a newly emergent LLM reasoning-behavior cluster making up the rest of the growth. Qualitatively, the declining tasks have not vanished: claim search shows their residual claims thinning and dispersing into the noise layer and neighbouring clusters rather than re-forming as standalone contributions — fragmentation that paper- or keyword-level counts cannot distinguish from disappearance. Figure 4 annotates all twenty clusters with their 2020 and 2025 shares.

5.1 Use Cases

Is this area still growing? A researcher weighing a retrieval project wants to know whether the area is expanding before committing to it. Typing `retrieval` into the Map search highlights every claim whose text or source title matches, and ranks clusters by how many of those matches they hold. The top-ranked cluster's trend page shows its share of papers rising from 0.7% in 2020 to 3.3% in 2025, and the ranking places that growth in dense-retrieval and LLM-adjacent clusters rather than in classic IR. Each point on the trajectory opens the claim behind it and links to its paper.

What does a venue publish? Compare takes two cohorts and ranks clusters by their gap in paper share. Figure 3 contrasts ACL with COLING: COLING keeps a stronger classic-NLP profile (embeddings, aspect-based sentiment, NER, event extraction), while ACL leans toward efficient reasoning, multilingual LLMs and vision–language evaluation. A cohort can also be an author or a keyword-defined subsample, so `bert` against `llm` (Figure 5) or two research groups. The *over time* mode makes the comparison diachronic.

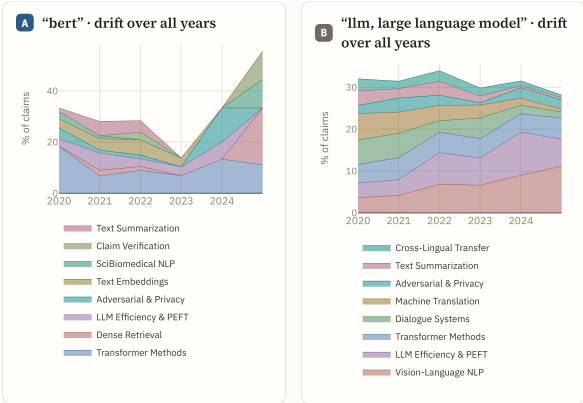

Figure 5: **Compare view, *over time* mode:** yearly topic mix (top clusters, % of cohort claims) for the `bert` cohort (197 papers) vs. `llm / large language model` (3,253 papers): the residual `bert` agenda concentrates in claim verification and biomedical NLP, while the `llm` cohort shifts toward vision–language modeling and efficiency.

What is emerging, and what is declining? Trends ranks every cluster by its 2020→2025 shift with bootstrap intervals, so both ends appear in one view: LLM reasoning behaviour and vision–language modeling at the growing end, syntactic parsing (-4.9 pp, -97% relative) at the other. Because prevalence counts papers with at least one claim in a cluster, a topic that is still discussed but no longer contributed looks different from one that has disappeared: claim search shows the residual parsing claims scattered across the noise layer and neighbouring clusters instead of forming a cluster of their own. A narrower term behaves the same way. Searching `humor` puts almost all of its matches in the noise layer, each still linked to its source paper.

6 Conclusion

Drift Inspector turns a validated claim-extraction pipeline into an interactive instrument for watching a field reorganize itself, its trends reflecting what papers assert they add rather than the vocabulary of their motivation. The system is fully open (MIT code, CC BY data) and corpus-agnostic: any venue with abstracts can be mapped in one pass. Next step is implementing a hierarchical theme system aggregating clusters into navigable super-themes.

Limitations

The analysis relies on abstracts, which omit technical nuance and negative results, and some contributions surface only in the paper body. All validated results are on a single venue (EMNLP): the larger extractions we release (six *ACL venues; the full anthology) are only LLM-judge-monitored, not human-revalidated, so their per-cluster numbers warrant more caution. The pipeline rests on an LLM extractor that can introduce errors (hallucination, under/over-splitting), mitigated but not eliminated by schema and human validation and an aligned LLM judge; human validation was by the authors on 136 claims, so agreement partly reflects shared training, and the negatives are curated corruptions rather than sampled natural errors. Clustering depends on UMAP/HDBSCAN hyperparameters and the embedding model; headline drift directions are stable under bootstrap resampling, eight encoders, and a lexical representation, but cluster *boundaries* and labels may vary, and drift statistics are descriptive, not causal.

Ethics Statement

The system analyzes publicly available ACL Anthology abstracts (CC BY 4.0) and involves no private data. The main risk is misinterpretation: cluster prevalence reflects sampling, extraction, and clustering choices and should be read as a descriptive signal about abstracts, not a verdict on subfields' scientific value: a decline in standalone parsing contributions does not mean parsing is solved or worthless. We document all parameters and release all artifacts to keep the measurement inspectable.

Acknowledgements

The work was supported by the grant for research centers in the field of AI provided by the Ministry of Economic Development of the R.F. in accordance with the agreement 000000C313925P4F0002 and the agreement with Skoltech №139-10-2025-033.

References

Alejandro H. Artiles, Martin Weiss, Levin Brinkmann, Anirudh Goyal, and Nasim Rahaman. 2026. Alien science: Sampling coherent but cognitively unavailable research directions from idea atoms. *CoRR*, abs/2603.01092.

Arman Cohan, Sergey Feldman, Iz Beltagy, and 1 others. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based TF-IDF procedure. *CoRR*, abs/2203.05794.

Yixin Liu, Alexander R. Fabbri, Pengfei Liu, and 1 others. 2023. Revisiting the gold standard: Grounding summarization evaluation with robust human evaluation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4140–4170, Toronto, Canada. Association for Computational Linguistics.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. UMAP: uniform manifold approximation and projection. *J. Open Source Softw.*, 3(29):861.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, and 1 others. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Saif M. Mohammad. 2020. NLP scholar: An interactive visual explorer for natural language processing literature. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 232–255, Online. Association for Computational Linguistics.

Monarch Parmar, Naman Jain, Pranjali Jain, P. Jayakrishna Sahit, Soham Pachpande, Shruti Singh, and Mayank Singh. 2020. NLPExplorer: Exploring the universe of NLP papers. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II*, volume 12036 of *Lecture Notes in Computer Science*, pages 476–480. Springer.

Vinodkumar Prabhakaran, William L. Hamilton, Dan McFarland, and Dan Jurafsky. 2016. Predicting the rise and fall of scientific topics from trends in their rhetorical framing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1170–1180, Berlin, Germany. Association for Computational Linguistics.

Aniket Pramanick, Yufang Hou, Saif Mohammad, and Iryna Gurevych. 2023. A diachronic analysis of paradigm shifts in NLP research: When, how, and why? In *Proceedings of the 2023 Conference on*

Empirical Methods in Natural Language Processing, pages 2312–2326, Singapore. Association for Computational Linguistics.

Aniket Pramanick, Yufang Hou, Saif M. Mohammad, and Iryna Gurevych. 2025. The nature of NLP: Analyzing contributions in NLP papers. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25169–25191, Vienna, Austria. Association for Computational Linguistics.

Shaurya Rohatgi, Yanxia Qin, Benjamin Aw, and 1 others. 2023. The ACL OCL corpus: Advancing open science in computational linguistics. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10348–10361, Singapore. Association for Computational Linguistics.

Tim Schopf, Karim Arabi, and Florian Matthes. 2023. Exploring the landscape of natural language processing research. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1034–1045, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Tim Schopf and Florian Matthes. 2024. NLP-KG: A system for exploratory search of scientific literature in natural language processing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 127–135, Bangkok, Thailand. Association for Computational Linguistics.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. 2023. SciRepEval: A multi-format benchmark for scientific document representations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5548–5566, Singapore. Association for Computational Linguistics.

Mayank Singh, Pradeep Dogga, Sohan Patro, and 1 others. 2018. CL scholar: The ACL anthology knowledge graph miner. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 16–20, New Orleans, Louisiana. Association for Computational Linguistics.

Sotaro Takeshita, Simone Ponzetto, and Kai Eckert. 2024. GenGO: ACL paper explorer with semantic features. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 117–126, Bangkok, Thailand. Association for Computational Linguistics.

Sotaro Takeshita, Tornike Tsereteli, and Simone Paolo Ponzetto. 2025. GenGO ultra: an LLM-powered ACL paper explorer. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 242–251, Vienna, Austria. Association for Computational Linguistics.

Chenhao Tan, Dallas Card, and Noah A. Smith. 2017. Friendships, rivalries, and trysts: Characterizing relations between ideas in texts. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 773–783, Vancouver, Canada. Association for Computational Linguistics.

Qwen Team. 2025. Qwen3 technical report. *CoRR*, abs/2505.09388.

Ana Sabina Uban, Cornelia Caragea, and Liviu P. Dinu. 2021. Studying the evolution of scientific topics and their relationships. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1908–1922, Online. Association for Computational Linguistics.

Ruida Wang, Jipeng Zhang, Yizhen Jia, Rui Pan, Shizhe Diao, Renjie Pi, and Tong Zhang. 2024. Theorem-Llama: Transforming general-purpose LLMs into lean4 experts. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11953–11974, Miami, Florida, USA. Association for Computational Linguistics.

A Extraction and Judge Prompts (Condensed)

The extractor returns JSON `{"claims": [{"text": ...}]}`. Core instruction (few-shot examples omitted; full prompts in the repository):

> **Task.** Extract Atomic Contribution Claims (ACCs) from NLP paper titles and abstracts — faithful, self-contained, atomic, contribution-bearing propositions about what the paper introduces, proposes, evaluates, demonstrates, or establishes.
>
> **Include:** new methods, models, objectives, training/inference procedures; new datasets, benchmarks, resources, tools, evaluation setups; empirical findings and analyses established by the paper. **Exclude:** background, motivation, related work, future work, release logistics; vague problem statements; prior-work claims; raw metric-only claims.
>
> **Rules:** one proposition per claim; no facts absent from the title/abstract; resolve pronouns and vague references; no meta-language ("this paper", "we"); mid-level granularity; no raw numbers; no near-duplicates; if unsure, omit; usually 1–5 claims, empty list if none.
>
> **Self-validation.** Silently check every candidate against: contribution-bearing, atomic, faithful, decontextualized; discard failures.

The independent LLM judge receives one candidate claim with its title and abstract and labels it Good/Bad/Unsure with a main-issue category (`background_context`, `unsupported`, `too_vague`, `not_self_contained`, `not_contribution`, `overgeneralized`, `mixed_claims`); it is prompted separately from the extractor and never sees human labels.

Human-validation sampling. We stratify the 180-item sample by year (seed 42; 54/18/18/18/18/54 across 2020–2025). Of these, 136 are claims from the 16,576-claim output, at most one per paper, and 44 are negative controls: invalid claims we built by injecting an unsupported detail, recasting motivation as a contribution, or over-generalizing. Annotators see only the title, abstract, and claim, and never learn which items are controls.

B Pipeline Configuration

Corpus. EMNLP main-track 2020–2025 from the ACL Anthology; each year balanced to 748 non-empty papers (the smallest usable year, 2020, has 751 papers of which 3 have no extractable contribution), seed 42, 4,488 papers and 16,576 claims in the analysis corpus. We do not balance the scale corpora by year: the six-venue instance keeps every main-track paper of its six venues (69,950 claims, 88 clusters, 2018–2026).

Extractor. `qwen3-235b-a22b-thinking-2507` via OpenRouter, temperature 0.2, top-p 0.9, JSON-object mode, no *max_tokens* cap; unparsable outputs dropped. It covered 4,960 abstracts (4,488 after year-balancing) for 42.1M billed tokens, 82% of them reasoning the caller never sees. The anthology used `gpt-oss-120b`, *reasoning_effort* medium, strict JSON schema, *max_tokens* 8000, temperature 0.

Embedding. `allenai/specter2_aug2023refresh_base` with the proximity adapter; claim text only, CLS pooling, max length 512.

Clustering. UMAP: *n_neighbors*=40, *n_components*=5 (clustering) and 2 (visualization), *metric*=cosine, seed 42. HDBSCAN: *min_cluster_size*=25, *min_samples*=5, *eom* selection; noise label -1. Descriptors: class-based TF-IDF (Grootendorst, 2022) over aggregated cluster claims with hyphen-aware (1,2)-gram tokenisation, a claim-frequency floor ($\geq$10 claims), a per-cluster coverage filter ($\geq$5%), and a Snowball-stemmer de-duplicator collapsing morphological/spelling variants; short (top-3) and extended (top-5) descriptors. All 80 clusters carry LLM-generated, author-reviewed short/full names; tables and figures use the short name, falling back to the descriptor.

Frontend. Year, color, search, and theme changes re-render through `Plotly.react`; search and click-through run client-side over per-point metadata. The portable single-file build embeds css, js, Plotly, and data ($\sim$8 MB for EMNLP, 18.9 MB at six-venue scale), with web fonts the only external request. Figure 7 shows Compare's bump-chart mode. effig:bump shows Compare's bump-chart mode.

C Encoder Ablation

Table 3 re-embeds and re-clusters the 16,576 claims with nine alternative representations under the identical UMAP/HDBSCAN pipeline (§4.3). ARI is measured against the canonical SPECTER2 clustering and SToP purity on the 653-paper overlap of Table 2; signs are matched by best Jaccard, which is sign-blind by construction. Recomputing all 9×15 signs under two alternative criteria never lowers a per-encoder count. Every representation preserves

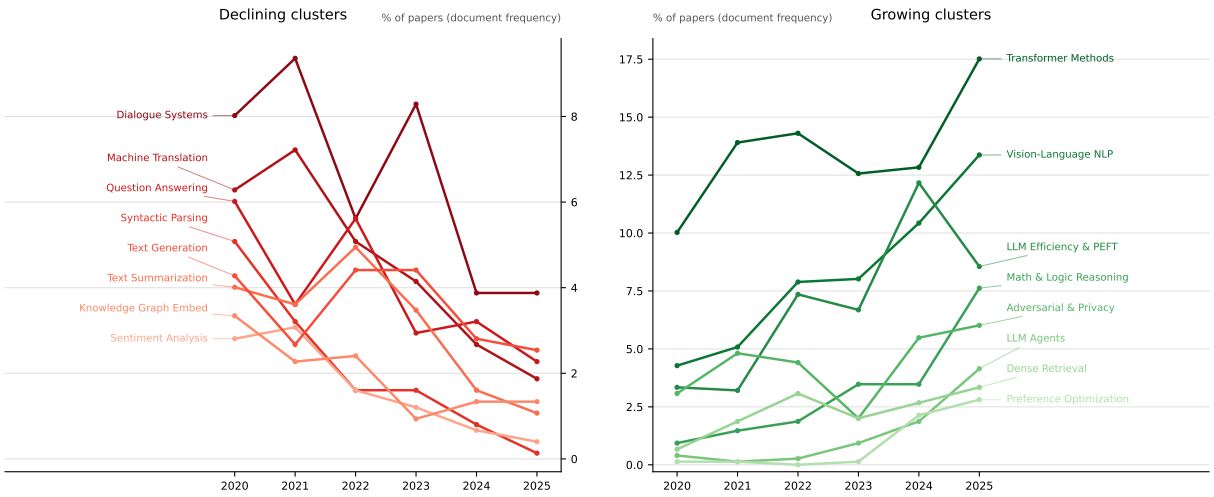

Figure 6: Per-year document-frequency trajectories for the eight most declining and eight most growing ACC clusters (paper-level DF, % of papers), computed from the canonical clustering. Labels are curated short cluster names; full names and c-TF-IDF descriptors ship with the released data. Headline endpoint shifts are significant with bootstrap CIs excluding zero (§4.3).

Encoder	k	Noise%	ARI	SToP	Sign
SPECTER2 *(ours)*	80	36.1	1.00	**0.689**	15/15
SciNCL	111	31.2	0.75	0.670	15/15
SPECTER1	85	34.1	0.69	0.658	14/15
MPNet	116	32.6	0.61	0.681	15/15
MiniLM	107	37.1	0.64	0.646	14/15
BGE-large	99	37.9	0.67	0.674	13/15
BGE-small	92	39.0	0.64	0.657	14/15
Nomic	108	40.6	0.50	0.670	14/15
E5-large	55	52.7	0.37	0.577	11/15
TF-IDF + SVD	—	—	—	—	12/15

Table 3: Encoder ablation on the 16,576 claims. k: clusters, noise excluded. **Sign**: headline shift signs preserved of 15. TF-IDF + SVD is a non-neural control, scored on sign preservation only.

at least 11/15 signs, so the drift is not an artifact of the encoder, and SPECTER2's top purity is why we make it canonical.

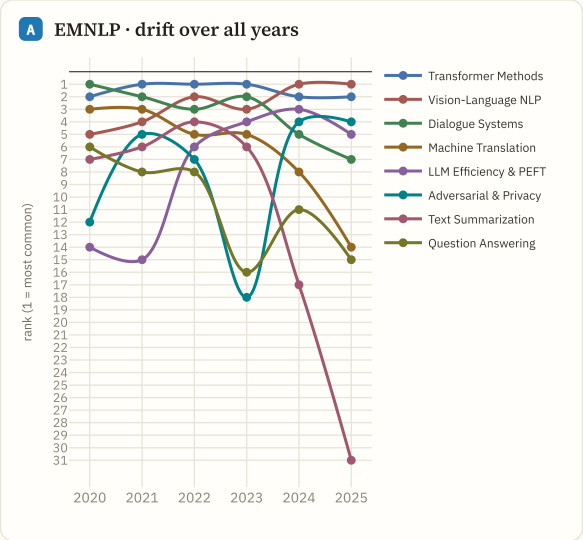

Figure 7: Single-conference diachronic view (Compare, *over time*, bump chart): EMNLP's top clusters ranked by yearly document frequency (1 = most common). Vision–language modeling reaches rank 1 by 2024; LLM efficiency climbs from 14 to 3–5; summarization falls from 7 to 31.