UNIVERSITÉ DU QUÉBEC EN OUTAOUAIS

DÉPARTEMENT DES SCIENCES ADMINISTRATIVES

WORKING PAPER NO. 0001

Research funded by the FIRC (Fonds institutionnel de recherche et de création), Université du Québec en Outaouais.

From Corporate Disclosures to a Russell 3000 Strategic Project Database

Identification, Measurement, Validation, and Research Applications

Manel Kammoun Djerrycharli Tandjambianda Simon-Pierre Boucher

Département des sciences administratives
Université du Québec en Outaouais

manel.kammoun@uqo.ca · djerrycharli.tandjambianda@uqo.ca · simon-pierre.boucher@uqo.ca

Gatineau – Pavillon Alexandre-Taché
283, boulevard Alexandre-Taché
Gatineau, Québec, Canada J9A 1L8

October 28, 2026

Version 1.0 (working draft under validation)

Working draft. Author order, affiliations and CRediT contributions remain subject to confirmation by all coauthors (see `docs/contributions.md`). Not for citation without permission.

Abstract

We document the construction of SPID, a project-level research database that reconstructs the strategic initiatives disclosed by U.S.-listed companies in their SEC filings. Starting from an issuer universe that merges a current S&P 500 snapshot with the holdings of the iShares Russell 3000 ETF (2,564 distinct SEC registrants), the pipeline collects every 10-K, 10-Q and 8-K filed since January 2010 (527,548 documents), segments them into 1.34 million sections, and submits 1.12 million eligible sections to a schema-constrained large-language-model extractor. The result is 308,065 evidence-linked project mentions, resolved into 121,099 candidate projects for 2,492 issuers, each tied to its SEC accession number, section and source passage, together with a controlled 21-category taxonomy, status, cited years, locations, technologies, partners and disclosed amounts. We describe the universe definition and its reconciliation, the distributed acquisition and parsing architecture, the extraction contract and its confidence semantics, the mention-to-project resolution rule and its known failure modes, and the point-in-time structure of the timelines. We then set out a validation framework—stratified human annotation, precision/recall on an independent gold standard, linkage audits and coverage gates—and state explicitly which of its components are implemented and which remain to be executed. Descriptive patterns are reported for the full corpus, with the caveat that a disclosed project is neither a verified investment nor a realized expenditure. The database should be read as a research instrument under validation rather than as an exhaustive census of Russell 3000 projects.

Keywords: SEC filings, text as data, corporate investment, strategic projects, information extraction, large language models, entity resolution, Russell 3000, measurement validation

JEL Classification: C55, C81, G31, G38, M41, O33

1. Introduction

A large share of what firms do strategically—build plants, open data centers, integrate acquisitions, launch products, move systems to the cloud, commit to decarbonization targets—is described, often repeatedly and over several years, in the narrative sections of their regulatory filings. Yet the standard quantitative record of corporate investment is an aggregate: capital expenditure, R&D expense, acquisitions, or counts of patents and plants. Researchers who want to observe the *unit of decision*—a specific, identifiable initiative with a start, a scope, a budget and a fate—have had to assemble hand-collected samples, rely on commercial databases with opaque coverage, or infer initiatives indirectly from word frequencies. This paper documents an attempt to build that unit systematically, from primary sources, for a broad cross-section of U.S.-listed companies.

The object we describe is SPID (*SEC Project Intelligence Database*), developed at the Université du Québec en Outaouais on top of the UQO EDGAR research corpus. Its first release, in June 2026, covered the 500 constituents of the S&P 500 (Tandjambianda et al., 2026). The present paper documents its extension, carried out on October 27–28, 2026, to an issuer universe that approximates the Russell 3000: 2,564 distinct SEC registrants, 527,548 annual, quarterly and current reports filed between January 2010 and October 2026, 1.12 million filing sections submitted to a schema-constrained large-language-model (LLM) extractor, and 308,065 resulting project mentions resolved into 121,099 candidate projects. Every retained mention carries the SEC accession number, the section identifier and a model-generated description grounded in the source passage, so that any number in the database can be traced back to a public document.

Our core research question is methodological: can we reliably identify, classify, date, link and quantify distinct strategic projects disclosed by Russell 3000 companies, while preserving source evidence and without claiming exhaustive coverage of real-world activity? The answer offered here is deliberately two-sided. On the positive side, the pipeline is reproducible end to end, every intermediate artifact is versioned, and the full corpus has been processed with a single extraction contract and an audit ledger that records which sections were seen. On the cautious side, several components that a fully validated measurement instrument requires—an independent human-coded gold standard, precision and recall estimates by category and field, an audit of cross-filing linkage—are specified in this paper but have not yet been executed at scale. We therefore describe SPID as a *research database under validation*, and we keep a running register that separates implemented, documented-but-unverified, proposed and empirically verified claims (Appendix D).

The paper makes four contributions. First, it specifies the measurement target precisely: a *project* is a distinct, identifiable, issuer-associated strategic initiative described in one or more disclosures, as opposed to a *project mention* (a particular sourced statement), a *generic strategy* ("we invest in innovation"), a historical or third-party reference, or an uncertain candidate. Section 5

develops these distinctions and ties them to the existing 21-category SPID taxonomy, which we document rather than redesign. Second, it documents a scalable architecture: a frozen issuer universe reconciled against SEC identifiers; an acquisition layer that respects the SEC's fair-access limits by sharing one aggregate rate budget across ten cluster nodes and one external gateway; a parser that keeps raw-to-clean-text provenance; and an extraction ledger that makes the paid inference step resumable and idempotent (Section 4). Third, it exposes the extraction contract and the resolution rule in enough detail to be criticized: the prompt, the JSON schema, the confidence semantics, the deterministic project key, and the ways in which that key can both split and merge real initiatives (Section 6). Fourth, it lays out a validation protocol and a set of acceptance gates, and reports what the current build does and does not satisfy (Section 8).

Three caveats frame everything that follows. A present-day holdings file of an index-tracking fund is a proxy for index membership, not an authoritative historical constituent list; the universe is a snapshot, and survivorship is built into it (Section 3). A disclosed project is evidence that an issuer *described* an initiative, not evidence that the initiative happened, and a disclosed budget is a statement about intentions or estimates, not realized capital expenditure. Finally, the extractor only sees what the parser gives it: sections that were truncated, filings that failed to download, and exhibits outside the selected forms are invisible, and "zero projects" for an issuer-year must never be read as "no projects".

The rest of the paper is organized as follows. Section 2 positions the work in the text-as-data, corporate-investment and information-extraction literatures. Section 3 defines the universe and reports its reconciliation. Section 4 describes acquisition, parsing and the distributed build. Section 5 defines the project construct and the extraction contract. Section 6 covers mention-to-project resolution, timelines and amounts. Section 7 reports descriptive patterns. Section 8 presents the validation framework and the current audit. Section 9 discusses research applications and identification limits, and Section 10 concludes. Appendices provide the taxonomy, the data dictionary, the prompt, the evidence register and replication instructions.

2. Related Literature

Text as data in accounting and finance. The narrative content of SEC filings has become a standard empirical input. Loughran and McDonald (2011) showed that general-purpose sentiment dictionaries misclassify financial text and introduced finance-specific word lists; Loughran and McDonald (2016) survey the subsequent decade of textual analysis in accounting and finance, and Gentzkow et al. (2019) provide the general econometric framework for treating text as data, including the measurement-error and validation issues that any text-derived variable inherits. Dyer et al. (2017) document how 10-K disclosure has lengthened and become more boilerplate over time, a trend

that matters for us because repeated boilerplate is the principal source of false-positive "projects". Cohen et al. (2020) show that changes in otherwise repetitive filing language carry information, which motivates our emphasis on *repeated* mentions of the same initiative across filings rather than on single occurrences. Grimmer and Stewart (2013) offer the general methodological warning that automated text measures must be validated against human judgment on the specific task at hand, which is the stance adopted in Section 8.

Measuring corporate activity from disclosures. Closest to our construct are text-based measures of what firms do rather than how they feel. Hoberg and Phillips (2016) build product-market classifications from 10-K business descriptions; Hoberg and Maksimovic (2015) identify financially constrained firms from the language of the liquidity discussion; Hassan et al. (2019) measure firm-level political risk from the share of earnings-call text devoted to politics; Li et al. (2021) measure corporate culture with word embeddings; and Kelly et al. (2021) and Kalyani et al. (2025) track the emergence and diffusion of technologies through patents, calls and job postings. SPID differs from these approaches in its unit of observation. Rather than a firm-level score summarizing the intensity of a theme, it targets *discrete initiatives*—objects with a name, a type, a status, cited years, a location, partners and an amount—that can be followed from filing to filing. This is the project-centric view that commercial intelligence products sell, but with open provenance and a documented extraction contract.

Structured information extraction with language models. The extraction step relies on an instruction-tuned LLM used in a constrained-output mode: a fixed system prompt, a strict JSON object schema, temperature zero, and an explicit instruction to extract only initiatives grounded in the supplied passage. This design follows the general lesson of the text-as-data literature that the measurement instrument must be fixed, versioned and auditable, and it is deliberately conservative: the model is asked to prefer fewer, concrete projects to many vague ones, to return an empty list when no concrete project is described, and to attach a self-reported confidence. Semantic search over projects uses sentence embeddings (Reimers and Gurevych, 2019). We make no claim that self-reported confidence is calibrated; its calibration is one of the items of the validation protocol.

Entity resolution and record linkage. Linking repeated mentions of the same initiative across years is a record-linkage problem in the sense of Fellegi and Sunter (1969); Christen (2012) reviews the modern toolbox of blocking, comparison and classification. The current SPID resolver is a deterministic blocking rule (issuer, project type and a location or name token) rather than a probabilistic linker. We document its behavior and its failure modes in Section 6, and we treat the migration to an explainable, reversible linkage with human adjudication of ambiguous clusters as

part of the validation roadmap.

Infrastructure. The database is implemented as a single analytical file in DuckDB (Raasveldt and Mühleisen, 2019), with Parquet exports; acquisition respects the SEC's published fair-access policy (U.S. Securities and Exchange Commission, 2026), and the Russell 3000 is defined following FTSE Russell's published construction methodology (FTSE Russell, 2026), approximated here through a fund-holdings proxy (Section 3).

3. Sample: Issuer Universe and Filing Coverage

3.1. Estimand: current constituents at a snapshot date

Two different populations can be labeled "the Russell 3000". The first is the set of *historical point-in-time constituents*: for each date, the companies that FTSE Russell included in the index after its annual reconstitution (FTSE Russell, 2026). The second is the set of *current constituents at a snapshot date*. The two differ because of index entries and exits, mergers, delistings and IPOs; a panel built on the second population is subject to survivorship and backfilling biases that a point-in-time panel is not. SPID targets the second population. No licensed historical constituent list was available to the team at the time of the build, and FTSE Russell does not publish machine-readable membership lists for free. Any inference about the "Russell 3000 over 2010–2026" from this database is therefore an inference about firms that were in the proxy universe in October 2026 and about their filings back to 2010.

3.2. Universe construction and reconciliation

The issuer universe is frozen in a single JSON artifact, `data/metadata/universe.json`, built by `scripts/build_universe.py` and documented in `docs/universe_definition.md`. It merges two sources on the SEC Central Index Key (CIK):

1. the S&P 500 constituent list scraped from the public Wikipedia table on May 25, 2026 (503 rows, 500 distinct CIKs, with GICS sector and sub-industry), used for the first SPID release; and
2. the holdings file of the iShares Russell 3000 ETF (ticker IWV) published by iShares as `latest-holdings.csv`, with holdings as of October 23, 2026. IWV is an index-tracking fund; its holdings are a close but not exact replica of the index, and the file carries the fund's own sector labels.

Tickers are normalized to the SEC convention (share-class separators become hyphens: "BRK B" → `BRK-B`) and mapped to CIKs through the SEC's public `company_tickers.json`. Lines that are not operating companies (contingent value rights, escrow shares, rights when-issued) and tickers

without a CIK (banks that file with the FDIC rather than the SEC) are dropped and listed in the artifact's `unmatched` field. Table 1 gives the full reconciliation from input securities to issuers with resolved projects.

Table 1. Universe-to-dataset reconciliation

Step	
iShares IWV holdings file, equity lines with a ticker (as of Oct 23, 2026); count reported by the build log, source file not ar	
of which mapped to an SEC CIK (CVRs, escrow lines and FDIC-only banks excluded)	
distinct CIKs (share classes collapsed)	
S&P 500 constituents, cached snapshot of 2026-05-25 (500 CIKs)	
Merged issuer universe (distinct CIKs)	
in both lists	
S&P 500 only	
Russell proxy only	
Issuers with at least one 10-K, 10-Q or 8-K since 2010	
Issuers without any such filing (foreign private issuers filing 20-F/40-F, FDIC-only banks)	
Issuers with at least one resolved project	

Source: `data/metadata/universe.json`, `russell3000_constituents.json`, `sp500_constituents.json` and the merged
The 65 issuers without filings are listed in `data/issuers_without_filings.csv`; most are foreign private issuers (e.g., Spotify, C
Solar, Amdocs) that file annual reports on Form 20-F or 40-F, which are outside the current form scope, plus a few banks whose perio
with the FDIC. Seven issuers have filings but no resolved project.

Several design decisions deserve to be stated plainly. The proxy universe counts *issuers*, not securities: two share classes of the same registrant collapse into one CIK (2,587 equity lines map to 2,555 CIKs). Companies in the S&P 500 snapshot but no longer in the IWV file (nine issuers, typically recently acquired) are retained with an `in_sp500` flag so that the June 2026 release remains a subset. The universe table records, for each issuer, the SEC name, SIC code and description, state of incorporation, exchange and fiscal-year end from the SEC submissions header, the fund weight, and the cluster node that processed its filings (`shard_node`). The brief's recommendation of a `universe_snapshots` table with source checksums, membership dates and match confidence is only partially met: snapshot date, source URL and match method are recorded; file hashes and per-row match confidence are not yet stored and are listed as open items in the implementation audit (`docs/implementation_audit.md`).

3.3. Filing scope

For every issuer the pipeline enumerates the SEC submissions index (`data.sec.gov/submissions/CIK{cik}.json`, including the paginated archive blocks) and retains the primary document of every Form 10-K, 10-Q and 8-K filed on or after January 1, 2010. Amendments (`/A`), exhibits other than the primary document, proxy statements, registration statements and foreign-issuer forms are out of scope. The choice reflects where strategic initiatives are described in prose—Items 1, 1A, 2, 7 and 7A of the 10-K, the MD&A and risk-factor items of the 10-Q, and the body of the 8-K—and it is a defensible but not innocuous restriction: press releases attached as Exhibit 99 to an 8-K, which often carry the first and most detailed announcement of a project, are not read when they are not the primary document. Testing whether this scope omits important announcements is part of the validation roadmap.

Table 2. Filings in the corpus, by form

Form	Filings	First filing	Last filing	Raw HTML (GB)	Not parsed
10-K	31,775	2010-01-08	2026-10-27	88.6	3
10-Q	95,167	2010-01-05	2026-10-27	155.9	12
8-K	400,606	2010-01-04	2026-10-27	10.0	1
Total	527,548	2010-01-04	2026-10-27	254.5	16

"Not parsed" counts filings whose primary document could not be cleaned (parser error) or was not found on disk after download. Raw HTML size is the sum of primary-document sizes recorded at download time.

Table 2 summarizes the corpus: 527,548 primary documents, 254.5 GB of raw HTML, of which only 16 documents (0.003%) failed parsing. Figure 1 and Table 14 in the appendix show the coverage by year. The number of issuers with at least one filing rises from 1,392 in 2010 to 2,498 in 2026: the snapshot universe contains many companies that went public after 2010, so the early years of the panel are populated by the older and larger survivors. This is the concrete form taken by the survivorship caveat above.

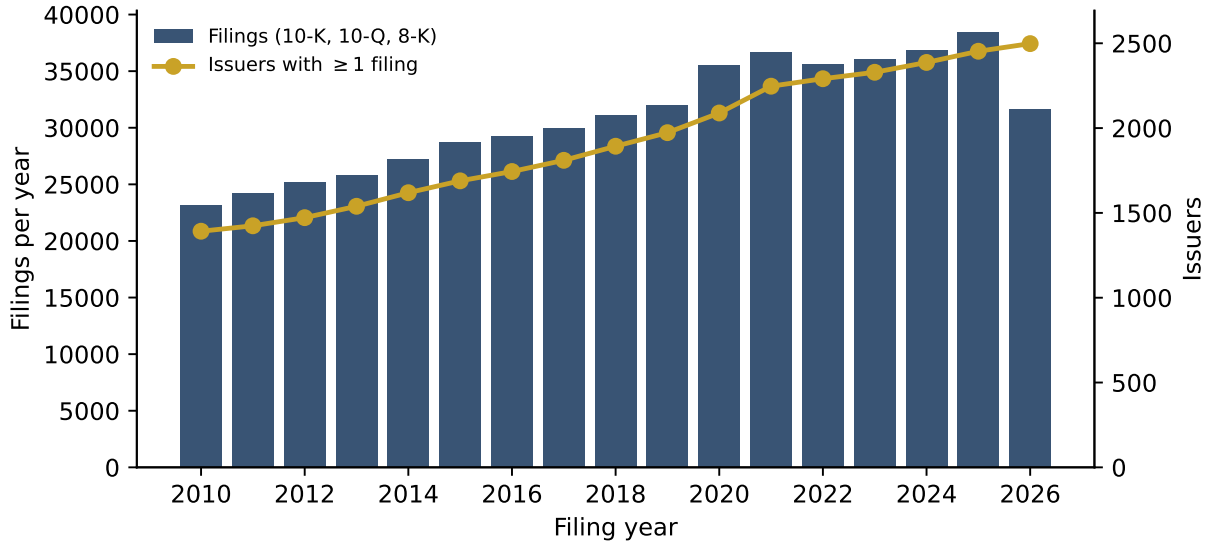


Figure 1. Filings and issuers per year in the corpus. The 2026 bar covers January 1 to October 27, 2026. Issuers that went public after 2010 enter the panel at their first filing, so the number of observed issuers increases mechanically over time under the snapshot design.

4. Architecture: Acquisition, Parsing and the Distributed Build

4.1. Overview

Figure 2 summarizes the pipeline. Two layers are distinguished throughout. The *corpus layer* (code in `src/` and `scripts/`) acquires and parses filings, chunks them, embeds the chunks and computes thematic factor scores; it is universe-agnostic and produces a DuckDB database (`edgar_research.duckdb`, 68 GB) plus Parquet exports. The *project layer* (code in `project-db/spid/`) reads the Parquet exports through views, runs candidate gating and LLM extraction, resolves mentions into projects, builds a knowledge graph and project embeddings, and persists everything in a second, much smaller DuckDB file (`spid.duckdb`, 3.2 GB). The separation matters for reproducibility: the expensive and paid step (LLM extraction) depends only on the frozen Parquet exports and on a ledger of processed sections, so it can be re-run, resumed or audited independently of the corpus build.

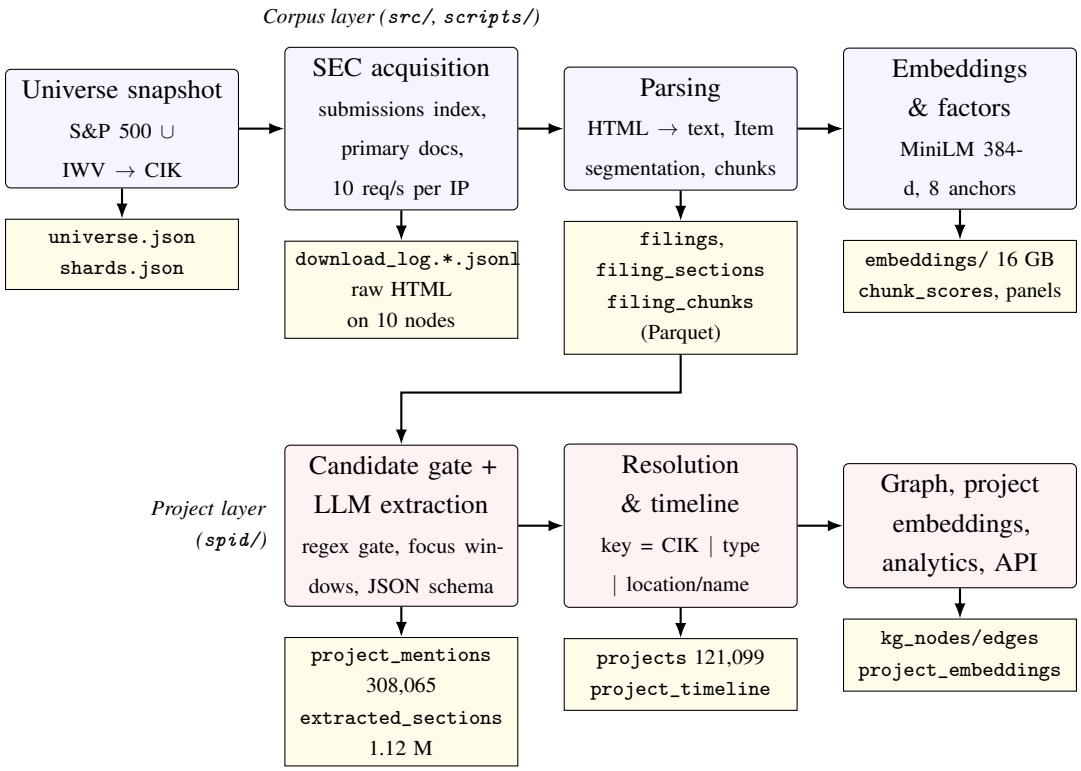


Figure 2. Pipeline architecture. Rounded boxes are processing steps; rectangles are persisted artifacts. The project layer reads the Parquet exports of the corpus layer through read-only views and never modifies them.

4.2. SEC acquisition

The SEC's fair-access policy allows ten requests per second per client and asks for a descriptive `User-Agent` with a contact address (U.S. Securities and Exchange Commission, 2026). Both constraints shaped the acquisition layer (`src/acquisition/bulk_downloader.py`). Requests are issued by a pool of threads behind a single token bucket; the bucket rate is a parameter, and when several processes share one public IP address the orchestrator divides nine requests per second among them. HTTP 403, 429 and 503 responses trigger a 65-second cool-down for the whole process rather than a per-request retry, so that a throttling signal is never answered with a burst. Downloads are idempotent: a filing is skipped when its file exists, and every completed download is appended to a JSON-lines log carrying the accession number, form, filing and report dates, primary document name, size and local path. A trusted-log option lets a node skip filings that an earlier corpus already contains without re-fetching them. The SEC submissions header of each issuer (name, SIC, state of incorporation, exchanges, fiscal-year end, former names) is saved alongside, and feeds the `companies` table.

The October 2026 build used eleven machines: ten Apple-silicon nodes of the UQO MacLustr cluster behind one residential IP address, and one Linux server in an OVHcloud data center in

Beauharnois, Québec, behind a second address. Issuers were assigned to nodes in proportion to CPU cores; half of each node's issuers were downloaded by the node itself (0.9 requests per second each) and half by the external server (one request per second per shard), then shipped to the node over SSH. A first allocation by contiguous CIK ranges proved badly unbalanced—old CIKs are large filers with up to 220 documents each—and was replaced after one hour by an interleaved allocation of the not-yet-started issuers. The whole acquisition (about 400,000 new documents) took seven hours at a sustained 17 requests per second, with no fair-access violation recorded.

4.3. Document processing

Each primary document is cleaned with `lxml`: `script`, `style`, `meta` and `link` elements are removed, tables are flattened to tab-separated text, lists are kept as line items, whitespace is normalized and files above 50 MB are skipped. The segmenter then locates "Item *n*" and "Part I/II" headings by regular expression: 10-Ks are split into up to fourteen sections (Items 1 to 9B), 10-Qs into seven (Part I Items 1–4, Part II Items 1, 1A, 2), and 8-Ks are kept as a single `full_body` section. A filing in which no heading is found is kept as one `full_document` section rather than dropped, so that the foreign-key chain from chunks to sections to filings is never broken. Sections are chunked into 512-word windows with a 64-word overlap for the embedding and factor layers (10.72 million chunks). Provenance is preserved at the section level: each section row stores its item number, order, word and character counts, and each chunk stores its character offsets within the section. Character-level offsets of sections within the raw HTML are *not* stored, a limitation recorded in the audit.

Table 3 reports the outcome. The 10-K sections are long (median 875 words, mean 5,551) and the 10-Q sections somewhat shorter; 8-K bodies are short (median 575 words). Two coverage statistics deserve emphasis because they bound what the extractor could see: 0.003% of filings failed parsing, and the segmenter's reliance on textual headings means that a 10-K whose items are rendered as images or in unusual markup collapses into a single `full_document` section (1,230 such 10-K sections were sent to the extractor).

Table 3. Parsed sections and sections submitted to the extractor, by form

Form	Parsed sections	Median words	Ledgered sections ($\geq$ 30 words)	Gate-passing calls	Sections with $\geq$ 1 mention	Mentions returned by the model (ledger)	Mentions stored	Issuers
10-K	341,207	875	143,704	135,310	30,877	81,622	81,495	2,464
10-Q	600,471	500	578,793	363,973	75,705	152,261	152,004	2,494
8-K	400,605	575	400,601	204,442	61,648	74,594	74,566	2,499
Total	1,342,283		1,123,098	703,725	168,230	308,477	308,065	2,499[†]

"Sections sent to the extractor" counts rows of the `extracted_sections` ledger: sections with at least 30 words, restricted for 10-K
3–6, 8, 9–9B, which are not submitted.

4.4. Distributed build and merge

Parsing, chunking and embedding ran on every node against the node's own shard, using the same code revision and a local DuckDB file; each node then exported its tables to Parquet. The master node (M3U96b, which also holds the original S&P 500 corpus) pulled the shards and merged them with an append-only rule: a row is inserted only if its primary key (accession number, section or chunk UUID) is absent, so that re-running the merge cannot create duplicates. The merged database holds 2,564 issuers, 527,548 filings, 1,342,283 sections, 10,720,890 chunks and as many 384-dimensional embeddings. Parsing throughput was about 100 filings per second per node (five to thirteen minutes per shard); embeddings took 15–40 minutes per node on Apple-silicon GPUs; the merge and the factor computation took seven minutes. The orchestration script (`scripts/cluster/orchestrate.py`) is stage-based and idempotent: every stage polls marker files on the nodes and can be restarted without side effects. Private infrastructure details (addresses, credentials, keys) are kept outside the repository and outside this paper.

4.5. Versioning

The repository is versioned in a private Git server; the revision used for the build described here is `2324b58` (October 28, 2026), which supersedes revision `df7905d` examined in the June 2026 methodological study (Tandjambianda et al., 2026). Model and prompt versions are recorded in the extraction ledger as the backend name; the OpenAI model identifier returned by the API during the October run was `gpt-4o-mini-2024-07-18`. A hash of the prompt and of the taxonomy is *not* yet stored per run—an item on the audit list—so prompt identity is currently guaranteed by the Git revision rather than by a per-run digest.

5. The Project Construct and the Extraction Contract

5.1. What counts as a project

The unit of measurement is a *distinct, identifiable, issuer-associated strategic initiative* involving a planned, announced, ongoing, completed, postponed or cancelled action: a plant expansion, a research program, a technology deployment, an AI initiative, a data-center build, a supply-chain transformation, an acquisition and its integration, a renewable-energy investment, and the other categories of the taxonomy. The database stores two tiers and, in its intended final form, five labels:

- `project_mention`: a particular sourced statement about an initiative, in one section of one filing, with the extracted fields and the model's one-sentence description;
- `project`: a resolved initiative described by one or more mentions, carrying a stable identifier and a timeline;

- `generic_strategy`: non-specific aspirations ("we continue to invest in innovation", "we may pursue acquisitions") that the contract instructs the model to exclude and that the resolver never sees unless the model fails to exclude them;
- `historical_or_third_party_reference`: initiatives of a competitor, a customer or a former parent, mentioned in the issuer's filing but not attributable to the issuer;
- `uncertain_candidate`: a plausible initiative without enough identity evidence (no name, location, counterparty or amount) to be counted.

The current build implements the first two tiers directly. The third is enforced only through the prompt and the confidence threshold; the fourth and fifth exist as instructions and as a confidence score, not as stored labels. Promoting them to explicit columns of the mention table, filled by annotators and eventually by a classifier, is the first schema change proposed in `docs/project_ontology.md`.

5.2. Taxonomy

The SPID taxonomy has 21 categories (Table 11 in Appendix A), defined in `spid/taxonomy.py` with a human-readable label and a set of lexical signals used by the rule-based extractor and by the candidate gate. The LLM extractor receives the 21 labels verbatim in its system prompt and must choose exactly one per project; a label outside the list is mapped back to the closest category by substring matching or, failing that, to `technology_deployment`. The categories are not mutually exclusive in substance (an acquired plant is both `ma_integration` and `plant_construction`; a data center is also `infrastructure`), and the model's choice in such cases is a classification decision we have not yet audited. We document the taxonomy as it stands rather than redesign it: any extension would be versioned, with a crosswalk from old to new codes, so that counts published under the current version remain reproducible.

5.3. Candidate gating and focus windows

A section reaches the model only if it contains at least 30 words and matches a broad lexical gate (project, initiative, program, invest, expan-, construct, build, launch, develop, deploy, implement, acqui-, partner, facilit-, capacity, capital, migrat-, transform, pipeline, plant, data center, automat-, sustainab-, renewable, restructur-; see Appendix B). The gate is deliberately permissive—it exists to avoid paying for sections that cannot contain a project—and its false-negative rate is unknown; sampling rejected sections to estimate it is part of the validation plan. For sections longer than 9,000 characters, the extractor keeps windows of about 600 characters around each gate hit (merged when they overlap) up to a budget of roughly 9,000–12,600 characters, so that the cost per call is bounded regardless of section length. This "focus" step is the main source of *silent truncation* in the pipeline:

a project described far from any gate word, or beyond the budget in a very long MD&A, is never shown to the model. The 10-K MD&A sections submitted have a median length of 9,905 words (roughly 60,000 characters), so truncation is the rule rather than the exception for annual reports, and the recall of the pipeline on long sections is an open empirical question.

5.4. The extraction contract

The LLM call is a chat completion with a fixed system prompt, the section name and the focused text as user message, temperature 0, JSON-object output mode and a 2,000-token completion limit. The prompt (Appendix B) instructs the model to act as a financial analyst extracting corporate projects, lists the 21 allowed types, and returns a JSON object with one entry per project and the following fields: `project_name`, `project_type`, `description` (one sentence grounded in the text), `objective`, `status` (planned, in_progress, completed, mentioned), `amount_usd`, `locations`, `technologies`, `partners`, `benefits`, `risks`, `years` and `confidence`. Four rules in the prompt define the construct operationally: extract only real initiatives, not boilerplate or hypotheticals; return an empty list when no concrete project is described; do not invent facts—numbers and locations must appear in the text; prefer fewer high-quality projects. Output is post-processed deterministically: amounts are coerced to a float in USD (with the raw string retained in `amount_raw`), list fields are truncated to eight items of at most 120 characters, years are parsed as four-digit integers, status is normalized, and mentions with confidence below 0.45 are dropped. The mention identifier is a SHA-1 of the accession number, section identifier, type and project name, so that re-running the extractor on the same section with the same output overwrites rather than duplicates.

What the contract does *not* guarantee should be equally explicit. The "description" is a model paraphrase, not a verbatim quote, and the character span of the supporting passage is not stored; grounding is therefore verifiable by a reader who opens the section, but not mechanically checkable. The model may extract a counterparty's initiative as the issuer's (the third-party label above). It may split one initiative into several entries when a section describes phases, or merge phases into one. It receives no information about the issuer other than the text, which protects against hallucinated background knowledge but also prevents it from recognizing a subsidiary name. And the same section is seen once; a project described across two sections of the same filing yields two mentions with distinct identifiers.

5.5. Cost, throughput and the extraction ledger

The October 2026 run submitted 968,134 sections in about twelve hours of effective extraction (48 concurrent calls, 27–36 sections per second), at a cost of roughly 775 USD for the full universe at current `gpt-4o-mini` prices (about 0.0008 USD per section). Every submitted section is written to

an `extracted_sections` ledger (section, accession, backend, number of mentions, timestamp), which makes the step resumable and prevents double billing. The ledger also protects against a failure mode encountered during the run: when the API key's credit balance was exhausted, the extractor's retry logic returned empty results that were recorded as "extracted with zero mentions" for 149,352 sections before the problem was detected. Those rows were removed from the ledger using the timestamp of the first quota error and the sections were re-submitted; the final ledger has no section processed during the outage. The episode is recorded in the research log because it illustrates a general point: a ledger that cannot distinguish "no project" from "call failed" is not sufficient, and an explicit `call_status` column is now part of the proposed schema.

6. Resolution, Timelines and Amounts

6.1. The resolution rule

Mentions are grouped into projects by a deterministic key computed in SQL (`spid/resolve.py`): the issuer CIK, the project type, and either the lower-cased first location of the mention or, when no location was extracted, the first token of at least four letters in the normalized project name ("general" when none exists). The project identifier is the MD5 hash of that key. Within a group, the representative name and description are those of the highest-confidence mention, the canonical location is the group's location token, the technologies are the union of the mentions' technologies, the status is that of the *latest* mention by filing date, the amount is the *maximum* across mentions, and the timeline is the ordered list of mentions with their filing date, form, status, amount, confidence and snippet. First and last cited years are the minimum and maximum of the years mentioned in the text; first and last *seen* are the filing dates of the earliest and latest mention.

The rule is transparent, cheap and reversible (the mentions are untouched), and it behaves well for the common case of a recurring physical program at one site. Its failure modes are equally clear:

- **False merges.** All mentions of one type at one location collapse into one project: every acquisition closed by a serial acquirer in the same city, every expansion of a multi-phase site (the Cheniere example of Table 9), every drug candidate of a pharmaceutical company with no location, which fall back to a name token that is often shared ("phase", "program", "study"). The mean of 4.99 mentions per `drug_pipeline` project versus 1.63 for `technology_deployment` partly reflects this.
- **False splits.** The same initiative named differently across filings ("Project Atlas" in 2019, "our ERP transformation" in 2021) or located at a different granularity ("Texas" vs. "Austin, Texas") yields distinct keys; so does a change of category by the model.
- **No cross-issuer identity.** A joint venture disclosed by both partners is two projects; a project inherited through an acquisition is a new project of the acquirer.

The 60.2% of projects with a single mention are the population in which false splits, true one-off disclosures and false positives are mixed, and the projects with more than ten mentions (3.1%) are where false merges concentrate. The brief's requirement—explainable, reversible merge/split decisions with match scores, resolver version and human adjudication of ambiguous clusters—is not met by the current rule and is the second schema change proposed: a `project_mention_links` table with a match score and an adjudication status, fed first by the deterministic key (score 1 by construction) and then by a probabilistic linker that uses name similarity, embedding similarity, location and time, with ambiguous clusters routed to review.

6.2. Timelines and point-in-time structure

Each mention is dated by the filing date of its document, and each project by the filing dates of its first and last mentions and by the years cited in the text. This is a point-in-time structure in one specific sense: the state of a project *as known at* a filing date can be reconstructed by filtering the timeline to mentions filed on or before that date, and nothing in a later filing is used to alter an earlier mention. It is *not* yet the event structure the brief describes. The database does not separate first announcement, SEC acceptance timestamp (only the filing date is stored), referenced period, expected start, observed start, expected completion, realized completion and cancellation; cited years are stored as an unordered list without roles; and a project's status is the latest statement rather than a dated transition. The proposed `project_events` table (Appendix C) would record each such event with its date, its evidence mention and a flag distinguishing information available at the acceptance date from information revised later.

6.3. Amounts

The amount of a mention is whatever USD figure the model associated with the project in the passage, coerced to a float; the raw string is retained. The project-level `total_amount_usd` is the *maximum* over mentions, chosen in the June 2026 release because the same budget is typically restated in successive filings and summing would multiply it by the number of mentions. The maximum avoids that error but introduces others: it cannot tell a transaction price from a capex budget, a contract value from a financing facility, an incurred cost from an estimate, or a figure for the whole program from a figure for one phase, and it is silently inflated when a merged project combines distinct initiatives. We therefore insist on two reading rules. First, `total_amount_usd` is "the largest amount an issuer cited in connection with this project", nothing more. Second, amounts must never be summed across projects or interpreted as realized investment. The proposed `project_amounts` table keeps every amount mention with its original value, currency, scale, reporting date, an `amount_type` (announced budget, estimated cost, capex incurred, transaction value, contract value, financing,

other) and an estimate/realization flag; filling `amount_type` requires either a second extraction pass or annotation, both of which are in the validation plan.

6.4. Data model: current and proposed

Table 12 in Appendix C lists the tables of the current build alongside the tables required by the brief and marks each as implemented, partial or proposed. In summary, the current `companies`, `filings`, `filing_sections`, `project_mentions`, `projects`, `project_timeline` and `extracted_sections` tables cover universe, filings, sections, mentions, projects, a timeline and an extraction ledger; the `issuers` history, `project_mention_links`, `project_events`, `project_amounts`, `extraction_runs` (with prompt, schema and taxonomy hashes) and `human_labels` tables are proposed and would be added by migration, never by destructive replacement.

7. Descriptive Patterns

All numbers in this section are produced by `code/export_stats.py` and `code/make_tables.py` from the merged databases at revision `2324b58`; none is typed by hand. They describe *disclosed* initiatives as detected by the current pipeline and resolved by the current rule. They are not estimates of real-world project counts, and they have not yet been corrected for the extraction and linkage errors whose measurement is described in Section 8.

7.1. From sections to projects

The extraction ledger contains 1,123,098 sections. The model returned at least one project for 168,230 of them (15.0%), yielding 308,065 mentions—0.27 per section on average and 1.83 per productive section. The density differs sharply by form and by item (Table 4): the MD&A items of the 10-K and 10-Q produce 39,099 and 99,305 mentions respectively, Item 1 (Business) of the 10-K 30,492, and the 10-Q financial-statement notes 30,159, whereas risk-factor and market-risk items produce almost none despite their length. The 8-K body yields 74,594 mentions from 400,601 filings. Resolution reduces 308,065 mentions to 121,099 projects (2.54 mentions per project); 60.2% of projects rest on a single mention, 17.0% on two, and 8.2% on more than five (Table 7).

Table 4. Sections submitted and mentions obtained, by form and item

Form	Section (item)	Ledgered sections	Mentions	Gate passed (calls)
10-K	`business`	29,285	30,492	29,118
10-K	`risk_factors`	29,158	2,797	28,941
10-K	`md_and_a`	28,919	39,099	28,050
10-K	`properties`	28,236	7,879	25,940
10-K	`quantitative_disclosures`	26,876	531	22,032
10-K	`full_body`	1,230	824	1,229
10-Q	`controls_and_procedures`	92,393	7,082	20,913
10-Q	`financial_statements`	91,882	30,159	90,106
10-Q	`md_and_a`	91,371	99,305	90,269
10-Q	`quantitative_disclosures`	83,138	808	46,720
10-Q	`unregistered_sales`	80,210	9,535	61,867
10-Q	`risk_factors`	72,560	4,420	33,826
10-Q	`legal_proceedings`	66,998	845	20,031
10-Q	`full_document`	241	107	241

8-K filings (400,601 sections, 74,594 mentions) are a single `full_body` section each and are omitted from the table. `full_body`/`full_document` rows are filings in which no Item heading was recognized.

7.2. Composition by category, sector and status

Table 5 and Figure 3 show the distribution over the 21 categories. M&A and post-merger integration dominate (34.5% of projects), followed by capital-expenditure programs (8.9%), strategic partnerships (8.2%), geographic expansion (6.5%) and plant or site construction (6.3%). The technology-related categories that motivated the first SPID release—data centers, AI initiatives, cloud migration, automation—are small in count (between 264 and 449 projects each) but carry high disclosed amounts (median 128–166 M USD for data centers and AI initiatives). Two readings of this composition are possible and the data do not discriminate between them: either acquisitions are genuinely the most frequently disclosed discrete initiative in SEC filings, which is plausible given that every material acquisition triggers an 8-K and recurring 10-Q notes; or the extraction contract, which asks for "concrete initiatives", is more easily satisfied by acquisitions (named counterparty, date, price) than by internal programs. Disentangling the two is a question for the human-annotation study.

Table 5. Resolved projects by category

Code	Category	Projects	Share (%)	With amount (N)	Median amount (M USD)	95% CI of the median	Mentions / project	≥ 3 distinct names (%)
ma_integration	M&A / integration	41,742	34.5	28,044	155.0	–	3.12	14.6
capex_program	Capital expenditure program	10,720	8.9	9,452	300.0	–	1.79	6.4
partnership	Strategic partnership	9,909	8.2	2,943	75.0	–	1.77	6.7
geographic_expansion	Geographic expansion	7,868	6.5	1,431	36.7	–	2.25	10.2
plant_construction	Plant / site construction	7,590	6.3	2,511	56.0	–	2.42	13.5
cost_reduction	Cost reduction / restructuring	6,481	5.4	4,378	50.0	–	2.03	5.7
drug_pipeline	Drug / clinical pipeline	5,360	4.4	778	46.1	–	4.99	27.9
product_launch	New product / launch	5,356	4.4	447	26.7	–	2.33	12.2
rd_program	R&D program	5,080	4.2	1,272	20.0	–	2.31	11.4
infrastructure	Infrastructure investment	4,481	3.7	1,694	48.0	–	1.90	7.5
manufacturing_expansion	Manufacturing / capacity expansion	4,140	3.4	1,040	45.0	–	2.13	11.3
technology_deployment	Technology deployment	4,128	3.4	1,219	135.0	–	1.63	4.8
digital_transformation	Digital transformation	2,642	2.2	240	34.2	–	1.99	6.2
erp_implementation	ERP / systems implementation	1,735	1.4	184	15.0	–	2.44	3.5
sustainability	Sustainability / ESG	1,305	1.1	410	34.4	–	2.00	6.1
supply_chain	Supply-chain initiative	684	0.6	139	38.0	–	1.71	3.5
data_center	Data center	449	0.4	129	166.0	–	1.98	9.3
energy_transition	Energy transition / renewables	415	0.3	109	109.0	–	2.09	10.1
ai_initiative	AI / ML initiative	376	0.3	20	127.5	–	1.68	4.8
automation	Automation / robotics	374	0.3	25	13.7	–	1.68	2.7
cloud_migration	Cloud migration	264	0.2	35	50.0	–	1.97	5.7
	Total	121,099	100.1	56,500	128.5	–	2.54	

"With amount" counts projects for which at least one mention carries a USD figure; "median amount" is the median across those pr

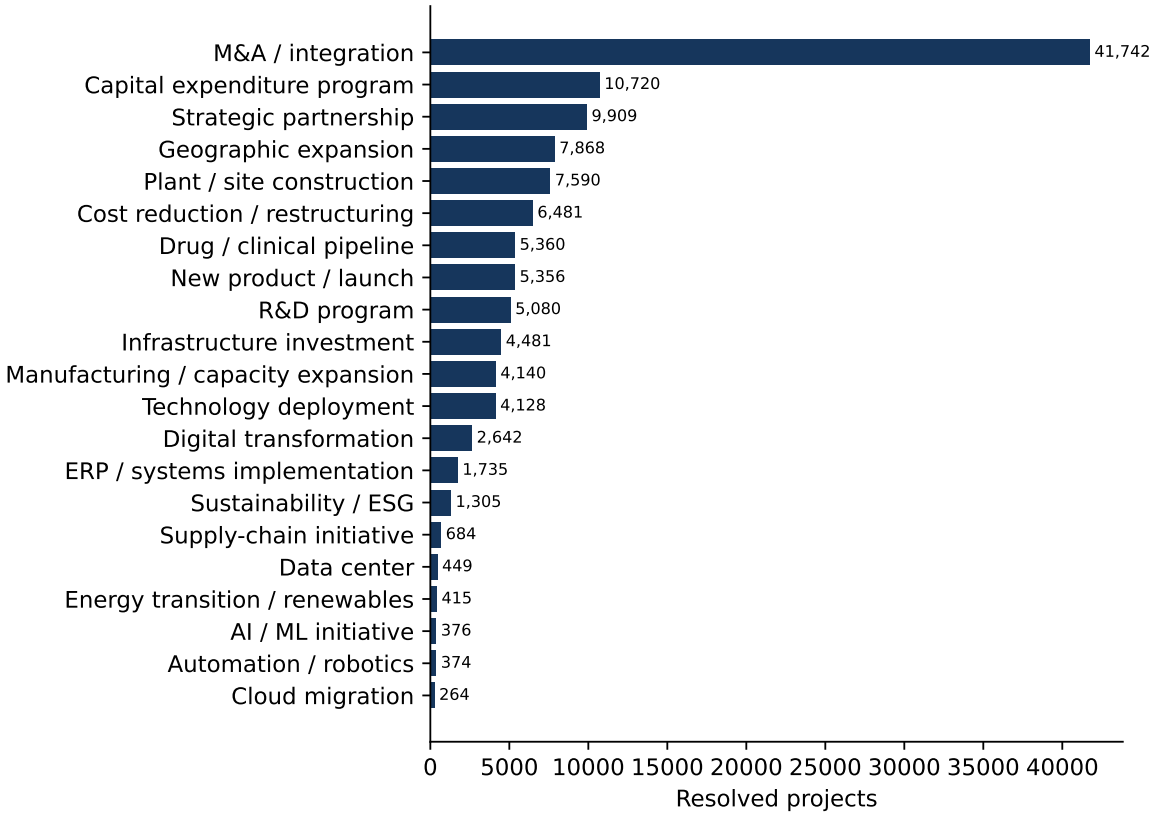


Figure 3. Resolved projects by taxonomy category (121,099 projects).

By sector (Table 6), health care (23,679 projects, 429 issuers) and industrials (17,890) contribute

the most projects; utilities, materials, energy and real estate have the most projects per issuer (63–75), and financials the fewest (32), consistent with the intuition that asset-heavy, regulated or resource-extracting firms narrate discrete physical programs while banks describe fewer discrete initiatives. The list of the most "productive" issuers (Table 15 in the appendix)—uranium and gold miners, airlines, a bitcoin miner, an oilfield-services company—illustrates a second mechanism: issuers whose business *consists* of projects (mines, routes, wells, data-center sites) and whose filings enumerate them one by one.

Table 6. Issuers and projects by sector

GICS sector (iShares)	Issuers in universe	of which S&P 500	Issuers with projects	Projects	Projects per issuer
Health Care	433	59	429	23,679	55.2
Industrials	393	79	386	17,890	46.3
Consumer Discretionary	276	48	267	15,072	56.4
Financials	462	76	446	14,223	31.9
Information Technology	333	73	325	11,756	36.2
Real Estate	146	31	145	9,001	62.1
Materials	126	26	117	7,407	63.3
Energy	119	21	107	6,710	62.7
Consumer Staples	103	36	103	5,921	57.5
Utilities	68	31	66	4,920	74.5
Communication Services	105	20	101	4,520	44.8

Sector labels come from the iShares holdings file (Russell proxy) or, for S&P 500-only issuers, from the Wikipedia GICS table. "Issue in universe" includes issuers without filings.

The latest status recorded for a project is `in_progress` for 41.6%, `completed` for 41.0%, `planned` for 15.1% and `mentioned` for 2.3% (Table 7). Because status is taken from the most recent mention, it is a snapshot of the last thing the issuer said, not the terminal fate of the project: a project that was announced, mentioned once as in progress and then silently abandoned is recorded as in progress. The span between the earliest and latest year cited *in the text* is zero for 55.5% of projects with known years and exceeds five years for 8.5%; the brief's distinction between announcement, filing, expected start, observed start, expected and realized completion, and cancellation is not yet implemented as separate dated events (Section 6).

Table 7. Status, number of mentions and cited time span of resolved projects

Latest status	Projects	Mentions per project	Projects	Span of cited years	Projects
`in_progress`	50,327	1	72,932	0 years	67,170
`completed`	49,687	2	20,544	1-2 years	17,307
`planned`	18,340	3-5	17,681	3-5 years	8,697
`mentioned`	2,745	6-10	6,173	>5 years	10,326
		>10	3,769	unknown years	17,599

7.3. Amounts, confidence and time

A USD amount is attached to 120,240 mentions (39.0%) and to 56,500 projects (46.7%). Table 8 reports the distribution of the largest amount cited per project: median 128.5 M USD, interquartile range 23–500 M, and a long right tail (P99 of 12.5 B USD). The sum of these maxima across projects is 55.9 trillion USD, a number we report only to make the point that it must *not* be interpreted: it adds transaction values, capex budgets, contract values, financing amounts and estimated costs across projects and years, it double counts initiatives that the resolver split, and it treats a stated budget as if it were spent. Section 6.3 explains how amounts are stored and what an `amount_type` field would need to add.

Table 8. Distribution of disclosed amounts and of projects per issuer

	P10	P25	Median	P75	P90	P99
Largest amount cited per project, M USD (N = 56,500)	3.2	23	128.5	500	1,503	12,501
Projects per issuer (N = 2,492)	12	24	42	67	94	140

The self-reported confidence of mentions is concentrated at the top of the scale (Figure 4; Table 16 in the appendix): 35.2% of mentions have confidence 1.0, 32.7% have 0.9, and only 1.6% fall at or below 0.6. Mentions below 0.45 are discarded at extraction time. The distribution says more about the model's reporting habits than about accuracy; whether a confidence of 0.7 corresponds to a lower precision than a confidence of 1.0 is precisely the calibration question that only labeled data can answer.

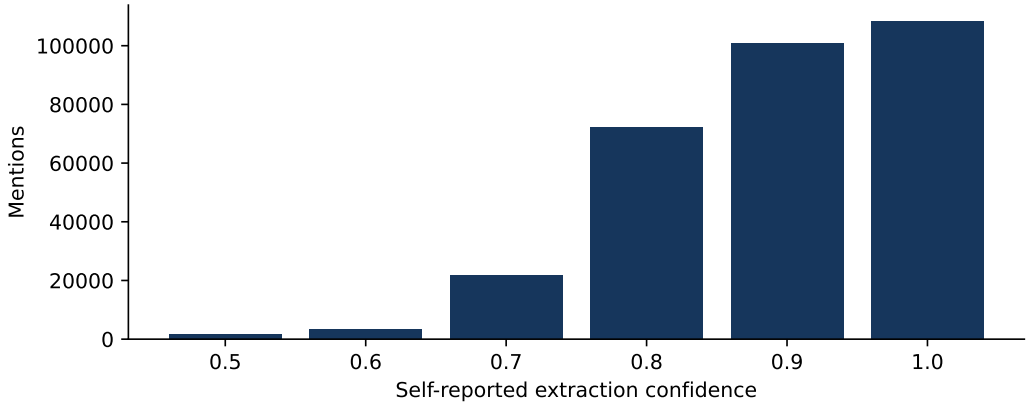


Figure 4. Self-reported extraction confidence of the 308,065 mentions (rounded to one decimal; mentions below 0.45 are not stored).

Over time (Figure 5; Table 14), mentions rise from 11,908 in 2010 to 23,220 in 2025, almost in proportion to the number of issuers observed (1,224 to 2,280 with at least one mention), while the number of projects first observed in a year is flat at 6,200–8,200. The rise is therefore mostly a coverage effect of the snapshot universe, not evidence of an increase in strategic activity; the flat count of new projects per year combined with a growing issuer base implies a declining number of new projects per issuer, which may reflect the lower disclosure intensity of the smaller firms that enter the panel later, or the resolver's tendency to attach later mentions to existing keys. Both explanations are testable with the panel.

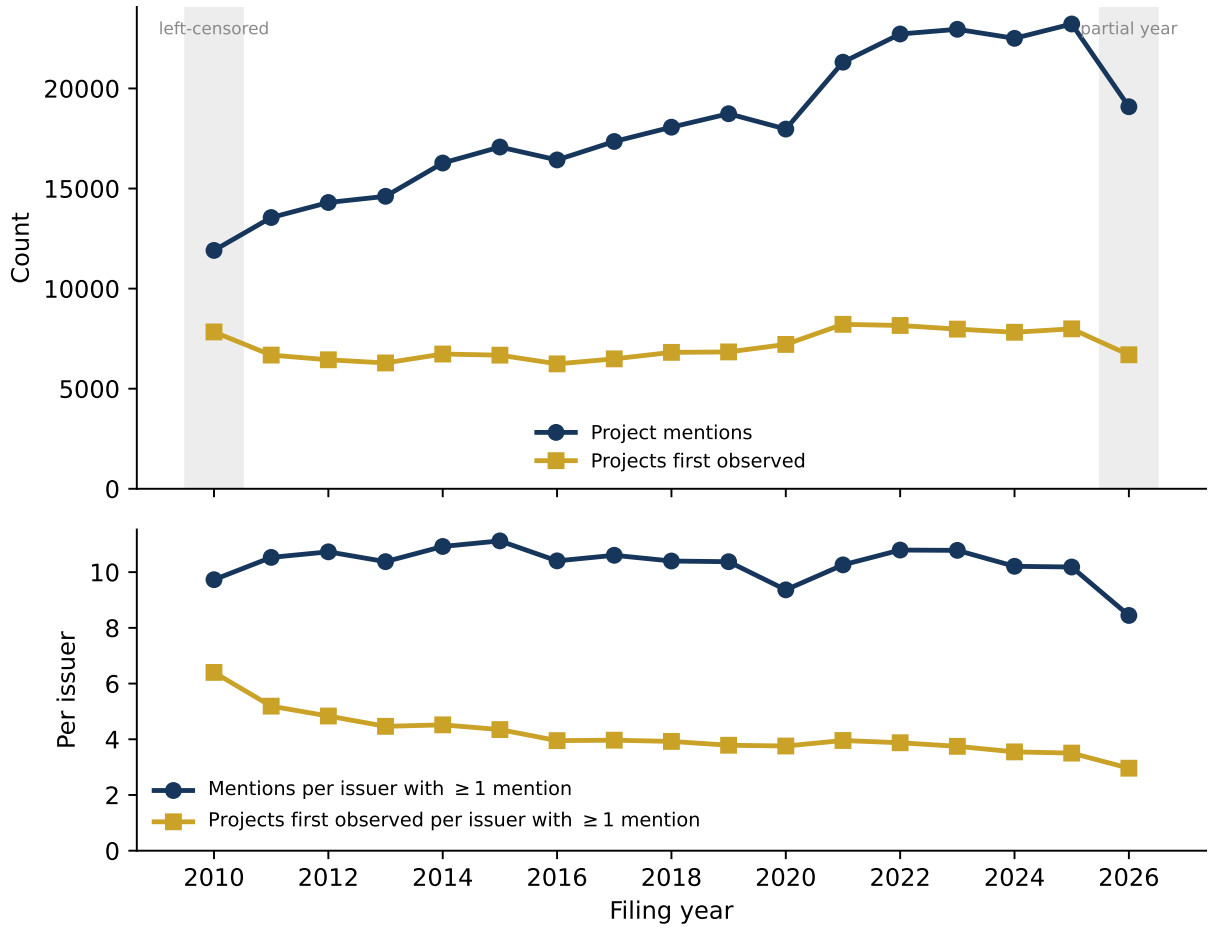


Figure 5. Project mentions and projects first observed, by filing year (2026 covers January to October 27).

7.4. A provenance example

Table 9 follows one resolved project through its mentions: the Corpus Christi liquefaction expansion of Cheniere Energy (ticker LNG), typed `plant_construction`, with eight mentions in 8-K and 10-K filings between April 2014 and March 2022 and a largest cited amount of 11.5 B USD. Each row is traceable: the accession number identifies the document on EDGAR, the section identifier locates the passage in the parsed corpus, and the snippet is the model's one-sentence description, which the contract requires to be grounded in the passage. The example also exposes the resolver's coarseness: the project key for this issuer and type is the location token "corpus christi", so the Stage 3 expansion and the earlier trains share one project identifier although Cheniere describes them as distinct phases. A reviewer can see this immediately because the mentions and their snippets are preserved; a user who consumes only the `projects` table cannot.

Table 9. Provenance of one resolved project: Cheniere Energy, Corpus Christi liquefaction (project `3e2705ba...`)

Accession	Filed	Form	Status / amount	Evidence snippet (model-generated description grounded in the section)
`0000003570-14-000070`	2014-04-08	8-K	planned / –	The construction of the first liquefaction train at the Corpus Christi Liquefaction facility is contingent upon receiving regulatory approvals and securing financing.
`0000003570-14-000114`	2014-05-30	8-K	planned / –	CCLNG will construct a second liquefaction train at its facility to sell LNG to IBERDROLA under a long-term agreement.
`0000003570-14-000117`	2014-06-02	8-K	planned / –	Cheniere Energy, Inc. plans to construct a second liquefaction train at the Corpus Christi Liquefaction facility.
`0000003570-14-000139`	2014-06-30	8-K	planned / –	Cheniere Energy, Inc. plans to construct a second liquefaction train at the Corpus Christi Liquefaction facility.
`0000003570-14-000153`	2014-07-17	8-K	planned / –	Cheniere Energy, Inc. plans to construct a third liquefaction train at its Corpus Christi Liquefaction facility.
`0000003570-14-000306`	2014-12-18	8-K	planned / –	Cheniere Energy, Inc. plans to construct a third liquefaction train at its Corpus Christi facility.
`0001193125-15-186045`	2015-05-13	8-K	in_progress / 11.5 B USD	The project involves the development, construction, operation, and maintenance of a natural gas liquefaction and export facility and related infrastructure in San Patrici
`0000003570-22-000029`	2022-03-07	8-K	in_progress / 5.5 B USD	The project involves the engineering, procurement, construction, pre-commissioning, commissioning, start-up, and testing of a mid-scale natural gas liquefaction facility

Snippets are the first 170 characters of the model-generated description stored in `project_timeline.snippet`; the full source section is retrievable through `project_mentions.section_id`. Amounts are shown only for mentions that cite one.

7.5. Knowledge graph and research-ready exports

The resolved projects are projected into a graph of 183,472 nodes (2,492 companies, 121,099 projects, 14,508 locations, 8,551 technologies, 36,822 partners) and 204,456 edges (`owns`, `located_in`, `uses`; Table 17 in the appendix); partner nodes are currently created without edges, a known gap. Each project also has a 384-dimensional sentence embedding of its name, type, description and technologies, which supports semantic search through the companion web application. Alongside the project layer, the corpus layer publishes the firm-year and firm-quarter panels of thematic factor scores (32,852 and 128,043 rows) used in earlier UQO work on AI

disclosure, now covering the full universe.

8. Validation Framework and Current Audit

The aim of identifying "all projects" of an issuer is a *coverage target*, not an assumption. Only projects that appear in the captured sources, survive parsing and gating, are recognized by the model and are correctly linked are observable, and each of these steps loses or distorts some. This section specifies how those losses are to be measured and reports what has been measured so far. The headline is simple: the *coverage* and *integrity* components are implemented and quantified; the *accuracy* components—which require independent human labels—are specified but not yet executed. Table 10 summarizes the status of the acceptance gates.

8.1. Coverage and integrity checks (implemented)

- **Universe reconciliation** (Table 1): every input line of the Russell proxy is accounted for as matched, dropped-as-non-company, or unmatched; the 65 issuers without filings are listed by name and reason.
- **Acquisition completeness:** for every issuer the SEC submissions index is enumerated in full (recent block and archive blocks) and every retained filing is logged; download failures are counted per issuer and were zero in the final run. Filing counts by form and year are published (Table 14).
- **Parsing coverage:** 16 of 527,548 primary documents could not be parsed (Table 2); filings without recognizable Item headings are kept as a single section rather than dropped.
- **Extraction ledger:** every section submitted to the model is recorded; the ledger was audited after the credit-exhaustion incident and all sections processed during the outage were re-submitted. The ledger does not yet record a per-call status, so an undetected API failure would still appear as "no project"; the run logs (retried errors, JSON parse failures: five in the June run, one in the October run) are the only current evidence that such failures were rare.
- **Referential integrity:** foreign keys from chunks to sections to filings to companies are enforced by the database; the merge of ten shards is append-only on globally unique keys and was checked for duplicate accession numbers and section identifiers (none).
- **Reproducibility of exhibits:** every table and figure of this paper is regenerated from the databases by two scripts in `code/`; the numbers can be compared against `data/stats.json`.

8.2. Accuracy evaluation (specified, not executed)

Gold standard. A stratified sample of sections will be drawn from the ledger with strata defined by form (10-K, 10-Q, 8-K), item, issuer size (fund weight tercile), sector, filing year (2010–2014,

2015–2019, 2020–2026) and disclosure length. The sample will include sections that produced mentions and, in equal number, randomly selected sections that produced none, so that both false positives and false negatives can be estimated. Two annotators will independently code each section: for each initiative present, its name, category (from the 21-code taxonomy), status, cited years, location, partners, amount and amount type, plus the five-way label of Section 5 (project, generic strategy, third-party, uncertain, none). Disagreements will be adjudicated by the third team member; inter-annotator agreement (Cohen's κ for presence and category, exact-match rates for fields) will be reported. The sample will be split into a development half, on which prompt and threshold changes may be tuned, and a locked holdout half, used once per release.

Metrics. On the holdout: candidate-gate recall (share of annotated projects whose section passed the gate); project-detection precision, recall and F1 at the mention level; category macro- and micro-F1; field accuracy for amount, years and location; the abstention rate (sections where the model returned an empty list) conditional on annotated content; and confidence calibration (precision by confidence bin, with bootstrap intervals). The same metrics will be computed for the rule-based extractor and, if budget allows, for a semantic-retrieval baseline, so that the marginal value of the LLM step is documented.

Linkage audit. From the resolved projects, a sample stratified by number of mentions (1, 2–5, 6–10, more than 10) and by category will be reviewed to label each project cluster as correct, over-merged (contains distinct initiatives) or under-merged (a duplicate of another cluster of the same issuer). Link precision and recall and the error costs of false merges versus missed links will be reported, with particular attention to long-running programs, subsidiaries and near-duplicate names.

Temporal test. For a sample of projects with at least three mentions, the status and amount reconstructed using only mentions filed on or before each date will be compared with the full-information values, to verify that no later information leaks into earlier states and to document how often status reverses.

Missingness. Zero-project issuer-years will be cross-tabulated by sector, size and filing count to distinguish "no disclosure" from "not processed"; the former is informative, the latter is a data gap, and the `coverage_status` field (complete for the defined scope, partial, failed, not observed) proposed for the issuer-year export will carry that distinction.

8.3. Acceptance gates

Table 10. Acceptance gates for a research-ready release and their status at revision `2324b58`

Gate	Requirement	Status
(a)	Target issuer universe and time window frozen and auditable	Met (snapshot design; file hashes pending)
(b)	Every published project links to at least one verifiable primary-source passage	Met at section level; character spans not stored
(c)	Duplicate, temporal and amount rules documented and tested	Documented; unit tests partial; no event or amount-type tables
(d)	Extraction and matching accuracy evaluated on independent human labels	Not met (protocol specified)
(e)	Issuer/form/year coverage and failures quantified	Met
(f)	Every empirical result regenerable from a pinned build	Met for this paper (`code/`, revision `2324b58`)
(g)	All coauthors have reviewed scientific claims and authorship	Pending

Until gates (c), (d) and (g) are met, the database is a *research database under validation*, and any use of its counts or amounts in empirical work should treat extraction and linkage error as measurement error of unknown magnitude.

9. Research Applications and Identification Limits

9.1. What the database can support

Four families of empirical designs are natural once the validation gates are passed; each is listed with the restriction it must respect.

Project counts and composition as firm-year variables. The issuer-year count of newly observed projects, by category, is a text-derived measure of disclosed strategic activity comparable in spirit to the thematic scores of the factor layer but with a discrete, interpretable unit. Because the panel is a snapshot universe, firm fixed effects and within-firm comparisons are safer than cross-sectional levels, and the later entry of smaller issuers (Figure 1) should be modeled explicitly.

Technology adoption and diffusion. The `technologies` field and the categories `ai_initiative`, `data_center`, `cloud_migration`, `automation` and `erp_implementation` allow the first-mention date of a technology to be located by issuer, sector and location, in the spirit of Kalyani et al. (2025) but from the issuer's own regulatory narrative. The small counts of these categories (a few hundred projects each) and the fact that technologies are free-text tags rather than a controlled vocabulary limit precision; a mapping of tags to a vocabulary is part of the proposed ontology work.

Event studies. Mentions in 8-K filings are dated to the day and are the closest the database comes to an announcement date (74,594 mentions). Because the primary document of an 8-K often refers to a press release filed as an exhibit that the pipeline does not read, the first *public* announcement may precede the first mention in the corpus; event dates should be verified against the exhibit or the newswire before use.

Project-level panels. For projects with several mentions, the timeline provides a sequence of status and amount statements that can be used to study announcement-to-completion durations, budget revisions and abandonment, subject to the caveats on status (last statement, not terminal fate) and on merged phases.

9.2. Identification limits

- **Disclosure, not activity.** The database measures what issuers chose to say in a regulated document. Disclosure thresholds, materiality judgments, boilerplate habits and litigation risk vary by firm, sector and time; a difference in project counts is a difference in disclosure before it is a difference in investment.
- **Snapshot universe.** Survivorship and backfilling are built in. Comparisons across years that mix cohorts of issuers confound activity with the composition of the observed panel.
- **Measurement error of unknown structure.** Until the human-coded evaluation is run, extraction precision and recall, category accuracy and linkage error are unknown. They are unlikely to be classical: boilerplate-heavy sectors and long MD&As are more exposed to false positives and to truncation, respectively.
- **Amounts.** Disclosed amounts are heterogeneous in type and scope and are never realized expenditure; the project-level maximum is a reading convenience, not an estimate.
- **Descriptive versus causal.** Associations between project disclosures and outcomes (returns, investment, employment) identify the joint determination of disclosure and outcomes, not the effect of the project. Causal designs require a source of variation in the *project* that is orthogonal to the decision to *describe* it, and the database does not supply one.

10. Limitations, Reproducibility and Conclusion

10.1. Limitations

Beyond the identification limits above, the current build has concrete technical limitations that bound its use. The issuer universe is a fund-holdings proxy at one date, not licensed Russell membership, and 65 of its issuers are outside the form scope. The parser stores section-level provenance but not character offsets into the raw HTML. The focus windows truncate long sections around lexical cues, so recall on annual reports is unknown and probably lower than on 8-Ks. The resolver is a deterministic key that both merges distinct phases and splits renamed initiatives, and the resulting project count is therefore an artifact of the key as much as a count of initiatives. Amounts carry no type and no realization flag. The 10-Q financial-statement and 8-K sections are read whole, while 10-K Items 3–6 and 8–9B are not read at all. And the ledger cannot yet distinguish an empty result from a failed call.

10.2. Reproducibility, ethics and terms

The corpus layer and the project layer are reproducible from the Git repository at revision `2324b58`, the frozen universe artifact, the download logs and the extraction ledger; the paid extraction step is resumable and idempotent, and the complete run costs roughly 775 USD at October 2026 prices. All numbers in this paper are regenerated by `code/export_stats.py` and `code/make_tables.py`. The raw filings are public documents retrieved under the SEC's fair-access policy with an identified research client; they are not redistributed, and the repository contains no credentials and no private infrastructure details. Derived datasets (project-level, mention-level, issuer-year and event-time exports) will be released with sample restrictions and a data dictionary once gates (c), (d) and (g) are met. The database does not contain personal data; the only names it stores are those of corporations, places and technologies as they appear in public filings.

10.3. Conclusion

We have documented a reproducible pipeline that turns 527,548 SEC filings of 2,564 issuers into 308,065 evidence-linked statements about 121,099 disclosed strategic initiatives, each traceable to an accession number and a filing section, classified into a 21-category taxonomy and organized into timelines. We have been equally explicit about what has *not* been done: the accuracy of extraction, classification and linkage has not been measured against independent human labels; amounts are not typed; events are not separated; the universe is a snapshot. The contribution of this paper is therefore the instrument and its audit rather than a validated measurement. The next step of the research program is the annotation study and the linkage audit described in Section 8; only their

results will determine whether the counts reported here can be used as variables, and with what measurement-error model. Until then, SPID is a research database under validation, and we ask that it be cited and used as such.

References

Christen, Peter, *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*, Berlin: Springer, 2012.

Cohen, Lauren, Christopher Malloy, and Quoc Nguyen, "Lazy Prices," *Journal of Finance*, 2020, *75* (3), 1371–1415.

Dyer, Travis, Mark Lang, and Lorien Stice-Lawrence, "The Evolution of 10-K Textual Disclosure: Evidence from Latent Dirichlet Allocation," *Journal of Accounting and Economics*, 2017, *64* (2–3), 221–245.

Fellegi, Ivan P. and Alan B. Sunter, "A Theory for Record Linkage," *Journal of the American Statistical Association*, 1969, *64* (328), 1183–1210.

FTSE Russell, "Russell US Equity Indexes: Construction and Methodology, v7.2," https://www.lseg.com/content/dam/ftse-russell/en_us/documents/ground-rules/russell-us-indexes-construction-and-methodology.pdf August 2026. Document dated August 2026; downloaded November 5, 2026.

Gentzkow, Matthew, Bryan Kelly, and Matt Taddy, "Text as Data," *Journal of Economic Literature*, 2019, *57* (3), 535–574.

Grimmer, Justin and Brandon M. Stewart, "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts," *Political Analysis*, 2013, *21* (3), 267–297.

Hassan, Tarek A., Stephan Hollander, Laurence van Lent, and Ahmed Tahoun, "Firm-Level Political Risk: Measurement and Effects," *Quarterly Journal of Economics*, 2019, *134* (4), 2135–2202.

Hoberg, Gerard and Gordon Phillips, "Text-Based Network Industries and Endogenous Product Differentiation," *Journal of Political Economy*, 2016, *124* (5), 1423–1465.

_ **and Vojislav Maksimovic**, "Redefining Financial Constraints: A Text-Based Analysis," *Review of Financial Studies*, 2015, *28* (5), 1312–1352.

Kalyani, Aakash, Nicholas Bloom, Marcela Carvalho, Tarek Hassan, Josh Lerner, and Ahmed Tahoun, "The Diffusion of New Technologies," *Quarterly Journal of Economics*, 2025, *140* (2), 1299–1365. Published version of NBER Working Paper 28999 (2021, DOI 10.3386/w28999).

Kelly, Bryan, Dimitris Papanikolaou, Amit Seru, and Matt Taddy, "Measuring Technological Innovation over the Long Run," *American Economic Review: Insights*, 2021, *3* (3), 303–320.

Li, Kai, Feng Mai, Rui Shen, and Xinyan Yan, "Measuring Corporate Culture Using Machine Learning," *Review of Financial Studies*, 2021, *34* (7), 3265–3315.

Loughran, Tim and Bill McDonald, "When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks," *Journal of Finance*, 2011, *66* (1), 35–65.

_ **and** _ , "Textual Analysis in Accounting and Finance: A Survey," *Journal of Accounting Research*, 2016, *54* (4), 1187–1230.

Raasveldt, Mark and Hannes Mühleisen, "DuckDB: An Embeddable Analytical Database," in "Proceedings of the 2019 International Conference on Management of Data (SIGMOD)" 2019, pp. 1981–1984.

Reimers, Nils and Iryna Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in "Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)" 2019, pp. 3982–3992.

Tandjambianda, Djerrycharli, Simon-Pierre Boucher, and Manel Kammoun, "SPID — SEC Project Intelligence Database: A Methodological Report on the S&P 500 Release," Unpublished internal methodological report, available from the authors, Université du Québec en Outaouais, Département des sciences administratives 2026. June 2026 study of revision `df7905d`; the report file was regenerated on October 28, 2026.

U.S. Securities and Exchange Commission, "Accessing EDGAR Data: Fair Access Policy," https://www.sec.gov/search-filings/edgar-search-assistance/accessing-edgar-data 2026. Accessed October 2026; URL re-verified November 5, 2026 (the former `/os/accessing-edgar-data` address redirects here).

Appendices

A. Taxonomy

Table 11. The 21-category SPID project taxonomy (`spid/taxonomy.py`, unchanged since the June 2026 release)

Code	Label
`manufacturing_expansion`	Manufacturing / Capacity Expansion
`plant_construction`	Plant / Site Construction
`data_center`	Data Center
`cloud_migration`	Cloud Migration
`ai_initiative`	AI / ML Initiative
`digital_transformation`	Digital Transformation
`erp_implementation`	ERP / Systems Implementation
`rd_program`	R&D Program
`product_launch`	New Product / Launch
`drug_pipeline`	Drug / Clinical Pipeline
`ma_integration`	M&A / Integration
`geographic_expansion`	Geographic Expansion
`capex_program`	Capital Expenditure Program
`supply_chain`	Supply Chain Initiative
`automation`	Automation / Robotics
`sustainability`	Sustainability / ESG
`energy_transition`	Energy Transition / Renewables
`infrastructure`	Infrastructure Investment
`cost_reduction`	Cost Reduction / Restructuring
`partnership`	Strategic Partnership
`technology_deployment`	Technology Deployment

B. Extraction contract

The system prompt below is passed verbatim to the chat-completion endpoint, with the 21 taxonomy codes and labels substituted at the placeholder. The user message is the string "`SECTION: <section_name>`" followed by the focused section text, truncated to 14,000 characters. Decoding parameters: temperature 0, JSON-object response format, 2,000 completion tokens. Model identifier recorded during the October 2026 run: `gpt-4o-mini-2024-07-18`.

```
You are a financial analyst extracting CORPORATE PROJECTS from SEC filing
text.
A ''project'' is any concrete initiative, investment, program, expansion,
construction,
product/R&D effort, acquisition/integration, technology deployment, partnership,
or
strategic program the company is pursuing.
Return STRICT JSON: {''projects'': [ ... ]}. Each project object has:
project_name : short descriptive name (use the official name if one is
stated)
project_type : EXACTLY ONE of the allowed types below
description : 1 sentence, grounded in the text (no speculation)
objective : the goal/why (or '''')
status : one of planned | in_progress | completed | mentioned
amount_usd : numeric USD figure if stated (e.g. 1500000000), else null
locations : array of places (city/state/country), else []
technologies : array of technologies involved, else []
partners : array of named partners/JV counterparties, else []
benefits : array of expected benefits, else []
risks : array of stated risks, else []
years : array of 4-digit years mentioned, else []
confidence : 0.0-1.0 (how clearly this is a real, concrete project)
Allowed project_type values:
- <code>: <label> (the 21 taxonomy entries of Table A1)
Rules:
- Extract ONLY real initiatives, not generic risk-factor boilerplate or
hypotheticals.
- If the text describes no concrete project, return {''projects'': []}.
- Do not invent facts. Numbers/locations must appear in the text.
- Prefer fewer high-quality projects over many vague ones.
```

The candidate gate is the case-insensitive regular expression `\b(project|initiative|program|invest|expan|?cent|automat|sustainab|renewable|restructur)\w*`. Sections without a match, or with fewer than 30 words, are not submitted.

C. Data dictionary and data-model audit

Table 12. Tables required by the research design, current implementation and status

Required table	Current implementation (database, table, key fields)	Status
`universe_snapshots`	`data/metadata/universe_*.json` + `companies` (CIK, ticker(s), name, sector, flags `in_sp500`/`in_russell3000`, fund weight, SIC, state, exchange, FYE, shard); snapshot date and source URL recorded	Partial: one file per month, no membership dates, no match confidence
`issuers`	`companies` (one row per CIK; former names from SEC header stored in metadata JSON only)	Partial: no name history, no parent/subsidiary links
`filings`	`filings` (accession PK, CIK FK, form, filing/report dates, SEC URL, local path, size, download and parse status)	Implemented; no acceptance timestamp, no checksum
`sections`	`filing_sections` (UUID PK, accession FK, name, item, order, text,	Implemented; no character offsets into raw HTML

The full field-level dictionary of the current tables is maintained in `docs/data_dictionary.md` of the working-paper folder and in the repository documentation.

D. Evidence register

Table 13. Evidence register: status of the main claims of this paper at revision `2324b58`

Claim	Status
Universe of 2,564 issuers built from the IWV holdings file (2026-10-23) and the S&P 500 cache (2026-05-25)	Implemented, verified (artifact counts)
527,548 primary documents of Forms 10-K/10-Q/8-K since 2010 downloaded; 16 unparsed	Implemented, verified (database counts)
Acquisition respected the SEC rate limit (aggregate token bucket; no 403 recorded)	Implemented; documented from run logs
1,342,283 sections, 10,720,890 chunks and embeddings; factor panels for 2,501 firms	Implemented, verified
1,123,098 sections submitted to the extractor; ledger free of outage artifacts	Implemented, verified after manual audit of the outage window
308,065 mentions, 121,099 projects, graph of 183,472 nodes	Implemented, verified
Each mention grounded in its source section	Documented (prompt rule), not mechanically verified
Extraction precision/recall, category accuracy, field accuracy, calibration	Not measured (protocol specified)
Linkage precision/recall of the resolution key	Not measured; failure modes documented with examples
Amounts typed and separated from realized expenditure	Not implemented (proposed)
Dated project events and point-in-time revision flags	Not implemented (mention-level timeline only)
Human-coded gold standard with two annotators and adjudication	Proposed
Author order, affiliations and CRediT contributions	Pending confirmation by the three coauthors

E. Additional tables

Table 14. Coverage by filing year

Year	Issuers	10-K	10-Q	8-K	Mentions	Issuers w/ mentions	Mentions per issuer	Projects first seen
2010	1,392	1,357	4,152	17,672	11,908	1,224	9.7	7,835
2011	1,424	1,409	4,260	18,533	13,551	1,287	10.5	6,679
2012	1,472	1,436	4,363	19,437	14,302	1,333	10.7	6,444
2013	1,539	1,483	4,530	19,773	14,611	1,408	10.4	6,286
2014	1,619	1,549	4,780	20,927	16,274	1,490	10.9	6,732
2015	1,689	1,635	4,983	22,077	17,073	1,535	11.1	6,675
2016	1,744	1,692	5,146	22,447	16,429	1,579	10.4	6,240
2017	1,810	1,745	5,359	22,897	17,350	1,636	10.6	6,490
2018	1,893	1,827	5,582	23,709	18,064	1,737	10.4	6,815
2019	1,973	1,903	5,829	24,256	18,737	1,806	10.4	6,837
2020	2,089	1,985	6,095	27,424	17,970	1,919	9.4	7,212
2021	2,247	2,116	6,606	27,944	21,312	2,077	10.3	8,216
2022	2,291	2,258	6,830	26,533	22,720	2,105	10.8	8,158
2023	2,329	2,295	6,938	26,851	22,955	2,129	10.8	7,977
2024	2,387	2,341	7,092	27,438	22,503	2,204	10.2	7,819
2025	2,453	2,406	7,300	28,716	23,220	2,280	10.2	7,990
2026	2,498	2,338	5,322	23,972	19,086	2,260	8.4	6,694

Table 15. Issuers with the most resolved projects

Ticker	Issuer	Resolved projects
UEC	Uranium Energy	240
UUUU	Energy Fuels	214
JBLU	Jetblue Airways	203
HAL	Halliburton	201
RIOT	Riot Platforms	195
EHC	Encompass Health	191
EBS	Emergent Biosolutions	187
MUX	Mcewen Inc	180
CDE	Coeur Mining	180
CLSK	Cleanspark	170
CWEN	Clearway Energy Inc Class C	168
ALK	Alaska Air Group Inc	168

Table 16. Self-reported confidence of mentions

Self-reported confidence (rounded)	Mentions
0.5	1,611
0.6	3,278
0.7	21,647
0.8	72,280
0.9	100,831
1.0	108,418

Table 17. Knowledge graph composition

	Type	Count
Edges	`owns`	121,099
Edges	`located_in`	54,583
Edges	`uses`	28,774
Nodes	`Project`	121,099
Nodes	`Partner`	36,822
Nodes	`Location`	14,508
Nodes	`Technology`	8,551
Nodes	`Company`	2,492

F. Replication

1. Clone the repository at revision `2324b58` and create the Python 3.12 environment (`requirements.txt`; DuckDB 2.0, pyarrow, lxml, sentence-transformers).
2. `python scripts/build_universe.py` rebuilds `data/metadata/universe.json`; the frozen artifact used here is kept with the paper (`data/`) so that a later IWV file does not change the sample.
3. `python scripts/download_filings.py -universe -rate 9` downloads the corpus (idempotent; about 255 GB; seven hours at two IP addresses). The cluster orchestration used for the October 2026 build is in `scripts/cluster/`.
4. `python scripts/build_database.py`, `parse_filings.py`, `build_embeddings.py`, `build_factors.py` build the corpus layer; `export_shard.py`/`merge_shards.py` are needed only for a distributed build.
5. In `project-db/`: `python -m spid pending` reports the sections to be extracted and the estimated cost; `python run_full_extract.py -workers 48` runs the extraction against the

ledger; `python -m spid resolve graph embed analytics` builds the project layer.

6. In the working-paper folder: `python code/export_stats.py <repo> data/` and `python code/make_tables.py` regenerate every table and figure; `latexmk` in `paper/` compiles the manuscript.