Disclosures

We are grateful for the computing resources provided by our universities, which were instrumental for pursuing this project. We declare that we have not received any grants, material help, financial incentives, or payments related to this project that could be construed as conflict of interest.

- Authors.

LLMs and Systematic Measurement Error

June 16, 2026

Preliminary Draft: Please Do Not Circulate

Abstract

Researchers increasingly construct text-based empirical measures with Large Language Models (LLMs), often treating model outputs as observed data rather than generated regressors. Using S&P 500 Item 1A risk disclosures from 2005–2024, we show that LLM-derived measures contain non-classical measurement error driven by two distortions: omission and injection. Omission is substantial: summaries retain only 40–63% of source vocabulary and selectively omit source-text language. Injection is also pervasive and not merely editorial: summaries introduce novel terms absent from the source that account for 16–42% of their most discriminative vocabulary, often changing salience, tone, and substantive content. Together, omission and injection directionally reshape narrative emphasis and predict outputs from downstream LLM tasks, including categorization, scoring, and ordinal classification. These distortions predict severity, specificity, complexity, and boilerplate measures, and generate firm-outcome associations consistent with non-classical measurement error. We document these patterns across models and textual constructs and propose diagnostics for structured LLM measurement error.

Keywords: Large Language Models, Text Analysis, Measurement, Corporate Disclosure, Generated Regressors, Risk Factors, Natural Language Processing, LLM Diagnostics

JEL Codes: C18, G14, G30, C81, M41

1 Introduction

Large language models (LLMs) have quickly become a standard tool for extracting quantitative measures from unstructured text. LLMs are adaptable to new problems and highly scalable to large tasks. They also often outperform traditional natural language processing approaches (e.g., Loughran and McDonald, 2011; Tetlock, 2007) in measuring sentiment, risk, readability, and narrative tone. These capabilities have led to a rapid incorporation of LLM-generated measures into empirical work.[1] While this progress is remarkable, scholars have predominantly treated LLM outputs as data; in reality, LLM outputs contain error because they are produced by statistical models, and the properties of this estimation error are largely unconsidered.

In this paper, we study LLM measurement errors within a standard finance corpus, Item 1A risk disclosures and earnings call transcripts. We formalize LLM-generated measures as a noisy proxy for the latent variable that a researcher is interested in, contaminated by a model-specific wedge that captures how the model compresses input, draws on its priors, and allocates attention. The wedge takes two distortion forms—a *calibration distortion* that rescales real effects, and a *realization distortion* that manufactures associations the source does not support—both of which violate the classical measurement-error assumptions that would justify treating LLM scores as unbiased. We make the wedge measurable through two summary-based proxies: *Omission* (source vocabulary the LLM drops) and *Injection* (terms the LLM adds from its priors). Applying this framework to four frontier models (GPT-4o-mini, Grok-3-mini, Phi-4, Llama-3.3-70B) on finance corpora, we show that the wedge proxies enter significantly in downstream regressions in ways that signal non-classical measurement error rather than attenuation bias.

Our primary diagnostic exploits how language in LLM summaries differs from the original documents. These differences reveal source content that LLMs ignore (omission) and terminology that LLMs introduce (injection), and let us construct empirical proxies for both. For each LLM, we prompt the model to summarize each document, identify the top 1,000 most distinctive terms (by lift) in the source and in the summary, and measure omission as the fraction of distinctive source terms dropped after summarization and injection as the fraction of distinctive summary terms

[1] LLM measurement applications are multifaceted. Just within corporate finance, LLMs have shed light on aspects of corporate disclosure, firm exposure to generative AI, earnings predictability, supply-chain risk, and ESG narratives, among many other topics (Kim et al., 2023; Eisfeldt et al., 2023; Lopez-Lira and Tang, 2023; Sikochi et al., 2024).

absent from the source. We complement these lexical proxies with two embedding-based diagnostics that surface the same compression through different lenses: a semantic fidelity measure (Coverage and Grounding, computed from sentence-level cosine alignment between source and summary in `all-mpnet-base-v2` embedding space) and per-token attention masses extracted at the moment each LLM emits a construct score, which expose the same attention bottleneck at scoring time as at summarization time.

LLM summaries are aggressive in both directions. Across the four models, summaries omit between 43% and 60% of the top-1,000 most distinctive source terms when pooled across all 26 risk categories, and the omission rises to 75–77% on the top-1000 most discriminative slots. Injection is concentrated in those same high-information slots: between 27% and 42% of the top-50 discriminative words in LLM summaries do not appear in the source text at all, reflecting vocabulary drawn from the models' training rather than firm-specific disclosure. The same patterns replicate qualitatively in the earnings-call corpus, confirming that the wedge is a property of LLM processing rather than an artifact of the legalistic Item 1A register.

The wedge is heterogeneous across the 26 risk categories of Fathollahi et al. (2026). Omission bites hardest in categories with templated, repetitive language — regulatory compliance, supply chain, competition & strategy, and litigation each see GPT-4o-mini omit roughly two-thirds of the top-1,000 distinctive terms — and bites least in categories with firm-specific operational and financial content (growth management, credit, liquidity, retaining roughly half). Temporal patterns also vary by model: Llama-3.3-70B and Phi-4 trend toward heavier Omission in the early part of our sample, while gpt-4o-mini and grok-3-mini are comparatively stable throughout.

Injected vocabulary differs systematically from human risk disclosures and follows patterns shaped by each model's training data. For Natural and Physical Environment risks, LLMs tend to introduce terms such as *droughts*, *wildfires*, *flooding*, and *radioactive*, while human disclosures emphasize *contamination*, *cleanup*, and *temperature changes*. For Political and Social Environment risks, LLMs often add *Taiwan*, *wars*, and *Chinese*, whereas firms emphasize *nationalization*, *hostilities*, and *customs*. The injected terms reflect what was salient in news discourse during training, not the enduring compliance categories firms use — models trained in 2023–2024 disproportionately weight Taiwan-related tensions and GDPR compliance for this reason.

These pervasive lexical patterns shape downstream LLM constructs. We prompt each LLM

to score four constructs — severity, specificity, complexity, and boilerplate — from each Item 1A disclosure, and find that Omission and Injection jointly predict each construct with strong significance across all four LLMs. Because Omission and Injection are proxies for *how* the LLM selected its vocabulary rather than for what the source said, a random text-processing model would not produce systematic associations between them and the constructs. Their strong joint significance is therefore a warning sign that the construct scores carry the wedge's footprint — and motivates the formal diagnostic that follows.

To diagnose whether LLM errors are non-classical, we estimate three specifications for each construct–outcome pair: a *naive* regression of the outcome on the LLM construct; a *joint* regression that adds Omission and Injection as wedge proxies (with construct-coefficient sensitivity $|\Delta\%|$ relative to naive); and a *decomposition* that splits the construct into a wedge-predictable component (coefficient θ) and a wedge-orthogonal residual (coefficient λ). Across 288 Item 1A cells (4 LLMs $\times$ 4 constructs $\times$ 6 outcomes $\times$ 3 fixed-effects variants), $|\Delta\%|$ exceeds 10% in roughly 63% of cells, and in 63% of the 83 cells that were significant under the naive specification. Severity, the construct most often used in prior work, loses naive significance under the joint specification in five of six firm outcomes; in the most economically important cells (net income, capital expenditure), the decomposition attributes the surviving predictive content to the wedge channel θ rather than the construct channel λ. Boilerplate is the most wedge-robust of the four constructs. The earnings-call corpus produces the same qualitative pattern with somewhat softer magnitudes ($|\Delta\%|$ exceeds 10% in 44% of 144 cells), confirming that the wedge is a property of the construct–corpus pairing rather than an Item 1A artifact.

If used without adjustment, LLM-derived measures import the model's own assumptions into academic work, overstating effects in categories with heavy term injection and compressing meaningful differences across firms. Our findings are not an argument against using LLMs — their scale and adaptability are substantial. They are an argument for treating LLM outputs as generated regressors with a measurable wedge. Simple practices should become routine: reporting Omission and Injection alongside main results, running the joint and decomposition specifications as a screen for non-classical measurement error, and aggregating across models to filter idiosyncratic wedge variation that no single model's proxies can see.

Our findings contribute to four strands of research. The first strand uses LLMs as a productivity

tool to scale textual analysis, treating outputs as reliable measures of firm fundamentals (Kim et al., 2023; Eisfeldt et al., 2023; Lopez-Lira and Tang, 2023; Jha et al., 2024; Bybee, 2024). Kim et al. (2023) use ChatGPT to summarize 10-K MD&A sections and treat the summaries as reliable extractions; Jha et al. (2024) construct firm-level LLM scores and validate them against CFO survey responses; Huang et al. (2023) build a finance-pretrained transformer (FinBERT) that outperforms dictionary methods on sentiment tasks. These papers demonstrate impressive explanatory power of LLMs while showcasing their adaptability to different settings, but none opens the black box to examine how the internal workings of LLMs contaminate inference. Our paper contributes by formalizing LLM outputs as generated regressors with a measurable wedge and providing diagnostics that screen for non-classical measurement error.

Our research joins a second strand that critiques textual methods on measurement grounds, documenting problems with bag-of-words approaches, dictionary saturation, negation handling, and context insensitivity (Loughran and McDonald, 2011; Gentzkow et al., 2019; Cohen et al., 2020; Grimmer et al., 2022). Although LLMs overcome some of these limitations, our findings imply they introduce new biases that raise concern because they look like semantic understanding. Our measurement of omission connects to the NLP summarization literature, where Grusky et al. (2018) introduce extractive coverage as a metric for how much of a summary's vocabulary derives from the source, and See et al. (2017) and Narayan et al. (2018) document that neural summarizers systematically discard source vocabulary. Our measurement of injection connects to the NLP faithfulness literature, where Maynez et al. (2020) formalize extrinsic hallucination as the introduction of content not supported by the source, and Ji et al. (2023) survey its pervasiveness across generation tasks. We extend this literature by showing these properties are severe in regulatory risk disclosures — a domain with a legalistic, hedged register that differs fundamentally from the news and Wikipedia corpora on which prior work focuses — and by translating the lexical-level findings into measurement-error consequences for finance regressions.

A third strand formalizes LLM outputs as generated regressors and develops inference tools for downstream regressions using AI-generated variables (Battaglia et al., 2024; Egami et al., 2023; Ludwig et al., 2025; Fong and Tyler, 2021). Battaglia et al. (2024) prove that regressions using AI-generated variables suffer from a bias that does not vanish in large samples, and propose bias-corrected confidence intervals using validation data. Egami et al. (2023) develop Design-based

Supervised Learning (DSL), a doubly-robust estimator that combines many LLM labels with a small gold-standard sample. Ludwig et al. (2025) provide an applied econometric framework distinguishing prediction from estimation uses of LLMs. Our contribution to this literature is mechanistic: we identify omission and injection as specific text-processing pathologies that produce the error structure Battaglia et al. and Egami et al. seek to correct. Our joint and decomposition diagnostic complements these corrective estimators: researchers can run it cheaply as a screening tool before investing in validation data, and it flags when deeper corrections (DSL, validation samples) are warranted.

Fourth, our paper contributes to recent research identifying problems with LLMs in predictive contexts. After the initial wave of research into LLM return prediction properties, research was quick to point out forward-looking bias (Ludwig et al., 2024; Sarkar and Vafa, 2024; Lopez-Lira et al., 2025). Sarkar and Vafa (2024) formalize lookahead contamination and show that prompt-based mitigations fail. He et al. (2025) train chronologically consistent LLMs (ChronoBERT, ChronoGPT) pretrained only on text available at each historical timestamp, and Engelberg et al. (2025) propose "entity neutering" when chronologically-consistent models are unavailable. Other research checks whether conclusions continue to hold after the model's training cutoff (Jha et al., 2024; Lopez-Lira and Tang, 2024; Bybee, 2025; Bhagwat et al., 2025). Our findings complement this literature by providing a cross-sectional perspective on how LLM text processing contaminates inference. Distinct from the temporal dimension, omission and injection produce non-classical measurement error that biases cross-sectional regressions regardless of training cutoff. Our research thus generalizes the forward-looking bias critique to a form of cross-sectional contamination, and by offering diagnostics and best practices, it provides a path forward for future research using these tools.

2 LLMs as Generated Regressors

Researchers often use LLM output as explanatory variables in regressions of the form

$$Y_i = \alpha + \beta \cdot L_{im} + \delta' X_i + \epsilon_i, \tag{1}$$

where L_{im} is an LLM-generated output, such as a score, classification, or extracted quantity. Standard practice treats L_{im} as if it were directly observed, reporting $\hat{\beta}$ with t-statistics computed as if the score were measured without error.[2] Implicitly, the goal of estimating Equation (1) is to estimate the relation between Y_i and a latent economic construct θ_i,

$$Y_i = \alpha + \beta\theta_i + \delta' X_i + u_i, \tag{2}$$

where L_{im} serves as a proxy for θ_i. Effectively, L_{im} is the output of a first-stage statistical model whose properties the researcher does not control or observe. In econometric terms, every LLM-derived measure is a generated regressor (Pagan, 1984; Murphy and Topel, 1985), close in spirit to the double machine learning setup of Chernozhukov et al. (2018), in which the first stage is a high-dimensional black-box predictor. Standard corrections assume the researcher either observes the first-stage estimator or can resample estimates cheaply. Neither condition holds for LLMs: the first stage conditions on a researcher-chosen prompt whose effect is not separately identified from the model, and resampling requires repeated API queries that are prohibitively costly at the scale of conventional panel-data.

We formalize the LLM-generated regressor as a stochastic mapping from documents and prompts to textual, scalar, or categorical outputs. Let T_i denote unit i's source text.[3] A model m with pretraining priors Ω_m, applied to T_i with prompt P_i, produces

$$L_{im} = \theta_i + w_m(T_i, P_i, \Omega_m). \tag{3}$$

The term $w_m(\cdot) = L_{im} - \theta_i$ is the measurement wedge, the part of the output that reflects model processing rather than the construct θ_i. By construction, the wedge is not a channel for additional signal about θ_i, and any covariation between $w_m(\cdot)$ and the outcome generates distortion. Combining the estimable Equation (1) with Equations (2) and (3) yields the disturbance $\epsilon_i = u_i - \beta w_m$. Thus, the wedge must be mean-independent of the structural disturbance u_i, the construct θ_i, and

[2] LLMs are also used to generate outcome variables, and the issues we develop apply in spirit to that case as well, though through a different mechanism. We focus on the case where the LLM output is used, directly or indirectly, as an explanatory variable, since it is the most common in current empirical work and the bias mechanism is most transparent.

[3] The unit of analysis is general; i may index firms, filings, transcripts, contracts, or any text-bearing entity.

covariates X_i: $E[w_m \mid \theta_i, X_i, u_i] = 0$. Under this condition L_{im} is an unbiased noisy proxy for θ_i and the classical measurement-error case obtains: the wedge induces only attenuation, and standard errors-in-variables corrections recover β in expectation.

The classical case breaks down when the wedge is correlated with θ_i or u_i conditional on X_i. The wedge may be correlated with θ_i if the LLM misreads the construct in a way that depends on the level of θ_i, such as the model compressing a rating scale by pulling genuinely extreme disclosures toward the middle. This is a systematic miscalibration, rather than carrying additional signal, and it rescales β, distorting the magnitude of the relationship. The more serious failure occurs when $Cov(w_m, u_i) \neq 0$, indicating that w_m affects Y_i through unobserved variables other than through θ_i.

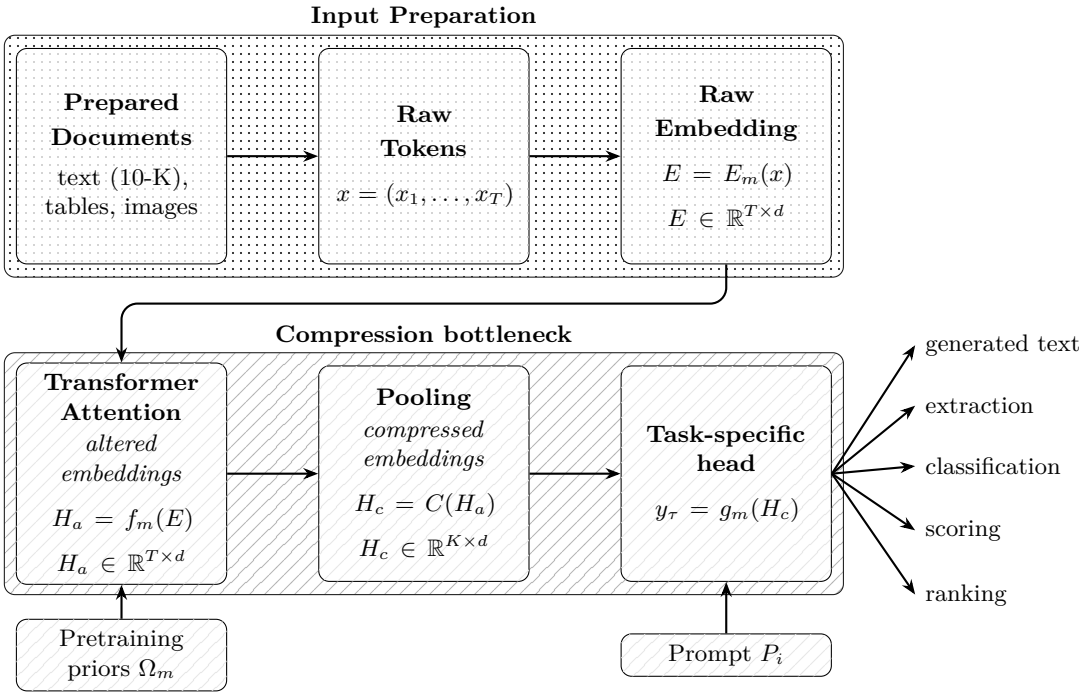
Non-classical measurement error from both sources of correlation arises through the three arguments of $w_m(\cdot)$, the text, the prompt, and the model's priors. First, the wedge may vary with features of the source text T_i, such as document length, linguistic register, or boilerplate and topical content, which bias the estimate if these features capture multiple concepts that are correlated with θ_i or with Y_i for reasons other than θ_i. Second, the wedge may vary with the prompt P_i in ways the researcher does not realize, such as ordering effects, how the prompt defines the LLM-generated regressor, or implicit anchors in the prompt language. Third, it may reflect the model's priors Ω_m, which over- or under-weight certain content, so that disclosures whose register matches the pretraining distribution are processed differently from those whose register does not.

2.1 LLMs and Compression

All three channels of measurement error stem from a single feature of how language models process text: they compress information. The model transforms input text into a smaller output, and that reduction is selective. Processing a long disclosure into a single score or category forces the model to discard most of what the text contains, and what it keeps is not chosen randomly.

Consider a disclosure stating: "Our operations depend on a limited number of suppliers for certain key components, and we do not have long-term supply agreements with most of these suppliers. A transition to new suppliers could result in delays, quality issues, and increased costs that could materially affect our results of operations." A human reader retains the specifics: limited suppliers, no long-term agreements, transition risks enumerated as delays, quality, and cost, with the impact qualified to operations. A model summarizing the passage may return only: "The company

Exhibit 1: LLM compression as structured measurement error



faces supply chain risks that could negatively impact financial performance." The conditional structure, the absence of contracts, and the enumerated consequences are compressed away.

Exhibit 1 illustrates the compression process through the LLM-based measurement pipeline that transforms the prompt and source material into a compressed task output. A document of text, tables, and images is first converted into tokens $x = (x_1, \ldots, x_T)$, and then into embedded token representations $E = E_m(x) \in \mathbb{R}^{T \times d}$. The tokenization and embedding process does not discard any information and is best understood as representation rather than measurement. That is, these stages render source content as numerical objects the model can process, but the LLM has not yet decided which parts of the text are economically important.

That selection begins in the transformer layers, which build contextualized token representations $H_a = f_m(E) \in \mathbb{R}^{T \times d}$. Attention does not reduce the T tokens to a smaller set of K summary vectors, so the representation is still token-level, but it is the first compression-like stage of the process. Each token's representation becomes a weighted mixture of information from the others, so the model is in effect deciding which parts of a long document matter for representing any given part, and tokens that receive little attention have less opportunity to shape what follows. The same layers also fold in content the model learned during pretraining, so the representation reflects

its priors as well as the source. Attention is therefore an informational compression stage and the entry point for model priors Ω_m and prompt-driven emphasis P_i, even though the size of the representation has not yet fallen.

The more literal bottleneck occurs at the pooling stage. After attention has rewritten each token in light of the broader document, pooling discards much of the per-token detail and produces a smaller object, $H_c = C(H_a) \in \mathbb{R}^{K \times d}$, with $K \ll T$ and often $K = 1$. By analogy, attention rewrites every card in a long deck by letting each card borrow information from the others, and pooling then writes a short summary of the deck and throws the individual cards away. This is where the size drops, and it is the step that decides which of the rewritten content survives into the compact object the task-specific head can use. The head, $y_\tau = g_m(H_c)$, then maps that representation into the requested output, generated text, an extracted field, a category, a score, or a ranking, which the figure orders from least to most compressed.

The central measurement problem is that the reported output is not produced from a transparent, exhaustive read of the full disclosure. It is produced from whatever survives attention, pooling, and the task-specific projection. As a result, LLM-generated measures carry structured rather than merely random measurement error whenever economically relevant content is de-emphasized, averaged away, or replaced by prompt- or model-driven judgment. Importantly, because every task reads the same compressed representation H_c, a distortion in H_c propagates to all outputs: the wedge that contaminates a score also shapes the category or summary the model would produce from the same input.

Compression depends on the chosen LLM and prompt design, not the properties of the source text. A prompt may ask an LLM to return a summary, a coarse category label, an ordinal rating, or a single score on risk severity, all based on the same source text. These task-specific outputs sit on a spectrum from least to most compressed, and they face a clear tradeoff. Richer outputs give the LLM more room to extract signals and audit its output, but they also provide more surface on which to editorialize, drift, or import content that the source text does not support. More compressed outputs reduce that surface, but they force the model to collapse a multidimensional risk profile into a score or rating. The source text is usually too nuanced for a single score to capture its content accurately and is often consistent with a range of values. The model thus settles on a point estimate by leaning heavily on its priors, which reflects its training as much as the filing content.

Exhibit 2: LLM Compression Grid: Signal vs. Distortion

	Intrinsic (from source text)	**Extrinsic From the model's parameters**
Signal Extraction	**Faithful extraction.** Reads mild single-supplier language as the high concentration risk it is, which a keyword measure misses.	**Useful context.** Knows the firm is a copper miner in a volatile market and scores vague price-exposure language higher, as an analyst would.
Distortion A: Calibration	**Boilerplate inflation.** Scores severity up on generic stock language ("could materially and adversely affect"), which carries no information.	**Stale-prior conflation.** Shades pandemic risk as systematic from a training-era prior, regardless of plainly idiosyncratic text.
Distortion B: Realization	**Structural misread.** Treats a hypothetical ("if a key customer were to leave") as a realized exposure.	**Look-ahead leakage.** Scores a 2006 mortgage-exposure disclosure as catastrophic because it knows 2008 happened.

Exhibit 2 organizes the effects of compression along two axes. The first is whether the LLM draws its output intrinsically from the source text or extrinsically from the model's parameters; the second is whether that output is signal or distortion. Distortion takes two forms. Distortion A is a calibration error, where the model assigns the wrong weight to content in the text or to its own prior. For example, the LLM may over-weight generic 10-K boilerplate or apply stale priors to a topic. Because the wedge then correlates with θ_i, violating $\text{Cov}(w_m, \theta_i) = 0$, calibration error rescales a real effect (β) but it does not fabricate one. Distortion B is a realization error, where the model reports content the text does not support, as when it reads a hypothetical risk as a realized adverse event. Because such disclosures tend to precede realized risks, the wedge correlates with u_i, violating $\text{Cov}(w_m, u_i) = 0$. Thus, realization error manufactures an association.

3 Empirical Analysis

3.1 Data

Our primary corpus consists of Item 1A risk factor disclosures extracted from annual 10-K filings of S&P 500 firms over 2005–2024, obtained from SEC EDGAR. We parse each firm's risk disclosure section into individual disclosures, treating each risk factor paragraph as the unit of analysis, following standard practice in the empirical disclosure literature (Campbell et al., 2014; Kravet and Muslu, 2013; Hope et al., 2016). The resulting sample contains 889,223 unique risk factor disclosures spanning 1,675 firms. We lemmatize all tokens, remove standard stopwords, and retain only terms

with a minimum corpus frequency of fifteen. We assign each disclosure to one of twenty-six risk categories (with 5 parent categories) using the taxonomy developed in Fathollahi et al. (2026) and presented in Appendix Table A.1. On average, Item 1A disclosures contain 11,575 words (median is 9,740), though length varies substantially across risk categories: operational and regulatory disclosures tend to be longer and more technically specific than financial risk disclosures.

Our second corpus consists of quarterly earnings call transcripts for S&P 500 firms over the same sample period, obtained from Refinitiv Workspace. Each transcript is structured into a prepared management section and an analyst question-and-answer session. Relative to Item 1A filings, earnings calls contain more timely language and, in the Q&A session, unscripted responses produced under real-time constraints. The earnings call corpus contains 97,776 transcripts spanning 1,563 unique firms, with an average of 9,781 words per transcript. We apply the same preprocessing pipeline as for the Item 1A corpus. Together, the two corpora represent complementary modes of firm communication. Item 1A disclosures are legally scrutinized, drafted annually by counsel and management, and designed to satisfy regulatory requirements. Earnings call transcripts are produced quarterly and subject to minimal advance preparation in the Q&A session.

We extract company financial information from Compustat yearly files for 10-K Item 1A corpus and the quarterly file for the earnings call transcripts. Stock prices come from CRSP. And finally, we borrow the word lists from Loughran and McDonald (2011) to measure tone in raw corpora and their corresponding summaries.

3.2 Measuring Omission and Injection from Generated Descriptions

Since θ_i is latent, the wedge w_{im} is not directly observable. We can, however, observe features of the LLM's processing through patterns in its output. We focus on the summarization task because it gives a direct comparison between source and output that is informative about the model's priors: we can ask which source vocabulary the model retained, which it omitted, and which terms it introduced from outside the source. The summarization task is the least compressed output task type (see Exhibit 1) and also reflects general properties of how the model transforms the same input into any output, including scores and classifications. The summarization task is our laboratory, but the patterns it reveals generalize to other tasks as they undergo the same LLM compression process presented in Exhibit 1.

For any source-summary pair, we partition the source vocabulary and the summary vocabulary into three disjoint sets: retained, omitted, and injected terms. Retained terms appear in both the source text and the generated summaries; omitted terms appear only in the source text, and injected terms appear in the summary but not in the source. Omission and injection are mechanical properties of LLM text processing, and they correspond to well-studied concepts in natural language processing.

Omission is related to the NLP concept of extractive coverage (Grusky et al., 2018). It sits on the intrinsic axis of Exhibit 2 and refers to the source-text vocabulary dropped during LLM summarization. Omission may improve signal extraction efficiency when it sheds redundant or non-discriminating language (i.e., noise). However, omission can distort the signal when it selectively drops or overweights non-redundant information from the source text that is relevant for θ_i or Y_i.

Injection sits on the extrinsic axis of Exhibit 2, adding terminology absent from the source-text. Injection improves signal when added terms correctly contextualize disclosure content that is supported by the source-text, but comes from the models priors. Injection adds to distortion when its priors cause it to reflect stale associations or knowledge of later outcomes that the source does not support.

To measure omission and injection rates for a given document, we compare the most discriminating vocabulary in the LLM summary against that in the original text, restricting attention to meaningful words so that vocabulary differences are not trivial. For each disclosure i and model m, we identify the top-n most discriminatory words in the source text and the top-n in the LLM's summary, sorting each by *lift*. The omission rate O_{im} is the share of source's top-n discriminating words that do not appear in the summary's top-n:

$$O_{im} = 1 - \frac{|V^{(n)}_{i,\text{source}} \cap V^{(n)}_{im,\text{summary}}|}{|V^{(n)}_{i,\text{source}}|}. \tag{4}$$

The injection rate I_{im} is the share of the summary's top-n discriminating words that do not appear anywhere in the source:

$$I_{im} = 1 - \frac{|V^{(n)}_{im,\text{summary}} \cap V_{i,\text{source}}|}{|V^{(n)}_{im,\text{summary}}|}. \tag{5}$$

We prompt four models (GPT-4o-mini, Grok-3-mini, Phi-4, and Llama-3.3-70B) to produce

summaries, keywords, and category assignments across the twenty-six risk categories listed in Table A.1. We compare LLM outputs to source text using a battery of vocabulary-sensitive metrics: token count, vocabulary size, concentration (proportion of top-100, top-500, and top-1000 words), normalized Shannon entropy, and the omission and injection rates defined above. Our analysis focuses on the top 1,000 discriminative terms that most distinguish categories according to lift (minimum corpus frequency of 15, lemmatized).

LLM Compression at Document-level

We report the aggregate summary statistics for four LLMs that summarize the Item 1A risk disclosure text in panel A of Table 1 and the statistics for two LLMs that summarize earnings call summary transcripts in panel B.[4] This summary provides our initial exploration of LLM omission effects vis-a-vis the lens of vocabulary omission and injection. Token counts fall sharply when moving from full-text disclosures to LLM summaries, but the degree of omission varies strongly across models. Phi-4 and Llama-3.3-70B compress aggressively, reducing token counts by approximately 92% (from 115–118M to 8.7–9.8M). GPT-4o-mini compresses by 86% (from 103M to 14.8M). Grok-3-mini reduces tokens by only 78% (from 126M to 27.2M).

Vocabulary shrinks correspondingly. Full disclosures contain 36,904–41,879 unique words; LLM summaries retain only 14,965–24,909, with Grok again the most expansive and Llama-3.3-70B the most restrictive model. Concentration gives us a glimpse of the LLM text compression process illustrated in Exhibit 1. In full disclosures, the top-100 words account for approximately 29% of all tokens; in LLM summaries, this figure rises to 34–41%. Top-500 concentration climbs from roughly 62% to 66–74%. LLMs appear to funnel diverse, firm-specific language into a narrower vocabulary where common words dominate.

Keywords display the most extreme concentration. The top-100 keywords account for 43–60% of all keyword tokens, and the top-500 account for 77–89%. Llama-3.3-70B's keywords are especially concentrated: just 4,404 unique terms, with the top-100 claiming 60% of tokens and the top-500 claiming 88%. When LLMs distill risk disclosures into salient terms, they converge on a remarkably small lexicon. Entropy remains relatively stable across sources (0.71–0.73), but concentration reveals

[4]We currently limit ourselves to gpt-4o-mini and llama-3.3-70B for all earnings call transcript analyses because we find LLMs omission and injection patterns are similar to those we find in Item 1A text.

an asymmetry: LLMs retain high-frequency vocabulary while omitting the long tail of distinctive, low-frequency terms.

Panel B shows that the unique vocabulary size in earnings call transcripts is about 3.5 times more than that of Item 1A disclosures. This varied vocabulary is expected because at least the Q&A part of the earnings calls are not scripted and therefore participants are likely to use varied language. Nevertheless, both GPT-4o-mini and Llama-3.3-70B reduce the number of tokens by 85% when they summarize the transcripts.

Beyond the omission diagnostics, a contextual analysis of the terms used in source dictionaries versus LLM summaries reveals how summarization is not neutral. Table 2 highlights how summaries differ from source texts contextually for several categories. For example, when humans use keywords like 'temperatures' or 'pollution' under natural and physical environment risk disclosures, LLM summaries describe these terms with injected vocabulary like 'droughts,' 'wildfires', 'toxic' and 'discharges.' Other risk factors exhibit similar subtle changes to meaning via LLM summarization.

Omission and Injection by Risk Factor

Figure 1 disaggregates omission and injection by major risk categories in our sample.[5] Following equations 4 and 5 above, Omission (O_{im}) measures the fraction of discriminative terms that drop out of the top 1000 after summarization, and Injection (I_{im}) measures the share of LLM vocabulary absent from the source. High Omission indicates greater vocabulary reduction. High Injection indicates more novel terminology introduced by the LLM.

Omission varies substantially across models. Llama-3.3-70B omits most aggressively (73.3% across risk categories), followed by Phi-4 (70.6%), then GPT-4o-mini (66.9%). Grok-3-mini omits the least (58.8%), consistent with its expansive style. At the category level, omission ranges from 57% to 74% depending on model and risk type, with disclosures related to financial risk facing about 1-2% more omission than other risk disclosures.

Injection and omission do not exhibit a mechanical negative relation. Grok omits the least and is the second highest in terms of injection (29.5% of top-1000 discriminative terms absent from the source text), whereas GPT-4o-mini injects the highest (40% of terms absent from the source

[5]Please see Figures A.1 and A.2 in the appendix for a more granular view of omission and injection rates across our full taxonomy of risk factors.

text). Llama-3.3-70B and Phi-4 inject at 20% and 17.4%, respectively. The two dimensions are conceptually interdependent but in practice are model-specific and operate quasi-independently, a point which we will demonstrate in section 3.5.

Category-level injection rates reveal where models struggle most. For GPT-4o-mini, Regulatory and Business Model-related disclosures, which fall under Regulatory and Strategic risk categories, elicit 44% injection each, meaning nearly half of its top discriminative terms for these categories do not appear in the source disclosures. Phi-4's worst category is Competition Strategy (24%), meaning about one-third of its discriminative vocabulary is not in the original source. Llama-3.3-70B also struggles with Organizational Development (29%). Categories involving operational processes, physical systems, and regulatory language consistently elicit high injection across models. These domains require specialized vocabulary that firms describe idiosyncratically. Models fill the gaps with prototypical terminology from pretraining.

Other categories prove more tractable. Macroeconomic Environment shows injection rates of 18-38% across models, with Llama-3.3-70B injecting just 18%. Credit ranges from 14-35%, Financial Markets from 18-38%. These categories feature standardized financial and legal vocabulary that models extract without supplementation. The pattern is consistent: established disclosure conventions permit faithful extraction, while operational or technical language forces models to inject.

This heterogeneity varies across models in ways that matter for research design. A study using GPT-4o-mini to analyze Regulations would face 44.3% injection; switching to Llama-3.3-70B would reduce this to 23%. While GPT-4o-mini injects the most terms in our setup, the pattern switches if we also take Omission into consideration. No model uniformly dominates. The optimal choice depends on which risk categories matter for the research question.

The nature of the injection also matters. Models systematically rewrite hedged legal language ("may have a material adverse effect") into causal assertions ("leads to," "faces," "poses significant threats") that never appeared in the original filings. The summary often reads more decisively than the source.

3.3 Attention Weights in Open Weight Models

To better understand the nature of how LLMs compress their inputs, we analyze how attention weights are allocated to different parts of the input text. We do this using the open-weight models Llama-3.3-70B and Phi-4, which give access to the model's internal states. For every Item 1A disclosure chunk in our sample, we feed the chunk to the model under our scoring prompt and capture the attention distribution at the token where the model emits each construct's continuous score, averaged across all transformer layers and attention heads. We partition the input into a first-10%, middle-80%, and last-10% region and compute the per-token attention mass each region receives, the share of total attention divided by the share of input tokens, so that 1.0× corresponds to uniform allocation.

As is illustrated in Figure 2, both models allocate disproportionately more attention per token to the first 10% of the input than to the middle 80% or the last 10%. Llama-3.3-70B places 8.6–9.0× uniform attention on the first 10% across all four constructs, while attending the disclosure body (middle 80%) at only 0.1–0.3× uniform, an order of magnitude below each token's fair share. Phi-4 shows the same pattern at lower magnitude (6.4–7.0× on the first 10%, 0.1–0.3× on the middle 80%). The pattern is nearly identical across the four constructs (severity, specificity, complexity, and boilerplate). If attention reflected each construct's semantic content, it would produce distinct distributions. This similarity across constructs suggests that this attention pattern reflects how the model attends to inputs.

The asymmetric attention pattern in Figure 2 is robust to three natural alternative explanations. First, it does not reflect the length of the input. By subdividing the source text differently, we construct shorter and longer variants of each chunk and find that the first-10% attention mass is essentially unchanged across input sizes. Second, the pattern is not driven by the position of the user's instructions within the prompt. Swapping the score instructions from the front to the back of the input, so that the disclosure text now occupies the opening tokens, leaves the first-10% attention mass essentially unchanged on both models. The bottleneck follows token position, not the location of the instruction text. Third, the pattern does not depend on the somewhat arbitrary 10% boundary. Re-computing the per-token concentration multiplier with boundaries ranging from 5% to 33% reveals an even sharper concentration at narrower cutoffs: each token in the first 5% of

the input carries 17× uniform attention on Llama-3.3-70B and 12× on Phi-4.[6]

3.4 Fidelity Loss during Text Summarization

The Omission and Injection rates in Section 3.2 measure the two channels of wedge distortion through differences in the vocabulary sets. While the vocabulary-based measures of distortion are easy to compute, they do not cover the full picture. For example, when a model paraphrases "key customer" as "important client", it could be considered as both omission and injection even though the meaning was preserved. For measurement error, we care whether LLMs faithfully summarize the source content rather than distort meaning. Therefore, we construct a semantic counterpart to the lexical decomposition in embedding space, where sentence-transformer cosines serve as a scalable proxy that captures contextual similarity beyond vocabulary.

For each disclosure-summary pair we encode every sentence with the `all-mpnet-base-v2` sentence transformer (Reimers and Gurevych, 2019) and adapt the cosine-matched precision/recall construction of Zhang et al. (2020) from the token level to the sentence level. *Coverage* is recall-style: for each source sentence, take its best cosine match to any summary sentence, then average across source sentences. *Grounding* is precision-style: for each summary sentence, take its best cosine match to any source sentence, then average across summary sentences. Coverage asks how much of what the source said the summary kept and grounding asks how much of what the summary said came from the source. A summary that drops content but rewords what it keeps will have low coverage and high grounding; a summary that retains content but embellishes with model-driven additions will have high coverage and low grounding. The grounding channel operationalizes the faithfulness/factual-consistency construct in the summarization literature (Laban et al., 2022). Figure 3 plots 1−Coverage and 1−Grounding on opposing half-axes so that a perfectly faithful summary registers as zero on both sides; a positive bar to the left is source meaning the summary failed to carry over, and a positive bar to the right is summary content the source does not anchor, making them the semantic counterparts of Omission and Injection.[7]

Total semantic drift, the sum of the two deficits, varies across the four LLMs (from 0.49 in Phi-4 to 0.53 in GPT-4o-mini). The substantive heterogeneity is in the channel mix, and it tracks

[6] See Appendix Figure A.5, panels (a), (b), and (c).

[7] Note that injection can be either the Calibration or Realization as noted in Exhibit 2. We are presently unable to distinguish between the two types.

the model-specific lexical patterns documented in Section 3.2. Phi-4 and Llama-3.3-70B, the two heaviest token compressors, lose meaning predominantly through the coverage channel: 1−Coverage of 0.300 and 0.319 against 1−Grounding of only 0.186 and 0.198. They drop a substantial fraction of source meaning but what they retain stays anchored to the source. GPT-4o-mini and Grok-3-mini show a more symmetric two-sided profile: GPT's deficits are 0.278 and 0.250; Grok's are 0.264 and 0.229. They omit less of the source than the heavy compressors but introduce more summary content that does not map back to anything in the disclosure. The semantic ordering of coverage loss (Llama > Phi-4 > GPT > Grok) reproduces the lexical ordering of Omission rates (73.3%, 70.6%, 66.9%, 58.8%), and the semantic ordering of grounding loss (GPT > Grok > Llama > Phi-4) reproduces the lexical ordering of Injection rates (40%, 29.5%, 20%, 17.4%).[8] The semantic and lexical measurements read the same model behavior on different scales.

The two channels map directly onto the wedge axes of Exhibit 2. Coverage loss is intrinsic-axis distortion: source content the model failed to carry into its output. Grounding loss is extrinsic-axis distortion: output content drawn from the model's priors Ω_m rather than from the source. The vocabulary set and embedding similarity methodologies show what falls out of a summary and what gets added to it are separate, model-specific channels of wedge distortion, and that both modes are present in every LLM we examine. The implication for measurement error is that an LLM-derived measure inherits the intrinsic-axis bias of the model's coverage profile and the extrinsic-axis bias of its grounding profile, and the two combine in different proportions across models.

3.5 Drift in Tone and Semantic Similarity

We showed that LLM summaries distort source meaning and split that distortion into intrinsic and extrinsic channels. In this context, two questions arise naturally. First, does the distortion have a direction? An LLM that consistently raises the density of some tonal categories and lowers others is doing something different from one that adds mean-zero noise: the former is an instance of Calibration distortion in Exhibit 2 and can bias downstream measures that depend on tonal density, while the latter only attenuates. Second, are the per-LLM wedge levels documented in Section 3.4 stable across the sample period, or do they drift with filing year? Figure 4 addresses both questions.

[8] Correlations at the disclosure level are 0.64 for 1-Coverage and Omission and 0.42 for 1-Grounding and Injection (within-model ranging 0.53-0.74 and 0.33-0.48 respectively).

In panel A, we characterize the tonal direction of summarization as density of six word lists (positive, negative, uncertainty, litigious, strong-modal, weak-modal) provided by Loughran and McDonald (2011) (LM) and Brysbaert et al. (2014) concreteness score. These densities are measured as a share of *raw* tokens in each source disclosure and its LLM summary. We standardize each summary's density against the distribution of source densities, so $+1\sigma$ on a given spoke means the summary loads that tonal dimension at one source-standard-deviation above the typical source disclosure; spokes pushed outside the source ring indicate amplification, spokes pulled inside indicate dilution. The radar makes three patterns visible. First, every LLM raises the per-token density of negative, uncertainty, and weak-modal language above the source baseline, with uncertainty and weak-modal the largest amplifications across all four models. This is a concentration effect rather than an addition effect: absolute counts of LM-category words fall in every summary relative to its source (see Appendix Figure A.6), and the summaries cut non-tonal vocabulary, such as boilerplate phrasing, named entities, dates, transitional language, at a higher rate than they cut tonal vocabulary. So the share of tokens that fall in the negative, uncertainty, and weak-modal categories rises. Second, positive tone moves in the opposite direction: three of four LLMs (GPT-4o-mini, Phi-4, Llama-3.3-70B) sit visibly inside the source baseline on positive density and Grok-3-mini sits at baseline; no LLM amplifies positive. Litigious and strong-modal language sit roughly at baseline for all four LLMs, and concreteness is amplified by Phi-4 alone. Third, the magnitude of total tonal drift differs across LLMs: Phi-4 has the largest aggregate departure from the source ring and Grok-3-mini the smallest. This magnitude ordering does not mirror the magnitude ordering of total semantic drift from Section 3.4 (Llama > Phi-4 > GPT > Grok), so tonal density and sentence-level meaning loss are correlated but not interchangeable channels of summarization wedge.

In panel B, we plot three time series over the 2005–2024 filing window for each LLM: the document-level MPNet cosine similarity from Section 3.4 (dark grey), the lexical Omission rate (red), and the lexical Injection rate (blue). Each series is z-scored within (LLM $\times$ metric) so that within-LLM year-to-year movement is visible on a common scale, and shaded bands are 95% confidence intervals on per-year means. We find that every panel shows visible year-to-year variation, and that variation is largest in the early years of the sample (2005–2008). The confidence bands are also widest in those years, which partly reflects the fact that the Item 1A filing requirement began with 2005 fiscal-year and the number of firms filing grew from there on for the next several years.

We also find that the direction of within-LLM co-movement among the three series is model-specific. For GPT-4o-mini, cosine and Omission both decline across the sample window while Injection rises; for Grok-3-mini, cosine and Omission move inversely; for Phi-4, cosine rises and Omission declines; Llama-3.3-70B shows the smallest within-window movement of the four. The temporal patterns are model-specific in both magnitude and direction and do not collapse to a single shared trend.

Both panels carry implications for measurement error, though the strength of the implication differs. Panel A's tonal redirection is an instance of Calibration distortion (Distortion A of Exhibit 2) at the summarization stage: the per-token tonal weight a disclosure carries is shifted in a direction common to all four LLMs with different magnitudes across models. The tendency of LLMs to drop non-tonal vocabulary faster than tonal vocabulary implies that, for example, per-token hedge density rises even though absolute hedge counts fall. Therefore, any downstream LLM measure that reads its input token-by-token and depends on per-token tonal density will inherit a denser hedged-and-negative text than the source provided. Whether that inherited shift translates into measurable construct bias for the four LLM-generated constructs we study (severity, complexity, specificity, boilerplate) is a separate question that we take up in Section 4. Panel B's contribution is descriptive: each LLM exhibits its own time pattern in cosine, Omission, and Injection across the sample window, and the patterns do not align across models. We do not read this as proof that the wedge is non-stationary in a way that biases cross-period regressions, but the visible LLM-specific time variation is a reason to check temporal stability of any LLM-derived measure before relying on it in a panel regression whose window spans the early years of the sample.

3.6 Cross-model Agreement: Classification, Priors, & Discriminative Vocabulary

We now turn to the question how LLM distortion propagates into the downstream tasks that researchers typically use LLMs for: text generation, classification, and scoring. The wedge, $w_m(T_i, P_i, \Omega_m)$, defined in Section 2 helps us address this question: if two LLMs produce different outputs on the same input, the difference must come from Ω_m. We begin with classification and text generation and report the results of six cross-model agreement tests in Figure 5.

Panel A measures the categorical label each LLM assigns when classifying the same S&P 500 Item 1A disclosure into one of 26 risk-factor categories: pairwise agreement ranges from 0.53 to 0.63, so two LLMs chosen at random assign the same category for, on average, 57% of disclosures

and disagree on the remaining 43%. Given that the input text (T_i) and prompt (P_i) are fixed, the disagreement can only come from model-specific processing (Ω_m) of the same input. Panel B asks each LLM to generate a discriminative-term dictionary for each of the 26 risk factor categories using only the category label and a short definition, with no source disclosure provided: pairwise Jaccard ranges from 0.04 to 0.12 with mean of 0.07, lower than the lower edge of every other panel. With no source text to anchor them, the four LLMs share essentially no discriminative vocabulary. The practical implication is that an LLM-generated dictionary is likely to reflect the priors of the LLM about the topic requested. The low numbers suggest that the models have different starting priors (without anchoring source text) and therefore are likely to have higher levels of disagreement among themselves.

The middle row applies TF-IDF lift to the raw Item 1A sentences each LLM grouped into a risk category, then measures Jaccard agreement on the resulting discriminative-term sets across the four LLMs (averaged over the 26 categories). Mean Jaccard is 0.70 on the broad lift-positive set (Panel C) and 0.28 on the top-50 tail (Panel D). The high broad-lexicon agreement in Panel C is a reflection of the fact that the underlying corpus is shared and human-written — the words being TF-IDF'd come from the same Item 1A vocabulary regardless of which LLM does the partitioning. The 50-60% lower top-50 agreement in Panel D is partly because the set of most discriminative terms across models are influenced by Ω_m-driven differences that led to an LLM assigning a human written sentence into a risk category. Therefore, the LLM priors produce sharply different vocabulary signatures even though the underlying sentence pool is the same human text.

The bottom row runs the same TF-IDF zoom as the middle row but on a different substrate — each LLM's own generated risk descriptions instead of human Item 1A sentences — giving mean Jaccard of 0.50 (Panel E) and 0.22 (Panel F). The substrate shift from human text (middle row) to LLM-written description (bottom row) lowers broad-lexicon agreement by 20 percentage points ($0.70 \rightarrow 0.50$) and discriminative-tail agreement by 6 percentage points ($0.28 \rightarrow 0.22$), making explicit how much of the middle-row agreement was carried by the underlying human corpus rather than by any LLM-level convergence. The structural pattern of the bottom row echoes the model-family alignment from earlier sections: Phi-4 and Llama-3.3-70B are the most aligned pair on description-derived vocabularies (0.61 in Panel E, 0.32 in Panel F), echoing the heavy-compressor profile documented in Section 3.2; Grok-3-mini is the description outlier, with its pairs falling to

0.13–0.17 in Panel F, echoing its lower omission rate and more expansive vocabulary.

Each LLM's description-derived top-50 vocabulary has its own distinctive signature across the 26 categories.[9] GPT-4o-mini's distinctive terms cluster around a threat-and-security register (*breaches, security, threats, attacks, incidents, vulnerabilities, cybersecurity*). Grok-3-mini concentrates on technical financial and corporate-legal jargon (*covenants, indebtedness, libor, liens, convertible, staggered, expropriation, bylaws*). Phi-4 favors operational and procedural language (*supply, vendors, logistics, procedures, managing, hedging, charter*) alongside specific event and regulatory references (*covid, gdpr, asbestos*). Llama-3.3-70B leans on strategic-action verbs and corporate-action nouns (*acquisitions, refinance, takeover, faces, adapt, generate*) together with a thread of risk-event language (*terrorism, war, default, intense*). These signatures are not arbitrary lexical fingerprints; they reflect each model's prior over what risk-factor language sounds like, surfaced under the free-generation task where the wedge reduces to pure Ω_m.

Together, the six panels describe the wedge $w_m(T_i, P_i, \Omega_m)$. In panels B, E, and F, the influence of Ω_m was higher, leading to a lower cross-model agreement. This relates to the extrinsic distortion (especially in the case of panel B). In Panels A, C, and D, where the shared source and prompt constrain Ω_m's influence, cross-model agreement is higher but still leaves visible cross-LLM differences. This finding lines up with intrinsic distortion (consistent with Calibration distortion, though Realization contributions cannot be ruled out from the agreement data alone). Overall, this implies that the wedge carries a systematic, model-specific component, and the choice of LLM shapes the categorical, vocabulary, and (as we test next) construct-scoring outputs that researchers rely on.

3.7 Multi-labeling, Ordinal Ratings, and Continuous Scoring

We now examine how LLMs fare when in scoring tasks, a common use case of LLMs in empirical work. Using a structured prompt (see Appendix A.8), we elicit four LLM-generated constructs—severity, complexity, specificity, and boilerplate—from all four LLMs in our study. For each construct, the LLM reports (i) its evaluability for the disclosure (binary 0/1), (ii) an ordinal rating (1–5), and (iii) a continuous score (0.0–10.0).

These four constructs have well-established antecedents in the accounting and finance literature.

[9] For each LLM, we score each word by its total in-category count across categories where it is top-50 for that LLM, minus the maximum of that same count across the other three LLMs. The top of each LLM's ranked list is the vocabulary signature reported below.

Severity is studied by Campbell et al. (2014) and Kravet and Muslu (2013), who link risk-factor severity to stock-return volatility and cost of equity. Specificity, the degree to which disclosures provide firm-specific rather than generic risk information, was developed by Hope et al. (2016), who show that specific risk-factor disclosures improve analyst forecast accuracy. Complexity is related to readability and plain-English literature (Loughran and McDonald, 2014; Bonsall IV et al., 2017). Boilerplate, the share of disclosure that is formulaic rather than informative, was studied by Lang and Stice-Lawrence (2015) and Dyer et al. (2017). Our contribution is not to introduce these constructs but to examine how LLM-based measurement of them is shaped by the wedge.

Panel A of Table 3 reports construct scores from the four LLMs on Item 1A disclosures. Two patterns emerge. First, LLMs disagree on which constructs they can evaluate. Mean evaluability is uniformly high for severity (0.96–1.00 across the four LLMs), but spans 0.22 (Grok-3-mini) to 0.98 (GPT-4o-mini) for boilerplate, 0.70–1.00 for complexity, and 0.64–1.00 for specificity. Second, despite this disagreement, the score distributions cluster tightly within narrow bands. Severity continuous means range 6.68–7.31 across the four LLMs (ordinal 3.71–4.00); boilerplate 7.15–7.46 (3.76–3.98); complexity 5.64–6.00 (2.94–3.15); specificity 4.34–4.92 (2.43–2.82). Each construct settles near a central tendency that is similar across LLMs, with within-LLM standard deviations generally below 1.5 on the 0–10 scale.

Panel B repeats the exercise on earnings-call transcripts using GPT-4o-mini and Llama-3.3-70B. The compression around a central tendency persists within each LLM, but the central tendencies themselves are more model-specific on earnings calls than on Item 1A. Severity continuous, which clustered around 6.68–7.31 on Item 1A, drops sharply on earnings calls to 4.62 (GPT-4o-mini) and 3.03 (Llama-3.3-70B); boilerplate falls from 7.15–7.46 to 5.17–5.19. Cross-LLM agreement is also weaker on the earnings-call sample: complexity continuous diverges 5.74 (GPT) to 7.80 (Llama) — a 2.1-point gap on the 0–10 scale — and severity diverges by 1.6 points across the same two LLMs (compared to within 0.28 and 0.03 points respectively on Item 1A).

Across both panels, the scoring pattern is consistent with model priors and corpus content playing a substantial role alongside disclosure-specific content. Read through the wedge $w_m(T_i, P_i, \Omega_m)$ lens: the central tendency appears to be shaped by Ω_m with some interaction with T_i, and the within-LLM compression suggests the disclosure-specific component θ_i has limited leverage on the score. We test this formally in the next section through a control-function diagnostic that asks

whether the construct scores carry non-classical measurement error tied to the wedge proxies.

4 Implications of LLM Measurement Error

We have shown that the wedge $w_m(T_i, P_i, \Omega_m)$ exists and operates through systematic, model-specific channels. We now ask whether the wedge matters in practice for empirical research that uses LLM-derived measures as regressors. Our test is a control-function diagnostic: if the wedge proxies, Omission (O_{im}) and Injection (I_{im}), introduced in Section 3.2 carry independent information about firm outcomes beyond the LLM score itself, the score contains non-classical measurement error tied to mechanical features of LLM text processing. We apply the diagnostic to two settings: LLM-generated constructs measured on Item 1A and earnings-call corpora (Section 4.1), and the direct-prompt LLM predictors of Jha et al. (2024) (Section 4.2).

4.1 Diagnostic Tests of Non-Classical Measurement Error

For each LLM-derived measure L_{im} used as a regressor for firm outcome $y_{i,t}$, we estimate three regression specifications. The first is the naive regression of equation (1), reproduced in the empirical setting as

$$y_{i,t} = \beta_{\text{naive}}\, L_{im,t} + \gamma' X_{i,t} + \tau_t + \varepsilon_{i,t}, \tag{6}$$

where $X_{i,t} = \{\text{Size, MTB, CashFlow, Tangibility, Age}\}$ are standard firm controls and τ_t are year fixed effects. The second adds the wedge proxies directly to the naive regression:

$$y_{i,t} = \beta_{\text{joint}}\, L_{im,t} + \rho_O\, O_{im,t} + \rho_I\, I_{im,t} + \gamma' X_{i,t} + \tau_t + \varepsilon_{i,t}, \tag{7}$$

and we report the construct-coefficient sensitivity $|\Delta\%| = 100 \times |\beta_{\text{joint}} - \beta_{\text{naive}}|/|\beta_{\text{naive}}|$. The third decomposes the measure into a wedge-predictable part and a wedge-orthogonal residual via a first-stage Tobit regression of L_{im} on the wedge proxies and controls, yielding $\widehat{L_{im}}$ and $\hat{e}$, then estimates

$$y_{i,t} = \theta\, \widehat{L_{im,t}} + \lambda\, \hat{e}_{i,t} + \gamma' X_{i,t} + \tau_t + \varepsilon_{i,t}. \tag{8}$$

Under the maintained assumption that O_{im} and I_{im} are mechanical artifacts of LLM processing orthogonal to firm fundamentals, θ captures the wedge-driven effect on the score and λ is the wedge-adjusted coefficient on the construct. Standard errors are CR1 cluster-robust on firm throughout.

In panel A of Table 4, we report the diagnostic for four LLM-generated constructs — severity, boilerplate, complexity, specificity — on Item 1A disclosures, against six firm outcomes (NI/Assets, CapEx/Assets, R&D ratio, leverage, sales growth, and the WW financial-constraint index). The construct coefficients are wedge-sensitive in many cells: across the 24 (construct $\times$ outcome) cells, $|\Delta\%|$ ranges from 3.9% (complexity on R&D ratio) to 311% (complexity on leverage, where β_{naive} is itself insignificant). Severity's significance weakens in five of six cells when wedge proxies are added (see β_{joint} column): three (NI/Assets, CapEx/Assets, WW index) lose significance entirely, two (leverage, sales growth) drop from 5% to 10% significance, and only R&D ratio retains 1% significance. The decomposition is most informative for NI/Assets and CapEx/Assets, where the wedge channel θ is significant at $p < 0.01$ and the wedge-orthogonal residual λ is insignificant; in those cells the naive severity–outcome relationship is concentrated in the wedge-predictable component of the score. For WW index, neither θ nor λ is significant, so severity loses explanatory power without the wedge clearly taking over. Complexity and specificity sit between severity and boilerplate: complexity loses significance in two cells (NI/Assets, CapEx/Assets) and gains significance in one (leverage); specificity loses significance in three cells (NI/Assets, CapEx/Assets, leverage). Boilerplate is the most wedge-robust of the four: all six cells retain significance under the joint specification, with $|\Delta\%|$ between 11.7% and 20.1%, and both θ and λ significant in most outcomes.

In panel B, we report the same diagnostic for severity on earnings-call transcripts using `gpt-4o-mini` and `llama-3.3-70B`. The picture is qualitatively different from Panel A, but not because the wedge is absent. Severity's coefficient is more stable under wedge controls than in Item 1A: $|\Delta\%|$ ranges from 0.4% to 18.0%, and severity retains significance in five of six outcomes under the joint specification (the exception is R&D ratio, where β_{naive} is only marginally significant to begin with). Yet the wedge is still active. The wedge proxies ρ_O or ρ_I are individually significant in four of the six outcome cells, and the wedge channel θ in the decomposition is significant in four cells (NI/Assets, R&D ratio, sales growth, WW index). What distinguishes Panel B from Panel A is the role the construct plays once the wedge is partialled out: the wedge-orthogonal residual

λ remains significant in five of six cells, so the construct and the wedge each carry independent information about firm outcomes. In Panel A, by contrast, the wedge channel often *displaces* the construct when both are included — the wedge accounts for the score's apparent relationship with the outcome rather than coexisting with a separate construct signal. We believe two non-mutually exclusive scenarios are consistent with the Panel B–Panel A difference: (i) the earnings-call corpus carries more disclosure-specific content than Item 1A risk disclosures, so the construct retains independent signal once the wedge is controlled for; or (ii) the construct-scoring task on earnings calls more closely resembles a direct-prompt setting in which the LLM is anchored on an evaluable property. Either way, the wedge is present in both corpora; what varies is whether it displaces the construct or coexists with it.

Figure 6 summarizes the sensitivity distribution across the full grid of cells: 288 Item 1A cells (4 LLMs $\times$ 4 constructs $\times$ 6 outcomes $\times$ 3 fixed-effects variants) and 144 earnings-call cells (2 LLMs $\times$ 4 constructs $\times$ 6 outcomes $\times$ 3 fixed-effects variants). The Item 1A $|\Delta\%|$ distribution exceeds the 10% threshold in roughly 63% of all 288 cells (panel a) and in roughly 63% of the 83 naive-significant cells (panel b). The earnings-call distribution is shifted left: roughly 44% of all 144 cells exceed the threshold (panel c), and roughly 39% of the 59 naive-significant cells do so (panel d). The contrast confirms that wedge sensitivity is a property of the construct–corpus pairing, not a uniform feature of LLM-derived regressors.

An analogous two-stage analysis for an alternative LLM output — risk-category classifications across the 26 risk-factor categories — is reported in Appendix Table A.2: the wedge proxies jointly predict classification at $p < 0.001$ in every category, indicating that their first-stage relevance for LLM outputs extends beyond the construct scores examined here.

4.1.1 When are Wedge Correction Sufficient?

The wedge diagnostic provides three related tests of classical measurement error: (i) the wedge-proxy coefficients ρ_O or ρ_I are nonzero in the joint specification, (ii) the construct coefficient shifts when wedge proxies are added ($\beta_{\text{naive}} \neq \beta_{\text{joint}}$, equivalently $|\Delta\%|$ is nontrivial), and (iii) the wedge-predictable and wedge-orthogonal channels of the score have different effects on the outcome ($\theta \neq \lambda$ in the decomposition). Rejection by any of the three indicates the score contains measurement error correlated with the outcome. The three are related but not equivalent: (ii) requires both (i) and a

correlation between the construct and the wedge proxies, while (i) and (iii) can reject even when the construct is roughly orthogonal to the wedge proxies. Panel B is an instance: $|\Delta\%|$ is mostly under 5% — severity from earnings calls is roughly orthogonal to the wedge proxies, so adding them barely shifts β — yet the wedge proxies remain significant predictors of the outcome and the decomposition channels differ.

Whether λ recovers an unbiased estimate of the underlying effect is a stronger claim and rests on the maintained assumption that O_{im} and I_{im} are mechanical processing artifacts orthogonal to firm fundamentals. Under that assumption, the wedge-orthogonal component of L_{im} inherits the signal that the construct genuinely carries about θ_i, and λ delivers a wedge-purified coefficient.

The assumption can fail in two ways, both of which compromise λ as a correction but leave the test valid. The first is incompleteness of the proxies: the wedge may operate through channels that O_{im} and I_{im} do not capture. If Ω_m leaks into the score through semantic priors that the lexical proxies do not register, then $\hat{e}$ still contains some wedge contamination and λ remains biased. The second is firm-side leakage: O_{im} and I_{im} may be correlated with firm fundamentals, not only with LLM processing. Vague disclosures may compress easily and also come from firms that systematically underperform; specific disclosures may resist compression and come from firms with different fundamentals. In that case the wedge proxies absorb real firm signal alongside processing artifacts, and λ is biased toward zero relative to the underlying effect.

The diagnostic's force as a test of non-classical measurement error survives both failure modes. Its force as a bias correction does not. In practice, researchers reporting β_{naive}, β_{joint}, the wedge-proxy coefficients ρ_O and ρ_I, and the sensitivity $|\Delta\%|$ together with the decomposition θ and λ allow readers to judge both how much measurement error the naive specification carries and how confident to be in the wedge-adjusted estimate.

4.2 Jha et al. (2024) Replication

The Section 4.1 results show that wedge contamination is non-trivial in construct-style LLM measures on Item 1A. The empirical literature using LLM-derived regressors does not always use construct ratings, however: Jha et al. (2024) prompt an LLM with single-question, outcome-specific tasks (e.g., does this earnings-call commentary suggest the firm will increase capital expenditure?) and use the resulting directional scores as predictors of realized firm outcomes. This is the leading

empirical example of what we call *direct-prompt* LLM measurement: the model is asked a single question targeted at a specific economic quantity, rather than a multi-attribute construct rating. We replicate their setting on our S&P 500 sample and apply the same wedge diagnostic to ask whether the contamination we document for constructs also affects this style of measurement.

Table A.3 replicates three headline results from Jha et al. (2024): their Investment score forecasting capital expenditure at $t+2$ (their Table 3 col. 4), capital expenditure at $t+4$ (Table 4 col. 2), and PT-scaled R&D at $t+2$ (Table 5 col. 8). We follow their specification — firm and year-quarter fixed effects with their five-control set {Size, MTB, CashFlow, Leverage$_t$, CapEx$_t$} — but on our S&P 500 corpus (2006–2024) using `gpt-4o-mini` and `llama-3.3-70B` in place of their `gpt-3.5-turbo`. The capital-expenditure result reproduces cleanly with both models: standardized betas at $t+2$ are 0.068 (GPT) and 0.059 (Llama), both within the 0.055–0.079 range Jha et al. (2024) report. The $t+4$ result is similar. The R&D result is the weakest of the three (marginal for Llama at $t = 1.95$, insignificant for GPT); Jha et al. (2024) themselves report a smaller standardized effect for R&D than for capital expenditure, so the relative weakness of the result is consistent across the original and our replication, though our R&D point estimates fall below conventional significance levels. On balance the direct-prompt measure transfers to our corpus with newer models, allowing us to subject it to the same wedge diagnostic we applied to construct measures.

Table 5 reports the diagnostic from equations (6)–(8) on the J-W direct predictors. Each predictor is paired only with its matched outcome (the capital-expenditure predictor against future capital expenditure, the cash predictor against future cash holdings, and so on), since a directional forecast about one quantity is interpretable only as a predictor of that quantity. The first stage for the decomposition is OLS rather than Tobit because the J-W score is continuous on the $-1/+1$ scale. The direct-predictor coefficients are strikingly insensitive to the wedge proxies. Across the 14 rows in Panels A and B, $|\Delta\%|$ is below 2% in 13 of 14 cells, and no naive-significant cell loses significance under the joint specification. The decomposition columns echo the joint result: λ closely tracks β_{naive} throughout, while θ on the wedge-predictable component is insignificant in every cell. The single exception to the small-$|\Delta\%|$ pattern is Llama's dividend predictor (14.5% sensitivity), where β_{naive} is itself insignificant.

The asymmetry between Sections 4.1 and 4.2 sharpens the paper's contribution. Construct-style measures (Table 4 Panel A) carry meaningful wedge contamination: many naive-significant cells lose

significance when wedge proxies are added, and in several of those cells the decomposition attributes the predictive content to the wedge channel θ. For the J-W direct predictors (Table 5), the construct coefficient is essentially unchanged by the inclusion of wedge proxies ($|\Delta\%|$ below 2% in 13 of 14 cells), the wedge channel θ is insignificant in every cell, and λ closely tracks β_{naive}. We do not read this as the absence of wedge effects — the wedge-proxy coefficients ρ_O or ρ_I are individually significant in 3 of 14 cells, and the CF diagnostic only purges contamination measurable through Omission and Injection — but as evidence that the J-W score is approximately orthogonal to these wedge proxies, so that whatever wedge contamination is present does not bias the construct's relationship with firm outcomes. The generated-regressor problem manifests as coefficient bias when the construct is systematically correlated with the wedge proxies; the direct-prompt design appears to produce scores for which that correlation is small. The design of the prompt — a single outcome-anchored question versus a multi-attribute construct rating — is itself a determinant of how vulnerable the downstream regression is to measurement-error bias.

5 Working with the LLM Wedge

The method outlined in Section 4.1 provides us a roadmap to identify and correct the part of the wedge that the wedge proxies ρ_O and ρ_I can see. Two additional mitigations could address what those proxies cannot. Ensemble agreement across models filters the idiosyncratic distortion no single model's proxies capture; cross-model variance supplies an inference correction the proxies do not address. In this section, we map each downstream task to its dominant cell of Exhibit 2 and to the mitigation it warrants.

5.1 Practical Guidance by Task Type

Text generation and dictionary construction. The dominant pathology is injection (extrinsic distortion): LLMs introduce terms absent from the source, loading the dictionary with model priors. Diagnose with ρ_I on discriminative-term lists; categories with ρ_I above 50% on top terms warrant skepticism (Figure 1 Panel B). Mitigate by intersecting or weighting vocabularies across models. Section 5.2 shows cross-model agreement marks low-wedge items.

Classification and labeling. The dominant pathology is shared model priors driving category

assignment rather than text content. Diagnose with cross-model agreement by category (Figure 5 Panel A); treat categories below 40% pairwise agreement with caveats or restrict the sample to them. Mitigate by ensemble voting, reporting per-item agreement alongside the assignment.

Scoring and ranking. The dominant pathology is calibration distortion (Exhibit 2, Distortion A): LLMs compress scores into a narrow band and inherit prior-driven central tendency. Diagnose with cross-model dispersion in ordinal ratings and continuous scores (Table 3). The Item 1A panel shows that on the same construct LLMs differ by 0.29 ordinal points on Severity (Grok 3.71 vs Llama 4.00) and 0.22 on Boilerplate (Phi-4 3.76 vs GPT 3.98) on a 1–5 scale, and that continuous severity scores cluster within a 0.63-point band (Grok 6.68 vs Llama 7.31) on a 0–10 scale — the rating is meaningfully model-dependent before any aggregation. Mitigate by reporting cross-model dispersion alongside the rating, treating scores as categorical when dispersion is large relative to the rating range, or averaging with model-specific calibration adjustments.

Regression with LLM-derived regressors. The dominant pathology is non-classical measurement error: the wedge correlates with the construct or the outcome, biasing β. Diagnose with the CF tests of Section 4.1 — significance of ρ_O or ρ_I, a gap between β_{naive} and β_{joint}, and the θ–λ split — and report all three specifications. Mitigate by treating β_{joint} as the bias-aware estimate; a θ that is large relative to λ in the decomposition signals that the documented relation is wedge-driven and should be interpreted as such.

5.2 Ensemble Agreement

If different LLMs carry different idiosyncratic distortions, consensus across models should mark cases where the surviving signal is low-wedge. Figure 7 provides the empirical support. For each canonical risk factor identified by all four LLMs, we compute the Doc2Vec cosine between the LLM's source disclosure and its own summary and bucket by the number of *other* LLMs assigning the same primary subcategory. The raw-cosine shift from no peer agreement to unanimous peer agreement is positive and monotonic-or-near-monotonic for all four LLMs, ranging from +0.010 for Phi-4 to +0.026 for GPT-4o-mini (all $p < 10^{-170}$); the figure plots these shifts in σ units of each LLM's cosine distribution for cross-LLM comparability. When peers converge on the subcategory,

the focal LLM's summary tracks the source more closely; when peers disagree, it paraphrases more freely. Inter-LLM agreement is therefore an observable marker of low wedge.

This licenses ensemble agreement as a second-stage filter on wedge variation that ρ_O and ρ_I cannot see, since the proxies read only one model's vocabulary while different models err differently. In practice: restrict the analysis sample to items above an agreement threshold, weight observations by agreement, or report the high-agreement subsample alongside the full sample. The cost is operational — multiple model runs and outputs to store — and is non-trivial at corpus scale.

5.3 LLM Standard Errors

Once the LLM regressor is treated as generated, conventional inference is invalid: standard errors ignore the variance contributed by the generator, and clustering does not apply because the generator's internal distribution is unobserved. Re-prompting the model at every bootstrap iteration is prohibitive at scale. When multiple models are already in hand for ensemble agreement, the cross-model variance at the observation level supplies an estimate of generator uncertainty at no additional inference cost; it can be carried into heteroskedasticity-robust standard errors or used as inverse weights in WLS.[10] Cross-model variance addresses inference, not bias — it complements the CF diagnostic and ensemble agreement, which address the bias channels of the wedge.

6 Conclusion

Large language models are increasingly used as measurement instruments in empirical research, but they are not neutral ones. Every LLM output is a transformation of the input in non-random ways. We formalize this concept by treating any LLM output as a generated regressor, $L_{im} = \theta_i + w_m(T_i, P_i, \Omega_m)$: a noisy proxy for the latent construct θ_i contaminated by a model-specific wedge w_m that reflects how the model compresses input, draws on its priors, and allocates attention. The wedge takes two forms: calibration distortion, which rescales real effects, and realization distortion, which manufactures associations the source does not support. Both violate the classical measurement-error assumptions that would justify treating LLM scores as unbiased proxies.

[10] Token-level log probabilities, exposed by some models, would give a within-model analog to this cross-model variance. We are currently testing whether log-prob–based variance estimates align with the cross-model estimates used here.

We empirically show that the wedge influences LLM outputs. LLMs summarizing Item 1A risk factors omit roughly half of the source vocabulary, inject terms drawn from model priors at category-specific rates that exceed 50% on top discriminative slots, and shift the tonal register of disclosures along Loughran–McDonald and concreteness dimensions; the same attention allocation that drives this in summarization recurs when the model scores constructs. Cross-model agreement on risk category labels and on discriminative vocabularies varies sharply across the 26 risk categories, with classification agreement below 40% in several. When LLM-derived constructs are used as regressors on firm outcomes, including Omission and Injection as wedge proxies eliminates the naive significance in many cells, a signature of non-classical measurement error rather than attenuation bias.

These results carry direct practical implications. Researchers using LLM-derived regressors should report the wedge proxies alongside their main estimates and run the joint and decomposition specifications as a diagnostic. Where multiple models are run, cross-model agreement filters the idiosyncratic distortion that no single model's proxies can see, and cross-model variance supplies a cheap inference correction for the variance that conventional standard errors ignore. By treating LLM outputs as generated regressors with a measurable wedge, rather than as costless replacements for human readings, the field can extract reliable signal from textual disclosures and build empirical findings that better withstand scrutiny.

References

Battaglia, L., Christensen, T., Hansen, S., and Sacher, S. (2024). Inference for regression with variables generated by ai or machine learning. *arXiv preprint arXiv:2402.15585*.

Bhagwat, V., Cookson, J. A., and Chukwuma Dim, M. N. (2025). The market's mirror: Revealing investor disagreement with LLMs. *SSRN Working Paper*.

Bonsall IV, S. B., Leone, A. J., Miller, B. P., and Mergenthaler, K. (2017). A plain english measure of financial reporting readability. *Journal of Accounting and Economics*, 63(2-3):329–357.

Brysbaert, M., Warriner, A. B., and Kuperman, V. (2014). Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911.

Bybee, L. (2024). The ghost in the machine: Generating beliefs with large language models. *Working Paper*.

Bybee, L. (2025). LLM-generated beliefs and macroeconomic expectations. *Working Paper*.

Campbell, J. L., Chen, H., Dhaliwal, D. S., Lu, H.-m., and Steele, L. B. (2014). The information content of mandatory risk factor disclosures in corporate filings. *Review of Accounting Studies*, 19(1):396–455.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *Econometrics Journal*, 21(1):C1–C68.

Cohen, L., Malloy, C., and Nguyen, Q. (2020). Lazy prices. *Journal of Finance*, 75(3):1371–1415.

Dyer, T., Lang, M., and Stice-Lawrence, L. (2017). The evolution of 10-k textual disclosure: Evidence from latent dirichlet allocation. *Journal of Accounting and Economics*, 64(2-3):221–245.

Egami, N., Hinck, M., Stewart, B. M., and Wei, H. (2023). Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. In *Advances in Neural Information Processing Systems*.

Eisfeldt, A. L., Schubert, G., Zhang, M. B., and Taska, B. (2023). Generative AI and firm values. *SSRN Working Paper No. 4436627*.

Engelberg, J., Manela, A., Mullins, W., and Vulicevic, L. (2025). Entity neutering. *SSRN Working Paper 5182756*.

Fathollahi, M., Grieser, W., and Venugopal, B. (2026). Operational resilience: Managing unhedgeable risks. *Working Paper*.

Fong, C. and Tyler, M. (2021). Machine learning predictions as regression covariates. *Political Analysis*, 29(4):467–484.

Gentzkow, M., Kelly, B. T., and Taddy, M. (2019). Text as data. *Journal of Economic Literature*, 57(3):535–574.

Grimmer, J., Roberts, M. E., and Stewart, B. M. (2022). *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton University Press.

Grusky, M., Naaman, M., and Artzi, Y. (2018). Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 708–719.

He, S., Lv, L., Manela, A., and Wu, J. (2025). Chronologically consistent large language models. *arXiv preprint arXiv:2502.21206*.

Hoberg, G. and Phillips, G. M. (2025). Scope, scale, and concentration: The 21st-century firm. *The Journal of Finance*, 80(1):415–466.

Hope, O.-K., Hu, D., and Lu, H. (2016). The benefits of specific risk-factor disclosures. *Review of Accounting Studies*, 21(4):1005–1045.

Huang, A. H., Wang, H., and Yang, Y. (2023). Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*, 40(2):806–841.

Jha, M., Qian, J., Weber, M., and Yang, B. (2024). Chatgpt and corporate policies. *NBER Working Paper 32161*.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Kim, A., Muhn, M., and Nikolaev, V. (2023). Bloated disclosures: Can ChatGPT help investors process information? *Working Paper*.

Kravet, T. and Muslu, V. (2013). Textual risk disclosures and investors' risk perceptions. *Review of Accounting Studies*, 18(4):1088–1122.

Laban, P., Schnabel, T., Bennett, P. N., and Hearst, M. A. (2022). SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Lang, M. and Stice-Lawrence, L. (2015). Textual analysis and international financial reporting: Large sample evidence. *Journal of Accounting and Economics*, 60(2-3):110–135.

Lopez-Lira, A. and Tang, Y. (2023). Can ChatGPT forecast stock price movements? Return predictability and large language models. *Working Paper*.

Lopez-Lira, A. and Tang, Y. (2024). Large language models and return prediction. *Working Paper*.

Lopez-Lira, A., Tang, Y., and Zhu, M. (2025). Can ChatGPT forecast stock price movements? Return predictability and large language models. *Working Paper*.

Loughran, T. and McDonald, B. (2011). When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *Journal of Finance*, 66(1):35–65.

Loughran, T. and McDonald, B. (2014). Measuring readability in financial disclosures. *Journal of Finance*, 69(4):1643–1671.

Ludwig, J., Mullainathan, S., and Rambachan, A. (2024). The unreasonable effectiveness of large language models in forecasting. *Working Paper*.

Ludwig, J., Mullainathan, S., and Rambachan, A. (2025). Large language models: An applied econometric framework. *NBER Working Paper 33344*.

Maynez, J., Narayan, S., Bohnet, B., and McDonald, R. (2020). On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1906–1919.

Murphy, K. M. and Topel, R. H. (1985). Estimation and inference in two-step econometric models. *Journal of Business and Economic Statistics*, 3(4):370–379.

Narayan, S., Cohen, S. B., and Lapata, M. (2018). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of EMNLP*, pages 1797–1807.

Pagan, A. (1984). Econometric issues in the analysis of regressions with generated regressors. *International Economic Review*, 25(1):221–247.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Sarkar, S. K. and Vafa, K. (2024). Lookahead bias in pretrained language models. *SSRN Working Paper 4754678*.

See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. In *Proceedings of ACL*, pages 1073–1083.

Sikochi, A. S., Cohen, L., and Malloy, C. (2024). LLM-assisted ESG disclosure credibility assessment. *Working Paper*.

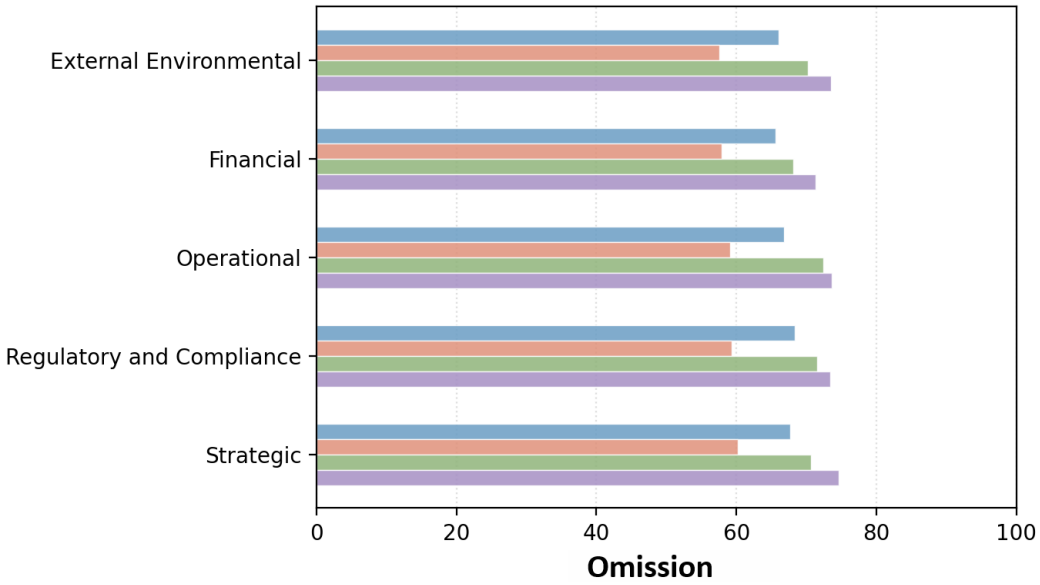
Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *Journal of Finance*, 62(3):1139–1168.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*.

Figure 1: **Omission and Injection Across Risk Factors**

In panels (a) and (b), we plot the omission and injection rates for discriminatory terms identified in our S&P 500 sample for each risk category. Omission is defined as $O_{im} = 1 - \frac{|V_{i,\text{source}} \cap V_{im,\text{summary}}|}{V_{i,\text{source}}}$. Injection is defined as $I_{im} = 1 - \frac{|V_{im,\text{summary}} \cap V_{i,\text{source}}|}{V_{im,\text{summary}}}$. We define *source* and *summary* vocabularies as the count of unique discriminative terms used by companies in risk disclosures Item 1A that we identify and LLM summaries, respectively. Higher omission indicates greater vocabulary reduction, and higher injection indicates more novel terminology introduced by the LLM.

(a) Omission of Discriminatory Terms



(b) Injection of Discriminatory Terms

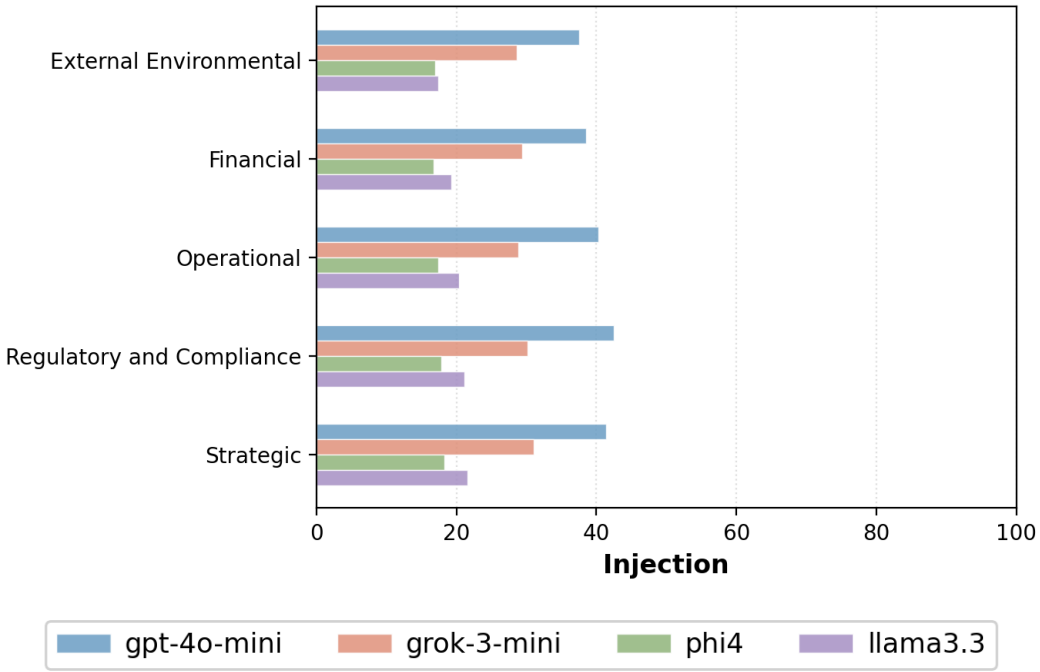


Figure 2: **Model Attention and LLM-generated Regressors.** In this figure, we plot the average per-token attention mass that Llama-3.3-70B and Phi-4 allocate to the first 10%, middle 80%, and last 10% of the input Item 1A when prompting for severity, specificity, complexity, and boilerplate (generated regressors). Per-token attention mass is the share of total attention in a position bin divided by the bin's share of input tokens; a value of 1.0 (dotted line) denotes uniform attention. Attention weights are extracted at the token where each model emits its continuous score for that construct, averaged across all transformer layers and attention heads. Means are taken across the 4,829 disclosure chunks (each up to 2,000 tokens) in our sample.

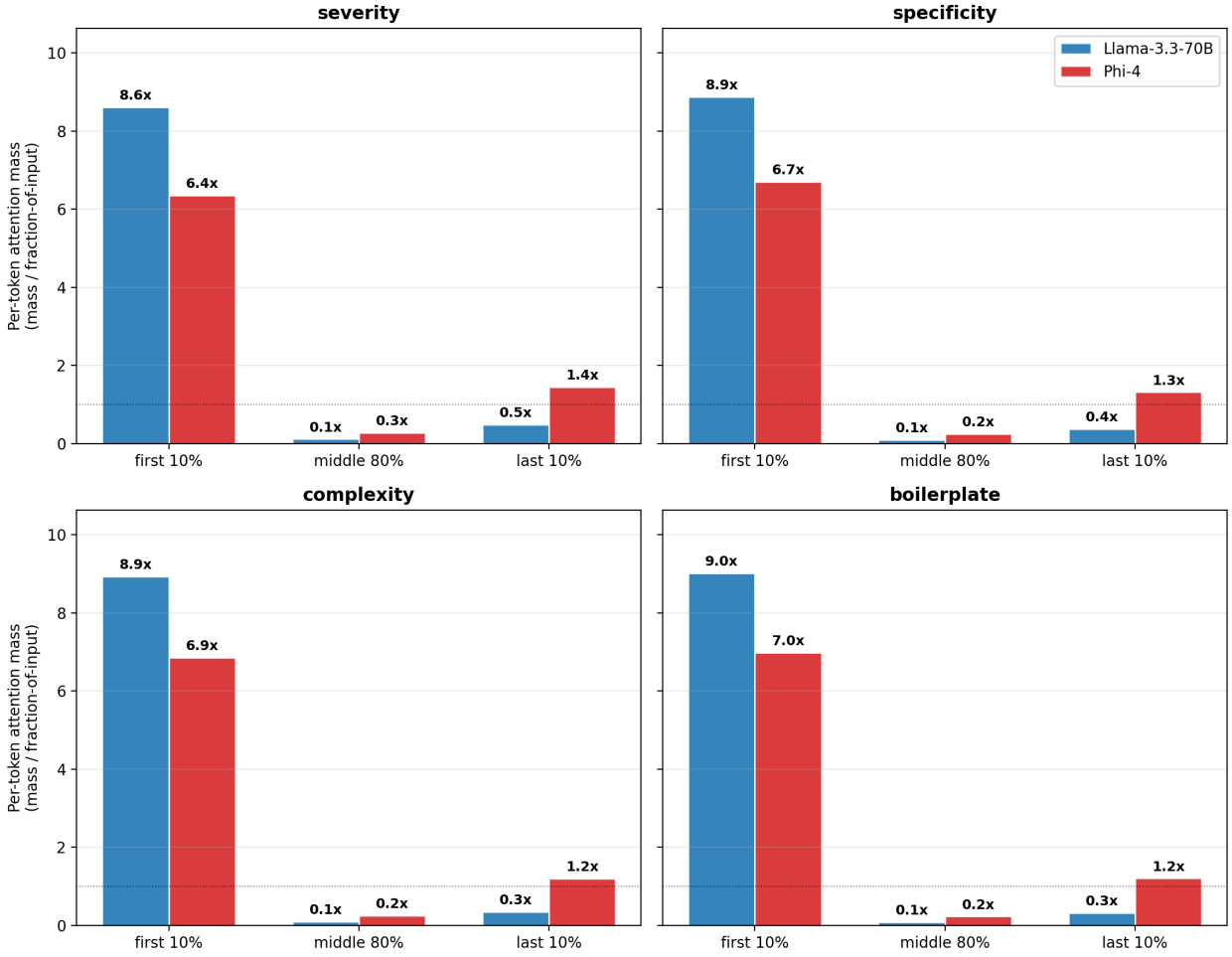


Figure 3: **Fidelity Lost During LLM Summarization**

This figure plots the average semantic fidelity lost when LLMs summarize risks disclosed in Item 1A. We split the fidelity lost into two complementary channels. The left half-axis (red) shows '1 - Coverage', which is the share of source meaning the summary failed to retain. The right half-axis (blue) shows '1 - Grounding', which is the share of summary content not anchored in the source. We embed the risk disclosure and LLM summary sentences in `all-mpnet-base-v2` embedding space. Using the embedding vectors, we identify the LLM summary sentence that has the highest cosine similarity for each risk disclosure sentence, and vice versa. For each risk factor in our sample, Coverage is the average cosine similarity of risk disclosure sentences with their top matched summary sentences, and grounding is the average cosine similarity of summary sentences with their top match risk disclosure sentences. The bars represent the means across the 3.4 million risk factors for S&P 500 firms identified by our four models. The center vertical axis at zero represents the ideal scenario when an LLM summarizes a risk disclosure without any fidelity loss (semantic drift). The average total fidelity lost by each model is noted below.

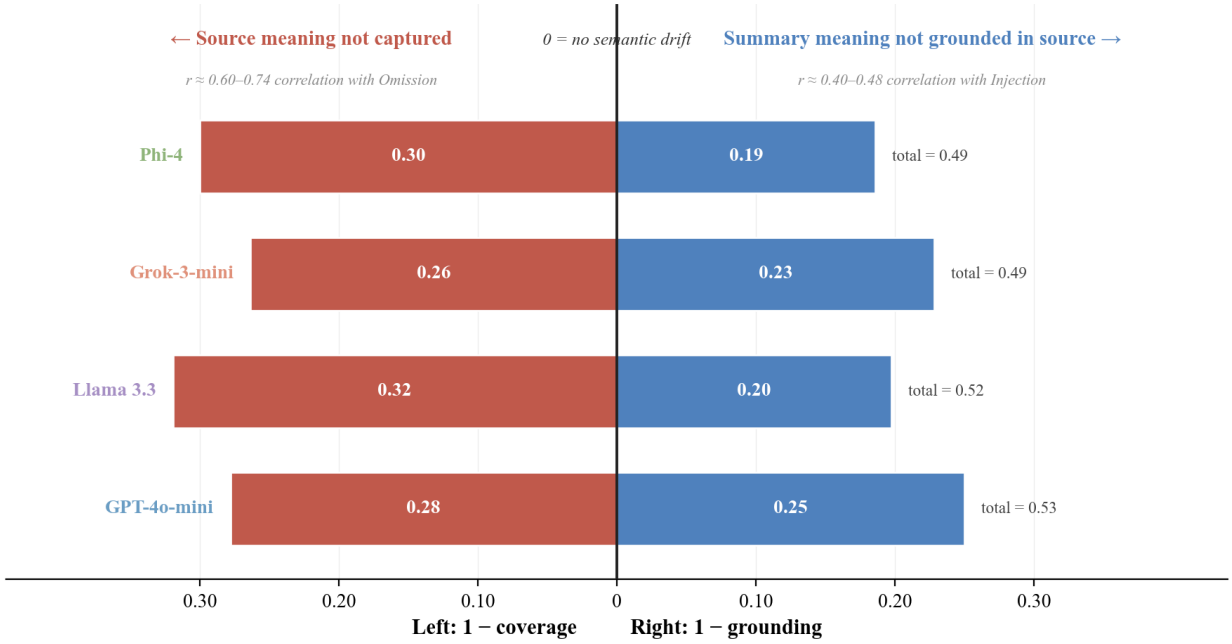
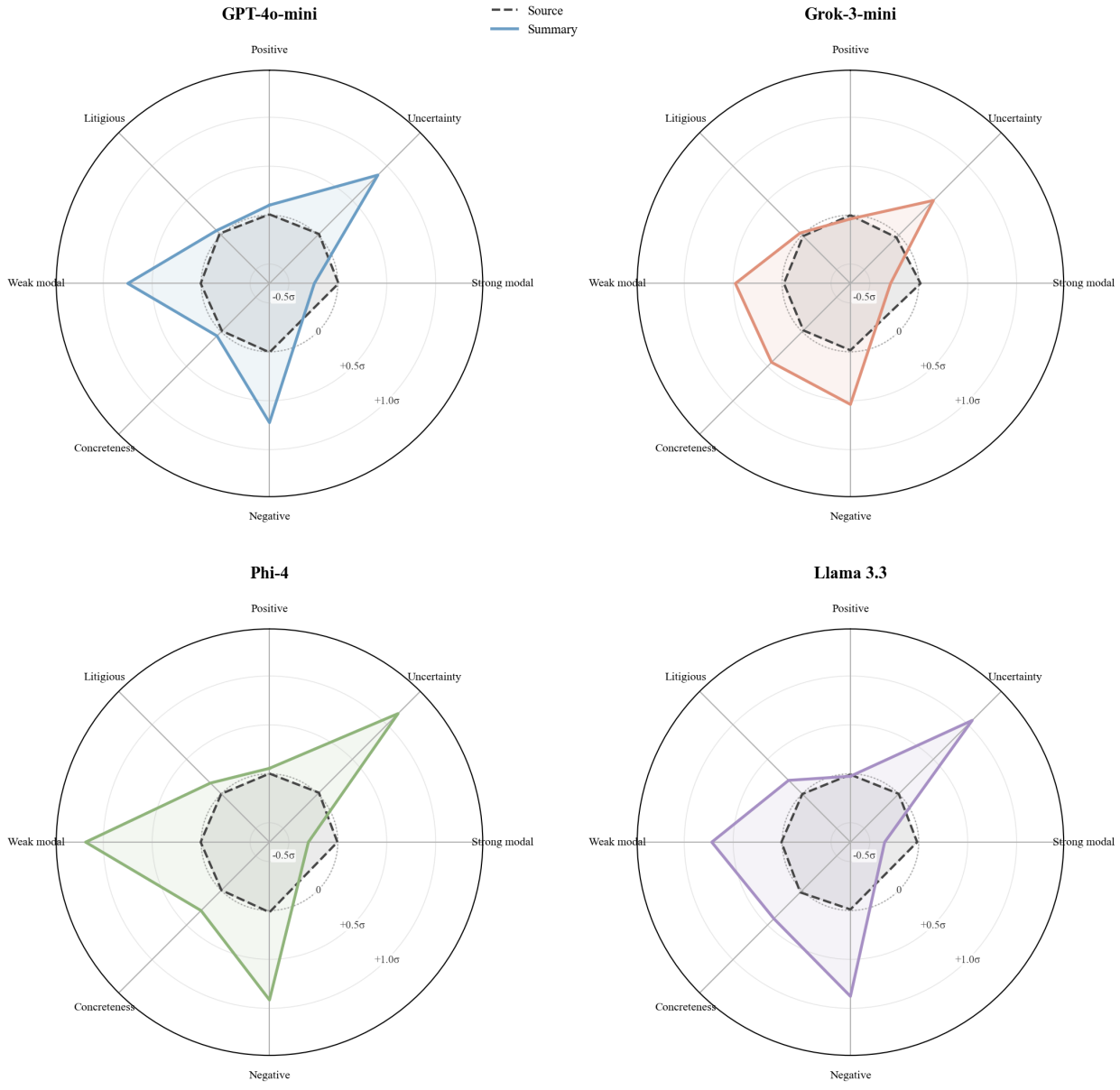


Figure 4: **Tonal and Semantic Changes from Source to Summary**

In panel A of this figure, we plot the average tone profile of Item 1A risk disclosure text (source, dashed black line) and LLM summary (solid) in a seven-dimensional tonal space: six Loughran-McDonald tone categories - positive, negative, uncertainty, litigious, and strong- and weak-modal - each measured as a share of the text's tokens, plus Brysbaert et al. (2014) concreteness (a per-token average). Each dimension is standardized against the distribution of source disclosures, so a value of $+1\sigma$ means the summary's density on that dimension lies one source-standard-deviation above the typical source disclosure. The dotted ring at 0σ marks the source baseline, and the rings are labeled in standard deviation units. We place semantically opposed dimensions on shared diameters, positive vs. negative, uncertainty vs. concreteness, strong- vs. weak-modal, and litigious alone. In panel B we plot three time series for each LLM: source–summary document-level MPNet cosine similarity (dark grey), Omission rate (red), and Injection rate (blue). The lexical omission and injection rates are computed over each LLM's discriminative-term vocabulary (top-1000 by lift per risk-factor subcategory). All three series are z-scored within (LLM × metric) so that the within-LLM year-to-year co-movement between metrics reads on a common scale. Shaded bands are 95% confidence intervals.

Panel A: Changes in tone



Panel B: Changes in semantic similarity, Omission, and Injection rates over time

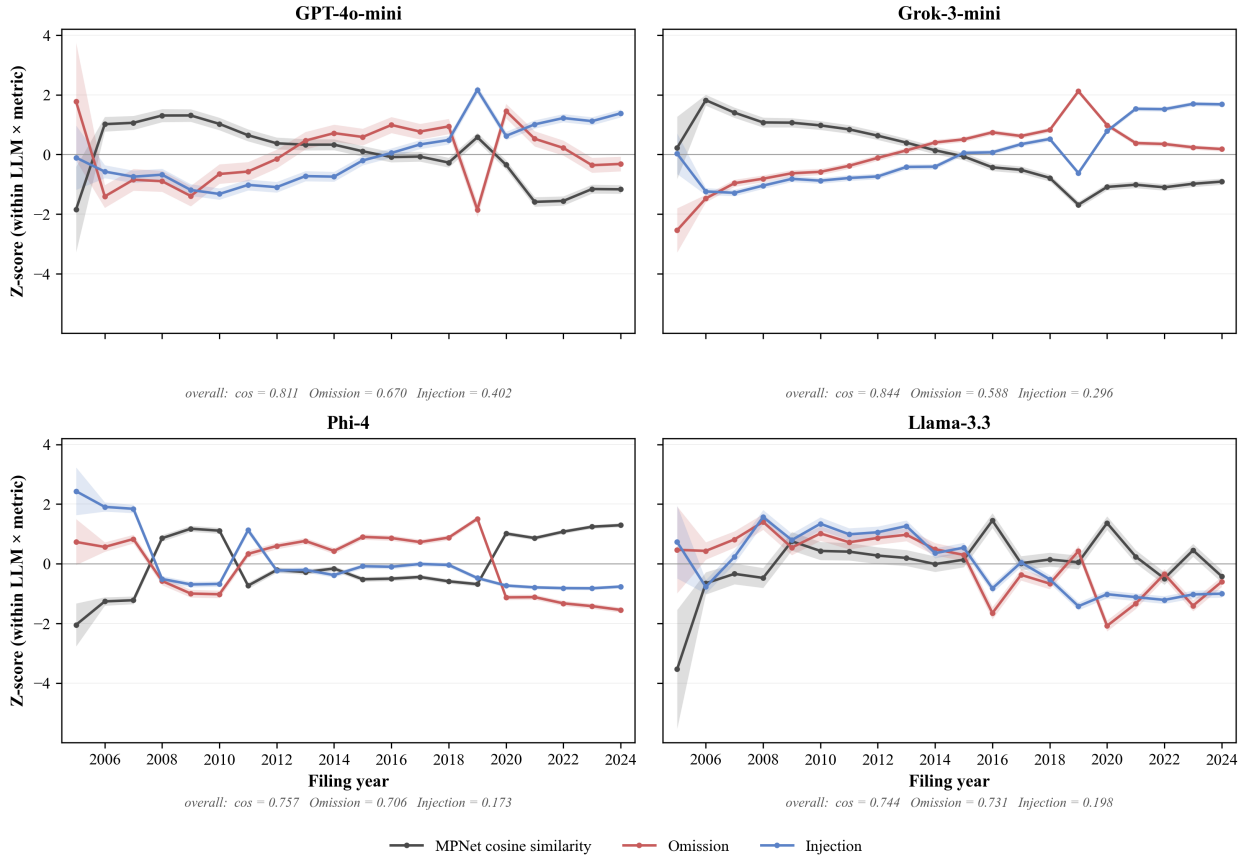
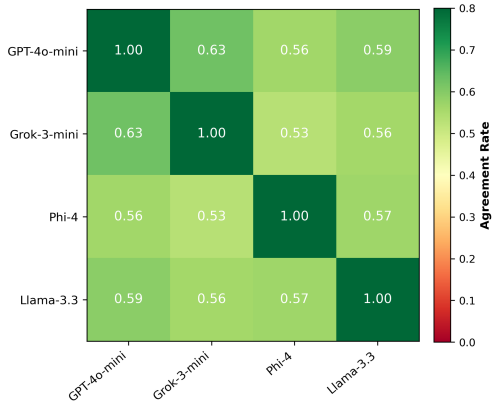
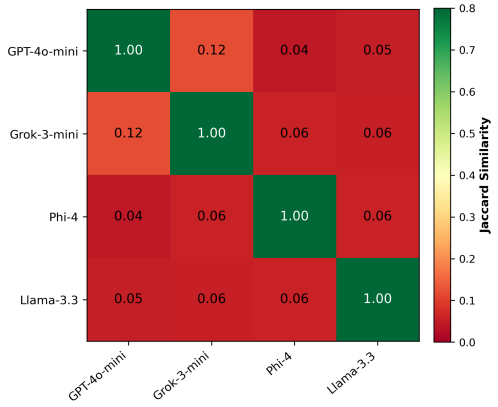


Figure 5: **Risk Factor Classification and Discriminatory Terms Agreement Across LLMs**

In this figure, we report the pairwise agreement across four LLMs for risk factor classification and discriminatory terms, averaged over our 26 risk factor categories. In panel (a), we show the pairwise classification agreement among the LLMs based on risk factors identified in Item 1A text. In panel (b), we ask LLMs to generate discriminatory terms for each of the 26 risk factor categories in our sample without feeding our data, which proxy for model priors. Panels (c) and (d) show the pairwise agreement of discriminatory terms identified using a TF-IDF process on raw Item 1A text for each risk factor category; panels (e) and (f) show the same fo the LLM generated summaries. Panels (c) and (e) are based on all discriminatory terms we identify using the TF-IDF process while panels (d) and (f) are based on the 50 most discriminatory terms.

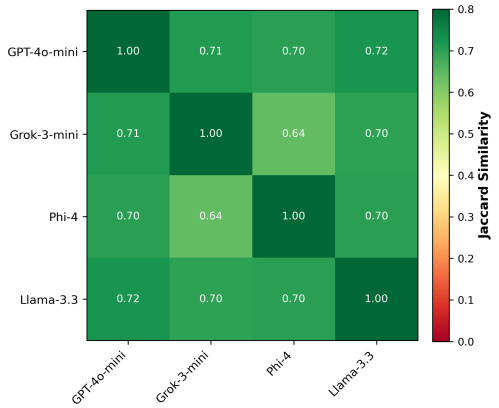


(a) Risk factor category agreement

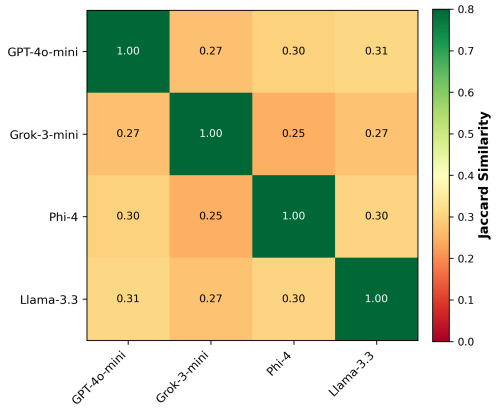


(b) LLM generated discriminatory terms

Raw Disclosure Sentences

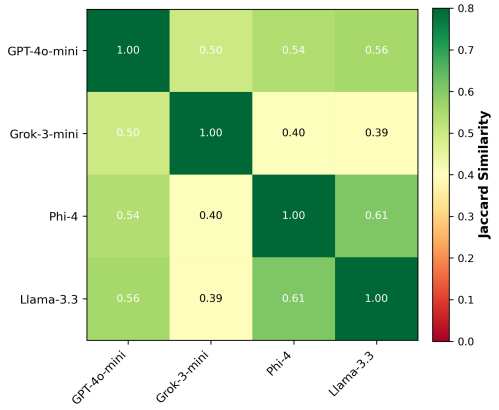


(c) All terms with lift > 0

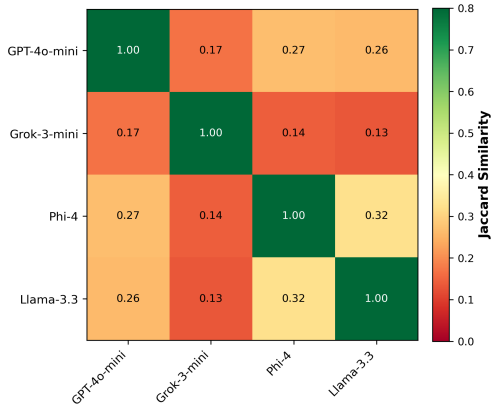


(d) Top-50 discriminatory terms

LLM-Generated Risk Descriptions



(e) All terms with lift > 0

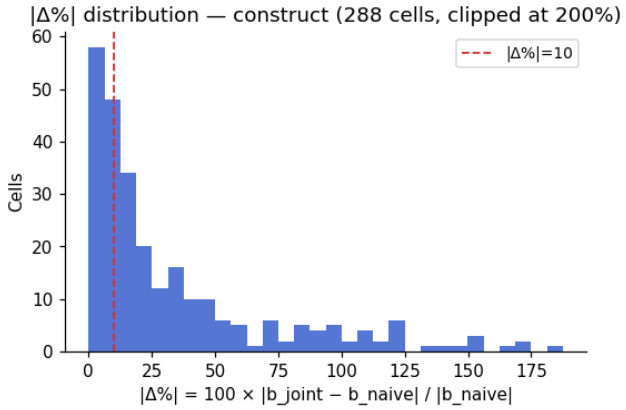


(f) Top-50 discriminatory terms

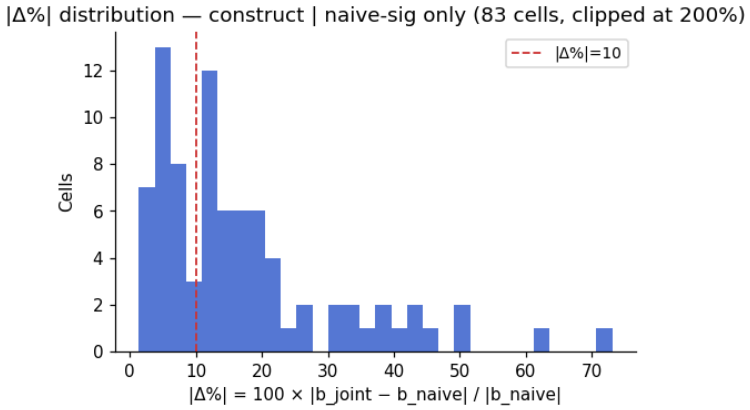
Figure 6: **Diagnostic Distributions: LLM Text Constructs**

In this figure, we plot the sensitivity of LLM generated regressors discussed in Table 4 to the inclusion of omission and injection measures and changes in fixed effects. In panels (a) and (b) of this figure, we report the sensitivity for coefficients in the Item 1A sample (panel a) of Table 4 and in panels (c) and (d) we do the same for the coefficients in the earnings call sample (panel b) of Table 4. We define coefficient sensitivity as $|\Delta\%| = 100 \times |b_{\text{joint}} - b_{\text{naive}}|/|b_{\text{naive}}|$. The Item 1A panels are based on 4 LLM models (gpt-4o-mini, llama-3.3-70B, grok-3-mini, phi-4) $\times$ 4 constructs (severity, boilerplate, complexity, specificity) $\times$ 6 firm outcomes $\times$ 3 fixed-effects variants. The earnings call panels are based on 2 LLM models (gpt-4o-mini, llama-3.3-70B) $\times$ 4 constructs $\times$ 6 firm outcomes $\times$ 3 fixed-effects variants. The red dashed line denotes 10% in the distribution.

Item 1A Risk Disclosures

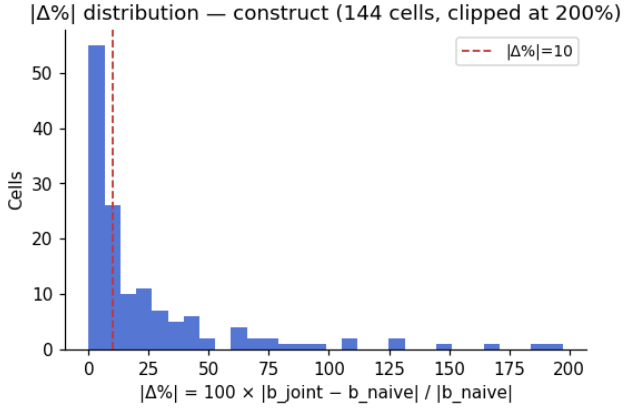


(a) $|\Delta\%|$ distribution — all 288 cells

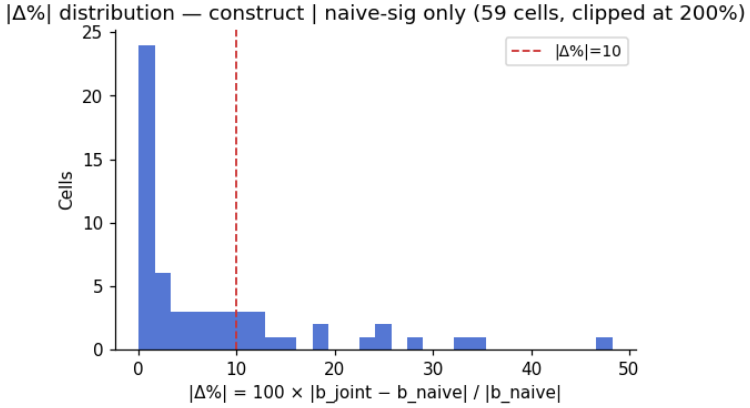


(b) $|\Delta\%|$ distribution — naive-significant cells (n=83)

Earnings Calls



(c) $|\Delta\%|$ distribution — all 144 cells



(d) $|\Delta\%|$ distribution — naive-significant cells (n=59)

Figure 7: **LLM Ensemble Agreement of Risk Factor Categories and Embedding Similarity**

The figure plots the shift in mean Doc2Vec cosine similarity between an LLM's source-text disclosure and its corresponding summary text (the LLM's own description of the risk factor), expressed in standard-deviation units and measured relative to a bucket-0, grouped by the number of other LLMs that assigned the same primary subcategory to the underlying risk factor. The x-axis takes values in 0, 1, 2, 3 representing the number of other models that agree with the focal model's risk factor category classification. The y-axis is the cosine similarity shift in σ units (vs bucket 0): for each LLM, we subtract the bucket-0 mean cosine from each disclosure's cosine and divide by that LLM's standard deviation, then average within each bucket; bucket 0 sits at exactly zero by construction and bucket k reads as how many standard deviations of that LLM's cosine distribution the bucket-k mean sits above bucket 0. This anchors all four LLMs at zero in bucket 0 and isolates the agreement-driven shift on a comparable scale across models. Embeddings come from a single PV-DBOW Doc2Vec model trained on the union of source and summary documents (300-dimensional vectors, window size 15, minimum count 5, 40 training epochs; hyperparameters following Hoberg and Phillips (2025)).

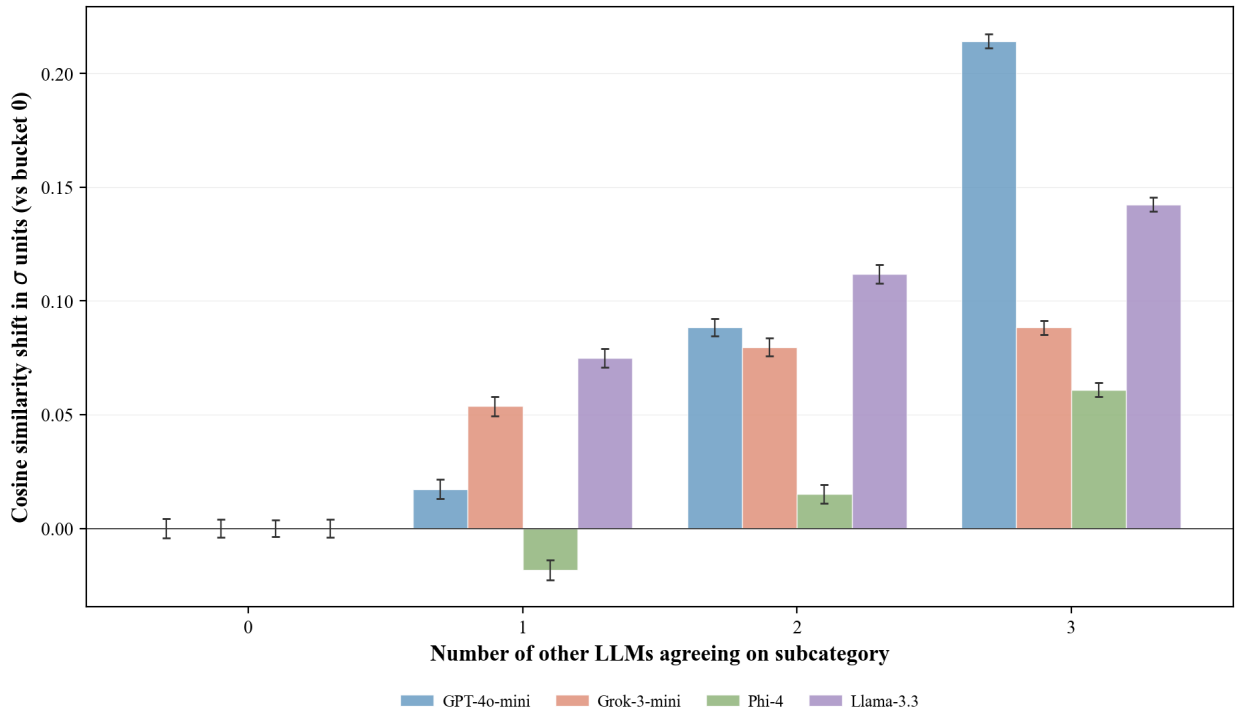


Table 1: **Omission Summary Statistics**

Panel A of this table reports aggregate summary statistics for four text sources across four LLMs; these are based on Item 1A risk disclosures. *Sentences* contain individual raw sentences discussing risk factors. *Keywords* contain uni- or bi-grams that LLMs identify as salient terms for grouping sentences into company-specific risks. *LLM* contains AI-generated summaries of Item 1A risks. *Full Disclosure* contains the complete set of Item 1A sentences associated with each identified risk. Top-N *Concentration* shows the percentage of total word frequency accounted for by the N most common words. *Entropy* extends this concept to the full word distribution. It is the normalized Shannon entropy of word frequencies, on [0,1], capturing how evenly tokens are spread across the vocabulary: values near 1 indicate diverse vocabulary use, lower values indicate concentration in a core vocabulary. Panel B is based on earnings call transcripts and report the metrics for raw earnings call text and the summaries generated by gpt-4o-mini and llama-3.3-70B.

Panel A: Item 1A risk disclosure text

			Concentration			
	Total Words	Vocabulary Size	Top-100 (%)	Top-500 (%)	Top-1000 (%)	Entropy
			gpt-4o-mini			
Keywords	1,185,692	5,017	54.30	85.99	93.60	0.739
Sentences	48,867,009	34,865	27.70	60.76	75.29	0.724
LLM Summary	14,826,475	15,010	41.39	74.03	85.78	0.718
Full Disclosure	103,587,174	36,904	29.40	63.16	77.57	0.708
			grok-3-mini			
Keywords	1,983,044	7,604	43.37	77.79	88.60	0.758
Sentences	66,656,778	39,174	27.14	59.99	74.62	0.719
LLM Summary	27,228,390	24,909	34.19	66.06	79.03	0.722
Full Disclosure	126,502,779	41,879	28.81	62.21	76.73	0.704
			phi4			
Keywords	1,454,344	6,020	50.78	83.85	92.33	0.740
Sentences	62,632,936	37,944	27.32	60.19	74.84	0.721
LLM Summary	9,813,635	16,801	36.13	69.82	82.43	0.736
Full Disclosure	118,078,881	39,633	28.99	62.48	76.98	0.706
			llama3.3			
Keywords	1,132,059	4,404	60.07	88.99	95.21	0.721
Sentences	58,914,054	37,329	27.53	60.44	75.04	0.721
LLM Summary	8,755,130	14,965	38.73	72.12	84.39	0.732
Full Disclosure	115,353,982	39,006	29.13	62.70	77.17	0.707

Panel B: Earnings call transcripts

Earnings call	139,475,964	132,427	31.58	61.50	74.79	0.646
gpt-4o-mini summary	19,587,545	53,666	38.32	68.53	80.21	0.663
llama-3.3-70B summary	21,949,144	62,953	39.32	67.10	78.49	0.655

Table 2: **Sample Vocabulary Injections**

This tables contrasts discriminative terms that appear exclusively in LLM-generated 1-shot summaries (injected) against terms in human-authored disclosure text for selected risk categories. LLM-injected terms appear in at least three of four models (GPT-4o-mini, Grok-3-mini, Phi-4, LLaMA-3.3) but are absent from human text and reflect model priors.

Risk Factor	LLM-Injected Terms	Human Text Terms
Natural/Physical Environment	*droughts, wildfires, toxic, flooding, radioactive, spills, discharges*	*temperatures, contaminated, cleanup, winter, catastrophes, tornadoes, pollution*
Reputation & Communication	*criticism, confusion, unethical, complaints, sentiment, sustainability*	*enhancing, allegations, recalls, misconduct, regulators, investigations*
Political Environment	*taiwan, wars, tensions, border, threat, chinese, priorities*	*nationalization, violence, hostilities, customs, greenhouse, ukraine*
Regulatory Compliance	*gdpr, bribery, epa, labeling, fcc, inspections*	*debarment, dodd, imprisonment, hipaa, disabilities, occupational*

Table 3: **LLM Task Evaluation Summary Statistics**

Panel A of this table summarizes the response of four LLMs to three tasks using Item 1A risk disclosures. Based on the given risk factor disclosure provided in the prompt: (1) can the LLM confidently evaluate Severity, Boilerplate, Complexity, and Specificity? (2) provide an ordinal ranking for the four constructs, and (3) provide a continuous score from 0 to 10 for the above four constructs. We report the mean, median, standard deviation, and inter-quartile range for the scores generated by the LLMs. Panel B reports the evaluation statistics when we prompt with the earnings call transcripts using got-4o-mini and llama-3.3-70B.

Panel A: Item 1A risk disclosure text

	Evaluability (binary 0/1)				**Ordinal Rating** (1–5)				**Continuous Score** (0–10)			
Statistic	Severity	Boilerplate	Complexity	Specificity	Severity	Boilerplate	Complexity	Specificity	Severity	Boilerplate	Complexity	Specificity
					gpt-4o-mini							
Mean	1.00	0.98	1.00	0.89	3.97	3.98	3.05	2.69	7.28	7.46	5.72	4.66
Median	1.00	1.00	1.00	1.00	4.00	4.00	3.00	3.00	7.50	7.50	5.50	5.00
Std. dev	0.00	0.13	0.01	0.31	0.19	0.18	0.28	0.48	0.48	0.44	0.52	0.60
Q1, Q3	[1.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[4.00, 4.00]	[4.00, 4.00]	[3.00, 3.00]	[2.00, 3.00]	[7.00, 7.50]	[7.00, 7.50]	[5.50, 6.00]	[4.00, 5.00]
					grok-3-mini							
Mean	0.96	0.22	0.70	0.64	3.71	3.88	3.07	2.82	6.68	7.23	5.64	4.92
Median	1.00	0.00	1.00	1.00	4.00	4.00	3.00	3.00	7.00	7.50	6.00	5.00
Std. dev	0.19	0.42	0.46	0.48	0.66	0.65	0.78	0.92	1.23	1.26	1.37	1.53
Q1, Q3	[1.00, 1.00]	[0.00, 0.00]	[0.00, 1.00]	[0.00, 1.00]	[3.00, 4.00]	[4.00, 4.00]	[3.00, 4.00]	[2.00, 4.00]	[5.50, 7.50]	[6.50, 8.00]	[5.00, 6.50]	[4.00, 6.00]
					phi4							
Mean	1.00	0.71	0.80	0.81	3.90	3.76	2.94	2.49	7.24	7.30	5.69	4.34
Median	1.00	1.00	1.00	1.00	4.00	4.00	3.00	2.00	7.50	8.00	6.00	4.00
Std. dev	0.06	0.46	0.40	0.39	0.32	0.45	0.35	0.55	0.78	1.21	1.01	0.79
Q1, Q3	[1.00, 1.00]	[0.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[4.00, 4.00]	[3.00, 4.00]	[3.00, 3.00]	[2.00, 3.00]	[7.50, 7.50]	[7.00, 8.00]	[6.00, 6.00]	[4.00, 5.00]
					llama3.3							
Mean	1.00	0.50	0.98	1.00	4.00	3.87	3.15	2.43	7.31	7.15	6.00	4.44
Median	1.00	0.00	1.00	1.00	4.00	4.00	3.00	2.00	7.50	7.50	6.00	4.00
Std. dev	0.02	0.50	0.14	0.03	0.20	0.65	0.50	0.69	0.78	1.53	1.00	1.24
Q1, Q3	[1.00, 1.00]	[0.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[4.00, 4.00]	[4.00, 4.00]	[3.00, 3.00]	[2.00, 3.00]	[6.50, 8.00]	[7.00, 8.00]	[5.50, 6.50]	[3.50, 5.50]

Panel B: Earnings call transcripts

	Evaluability (binary 0/1)				**Ordinal Rating** (1–5)				**Continuous Score** (0–10)			
Statistic	Severity	Boilerplate	Complexity	Specificity	Severity	Boilerplate	Complexity	Specificity	Severity	Boilerplate	Complexity	Specificity
					gpt-4o-mini							
Mean	1.00	1.00	1.00	1.00	2.54	3.05	3.33	3.82	4.62	5.17	5.74	7.12
Median	1.00	1.00	1.00	1.00	3.00	3.00	3.00	4.00	5.00	5.00	5.50	7.50
Std. dev.	0.07	0.05	0.07	0.07	0.71	0.45	0.56	0.57	1.14	0.83	0.85	1.12
[Q1, Q3]	[1.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[2.00, 3.00]	[3.00, 3.00]	[3.00, 4.00]	[4.00, 4.00]	[4.00, 5.50]	[5.00, 5.00]	[5.50, 6.50]	[7.50, 7.50]
					llama-3.3-70B							
Mean	0.98	1.00	1.00	0.98	2.10	3.14	4.15	4.15	3.03	5.19	7.80	7.32
Median	1.00	1.00	1.00	1.00	2.00	3.00	4.00	4.00	2.50	4.50	8.10	7.50
Std. dev.	0.14	0.04	0.04	0.14	0.67	0.82	0.67	0.87	1.08	1.74	1.21	1.51
[Q1, Q3]	[1.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[1.00, 1.00]	[2.00, 2.00]	[3.00, 4.00]	[4.00, 5.00]	[4.00, 5.00]	[2.50, 3.50]	[4.20, 6.50]	[7.50, 8.50]	[7.20, 8.20]

Table 4: **Generated Regressor Diagnostics: LLM Text Constructs**

This table diagnoses whether each LLM-generated text constructs (severity, boilerplate, complexity, and specificity) are contaminated by LLM-processing characteristics, omission (O_{im}) and injection (I_{im}) rates, when used as a regressor to investigate firm outcomes. Panels A1–A4 report results for each construct separately. Each row reports results for one firm outcome variable. β_{naive} is the coefficient on the construct in the naive baseline $y_{i,t} = \beta\,\text{Construct}_{i,t} + \gamma' X_{i,t} + \tau_t + \varepsilon_{i,t}$, with firm controls $X_{i,t} = \{\text{Size, MTB, CashFlow, Tangibility, Age}\}$ and time fixed effects τ_t. The joint specification adds the omission and injection rates: $y_{i,t} = \beta\,\text{Construct}_{i,t} + \rho_O\,O_{im,t} + \rho_I\,I_{im,t} + \gamma' X_{i,t} + \tau_t + \varepsilon_{i,t}$, yielding β_{joint}, ρ_O, ρ_I. $|\Delta\%| = 100 \times |\beta_{\text{joint}} - \beta_{\text{naive}}|/|\beta_{\text{naive}}|$ measures coefficient sensitivity. The decomposition splits the construct, via a first-stage Tobit regression of the construct on the wedges and controls, into the part predicted by the wedges $\widehat{\text{Construct}}$ and the orthogonal residual $\hat{e}$, then estimates $y_{i,t} = \theta\,\widehat{\text{Construct}}_{i,t} + \lambda\,\hat{e}_{i,t} + \gamma' X_{i,t} + \tau_t + \varepsilon_{i,t}$. Under the maintained assumption that O_{im} and I_{im} are LLM-processing artifacts orthogonal to firm fundamentals, λ is the bias-corrected coefficient on the construct. In panel A, we run these regression on the S&P 500 firms with annual 10-K Item 1A risk disclosures for the period 2005–2024 with Year fixed effects, and standard errors are CR1 cluster-robust on firm. In Panel B, we rerun the same test and report the results for the severity construct using regressors generated from earnings call transcripts for S&P 500 firms during our sample period. * $p < .10$, ** $p < .05$, *** $p < .01$.

Panel A: Item 1A risk disclosure text

		Coefficient on construct		Wedge coefficients			Decomposition	
Outcome	N	β_{naive}	β_{joint}	ρ_O	ρ_I	$\lvert\Delta\%\rvert$	θ	λ
				Panel A1: Severity				
NI/Assets	21,990	−0.0011 ∗ ∗	−0.0006	−0.0209 ∗ ∗∗	−0.0002	50.6%	−0.0042 ∗ ∗∗	−0.0006
CapEx/Assets	21,931	+0.0014 ∗ ∗	+0.0009	+0.0229 ∗ ∗∗	+0.0028	38.9%	+0.0043 ∗ ∗∗	+0.0009
R&D ratio	21,480	+0.1386 ∗ ∗∗	+0.1295 ∗ ∗∗	+0.2257 ∗ ∗∗	+0.2017 ∗ ∗∗	6.5%	+0.1897 ∗ ∗∗	+0.1295 ∗ ∗∗
Leverage	21,990	+0.0118 ∗ ∗	+0.0091∗	+0.1052∗	+0.0091	22.2%	+0.0254∗	+0.0091∗
Sales growth	21,899	+0.6190 ∗ ∗	+0.6970∗	−0.3368	+1.1494 ∗ ∗∗	12.6%	+0.1878	+0.6970∗
WW index	15,074	+0.0075 ∗ ∗	+0.0063	+0.0932	+0.0250	16.3%	+0.0131	+0.0063
				Panel A2: Boilerplate				
NI/Assets	22,054	+0.0023 ∗ ∗∗	+0.0027 ∗ ∗∗	−0.0164 ∗ ∗∗	−0.0034	14.9%	−0.0103 ∗ ∗	+0.0027 ∗ ∗∗
CapEx/Assets	21,928	−0.0044 ∗ ∗∗	−0.0035 ∗ ∗∗	+0.0131 ∗ ∗	+0.0002	20.1%	−0.0135 ∗ ∗∗	−0.0035 ∗ ∗∗
R&D ratio	21,526	−0.0661 ∗ ∗∗	−0.0751 ∗ ∗∗	+0.4837 ∗ ∗∗	+0.1070	13.5%	+0.3888 ∗ ∗∗	−0.0751 ∗ ∗∗
Leverage	22,054	+0.0203 ∗ ∗∗	+0.0163 ∗ ∗∗	+0.1442 ∗ ∗∗	+0.0729	19.4%	+0.1648 ∗ ∗∗	+0.0163 ∗ ∗∗
Sales growth	21,899	−0.2219 ∗ ∗	−0.1960 ∗ ∗	+0.5184	+0.7095 ∗ ∗∗	11.7%	−0.4926	−0.1960 ∗ ∗
WW index	15,121	−0.0236 ∗ ∗	−0.0280 ∗ ∗	+0.0951	+0.0991	18.9%	+0.0853∗	−0.0280 ∗ ∗
				Panel A3: Complexity				
NI/Assets	21,990	−0.0018 ∗ ∗∗	−0.0011	−0.0184 ∗ ∗∗	−0.0011	39.8%	−0.0040 ∗ ∗∗	−0.0011
CapEx/Assets	21,931	+0.0017 ∗ ∗	+0.0009	+0.0220 ∗ ∗∗	+0.0029	45.9%	+0.0043 ∗ ∗∗	+0.0009
R&D ratio	21,526	+0.1699 ∗ ∗∗	+0.1633 ∗ ∗∗	+0.1250	+0.1876 ∗ ∗∗	3.9%	+0.1903 ∗ ∗∗	+0.1633 ∗ ∗∗
Leverage	21,985	−0.0071	−0.0291 ∗ ∗∗	+0.2915 ∗ ∗∗	+0.0504	311.1%	+0.0646 ∗ ∗∗	−0.0291 ∗ ∗∗
Sales growth	21,899	+0.3053 ∗ ∗∗	+0.3600 ∗ ∗∗	−0.0310	+1.0361 ∗ ∗∗	17.9%	+0.1280	+0.3600 ∗ ∗∗
WW index	15,074	+0.0132 ∗ ∗	+0.0127∗	+0.0651	+0.0386	4.0%	+0.0149	+0.0127∗
				Panel A4: Specificity				
NI/Assets	22,054	−0.0019 ∗ ∗	−0.0018	−0.0137 ∗ ∗∗	−0.0042	7.3%	−0.0031	−0.0018
CapEx/Assets	21,995	+0.0016∗	+0.0014	+0.0168 ∗ ∗∗	+0.0048	11.6%	+0.0032	+0.0014
R&D ratio	21,462	+0.0651 ∗ ∗∗	+0.0562 ∗ ∗∗	+0.4398 ∗ ∗∗	+0.2141 ∗ ∗∗	13.7%	+0.2904 ∗ ∗∗	+0.0562 ∗ ∗∗
Leverage	21,985	+0.0221 ∗ ∗∗	+0.0107	+0.2011 ∗ ∗∗	+0.0788∗	51.4%	+0.0905 ∗ ∗∗	+0.0107
Sales growth	21,968	+0.2209	+0.2737∗	−0.4707	+0.8561 ∗ ∗	23.9%	−0.2409	+0.2737∗
WW index	15,121	+0.0153 ∗ ∗	+0.0199 ∗ ∗	+0.0653	+0.1105	30.1%	−0.0157	+0.0199 ∗ ∗

Panel B: Earnings call transcripts

		Coefficient on severity		Wedge coefficients			Decomposition	
Outcome	N	β_{naive}	β_{joint}	ρ_O	ρ_I	$\lvert\Delta\%\rvert$	θ	λ
NI/Assets	19,250	$+0.0005 * **$	$+0.0005 * **$	$-0.0098 * *$	-0.0008	4.6%	$+0.0058 * *$	$+0.0005 * **$
CapEx/Assets	19,241	$-0.0005 * *$	$-0.0005 * *$	$+0.0030$	$+0.0117 * *$	0.4%	-0.0010	$-0.0005 * *$
R&D ratio	19,250	$-0.0006*$	-0.0005	$+0.0508 * **$	$+0.0017$	18.0%	$-0.0286 * **$	-0.0005
Leverage	19,250	$+0.0257 * **$	$+0.0254 * **$	-0.1331	-0.1092	1.0%	$+0.0884$	$+0.0254 * **$
Sales growth	19,234	$-0.0531 * **$	$-0.0524 * **$	$+0.3160 * **$	$+0.2364 * **$	1.2%	$-0.2029 * **$	$-0.0524 * **$
WW index	19,250	$-0.0021 * **$	$-0.0021 * *$	$+0.0307$	-0.0339	2.9%	$-0.0354*$	$-0.0021 * *$

Table 5: **Generated Regressor Diagnostics: Jha et al. (2024) Replication**

This table diagnoses whether LLM regressors using in Jha et al. (2024) are contaminated by LLM-processing characteristics, omission (O_{im} and injection (I_{im}) rates, when used to investigate firm outcomes. Each row reports results for one firm outcome variable at time $t+2$ and outcomes are in percentage following the paper's method. β_{naive} is the coefficient on the generated regressor in their baseline regression: $y_{i,t+2} = \beta\,\text{Score}_{i,t} + \gamma' X_{i,t} + \alpha_i + \alpha_t + \varepsilon$, with $X_{i,t} = \{\text{Size, MTB, CashFlow, Leverage, } y_{i,t}\}$, firm fixed effects α_i, and year-quarter fixed effects α_t. The joint specification adds the omission and injection rates: $y_{i,t+2} = \beta\,\text{Score}_{i,t} + \rho_O\,O_{im,t} + \rho_I\,I_{im,t} + \gamma' X_{i,t} + \alpha_i + \alpha_t + \varepsilon$, yielding β_{joint}, ρ_O, ρ_I. $|\Delta\%| = 100 \times |\beta_{\text{joint}} - \beta_{\text{naive}}|/|\beta_{\text{naive}}|$ measures coefficient sensitivity. In the decomposition, we regress the *Score* on the wedges and controls to obtain the part predicted by the wedges $\widehat{\text{Score}}$ and the orthogonal residual $\hat{e}$, then estimate $y_{i,t+2} = \theta\,\widehat{\text{Score}}_{i,t} + \lambda\,\hat{e}_{i,t} + \gamma' X_{i,t} + \alpha_i + \alpha_t + \varepsilon$. Under the maintained assumption that O_{im} and I_{im} are LLM-processing artifacts orthogonal to firm fundamentals, λ is the bias-corrected coefficient on the direct predictor. We replicate Jha et al. (2024) analyses using S&P 500 firms with quarterly earnings-call transcripts for the period 2006–2024 with firm + year-quarter fixed effects and the standard errors are cluster-robust to firm. * $p < .10$, ** $p < .05$, *** $p < .01$.

			Coefficient on direct		Wedge coefficients			Decomposition	
Predictor	Outcome $y_{i,t+2}$	N	β_{naive}	β_{joint}	ρ_O	ρ_I	$\lvert\Delta\%\rvert$	θ	λ
			Panel A: gpt-4o-mini						
capex	CapExRatio	18,489	+0.24 * **	+0.25 * **	−0.13	+0.50 * **	0.5%	−0.84	+0.25 * **
cash	CashRatio	2,350	+1.44 * **	+1.45 * **	+0.79	+3.46	0.5%	−3.75	+1.45 * **
div	DivQuarterly	13,361	+0.05 * **	+0.05 * **	+0.19	+0.29	0.3%	+0.89	+0.05 * **
lev	Leverage	15,827	+0.82 * **	+0.82 * **	−1.00	−0.61	0.3%	−2.09	+0.82 * **
ocf	OpCFRatio	19,095	+0.38 * **	+0.38 * **	+0.58	−1.44*	0.8%	+6.10	+0.38 * **
rd	RDRatio_TC	16,959	+0.03	+0.03	+0.44	−0.10	1.6%	+0.20	+0.03
sale	SalesGrowth	19,220	+20.72 * **	+20.73 * **	−1.97	+1.17	0.1%	+14.34	+20.73 * **
			Panel B: llama-3.3-70B						
capex	CapExRatio	17,069	+0.15 * **	+0.15 * **	+0.23	+0.20	0.0%	+0.13	+0.15 * **
cash	CashRatio	15,181	+0.48 * **	+0.48 * **	+3.83	+2.81	0.1%	+4.07	+0.48 * **
div	DivQuarterly	10,595	+0.01	+0.01	+0.04	+0.54 * *	14.5%	−1.07	+0.01
lev	Leverage	14,424	+1.32 * **	+1.32 * **	−0.77	+0.33	0.1%	−2.26	+1.32 * **
ocf	OpCFRatio	18,469	+0.28 * **	+0.28 * **	+0.99	−0.84	1.5%	+4.19	+0.28 * **
rd	RDRatio_TC	14,334	+0.04*	+0.04*	+0.11	−0.05	0.7%	+0.24	+0.04*
sale	SalesGrowth	19,219	+17.87 * **	+17.92 * **	−7.59	+5.09	0.3%	+3.10	+17.92 * **

A Appendix: Additional Figures and Tables

Figure A.1: **Vocabulary Omission: Granular Risk Factor Categories**

In this figure, we plot the Omission rates for discriminatory terms identified for each of the 26 risk categories in our sample. Omission is defined as $(Vocab_{\text{Human}} - Vocab_{\text{LLM}} \cap Vocab_{\text{Human}})/Vocab_{\text{Human}}$. We define Human and LLM vocabularies as the count of unique discriminative terms used by companies in their Item 1A and LLM summaries, respectively.

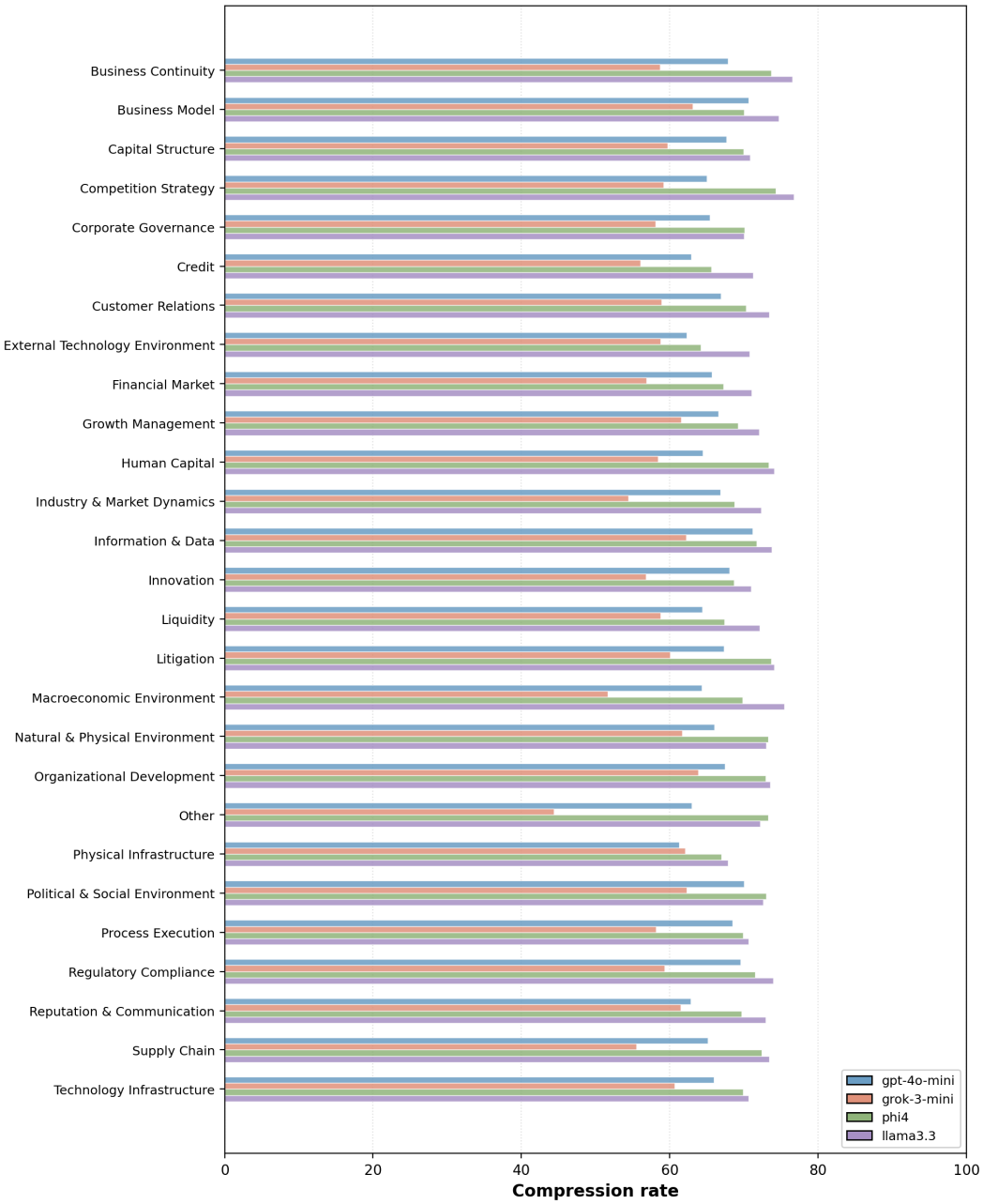


Figure A.2: **Injection of Vocabulary: Granular Risk Factor Categories**

In this figure, we plot the injection rates for discriminatory terms identified for each of the 26 risk categories in our sample. Injection is defined as $(Vocab_{\text{LLM}} - Vocab_{\text{LLM}} \cap Vocab_{\text{Human}})/Vocab_{\text{LLM}}$. We define Human and LLM vocabularies as the count of unique discriminative terms used by companies in their Item 1A and LLM summaries, respectively.

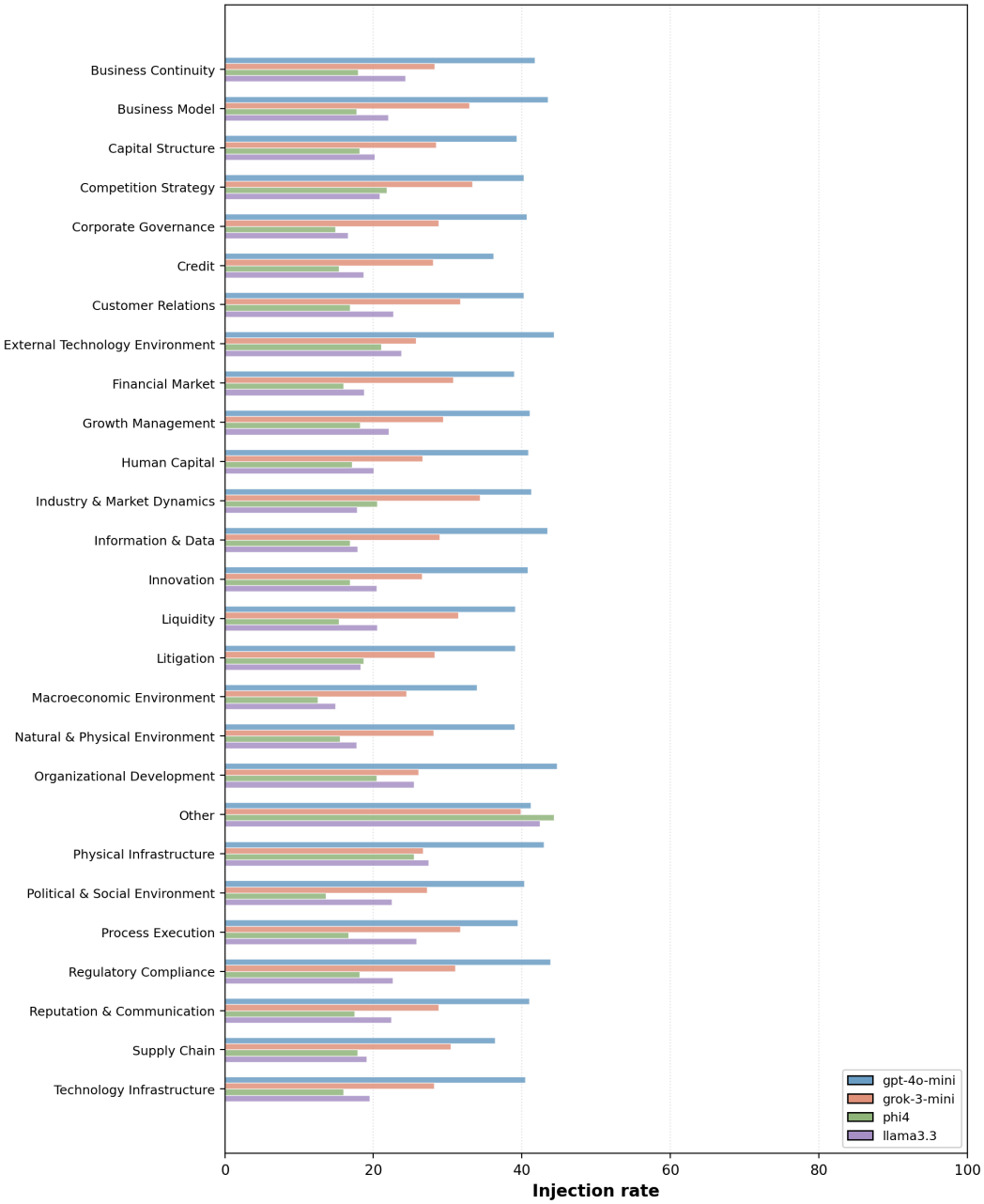


Figure A.3: **LLM-generated Dictionary Terms Agreement By Risk Factor Category**

In this figure, we plot the pairwise agreement rates between LLM-generated dictionaries across 26 risk factor categories. For each category, we compute the share of terms in a base model's dictionary that appear in each comparison model's dictionary.

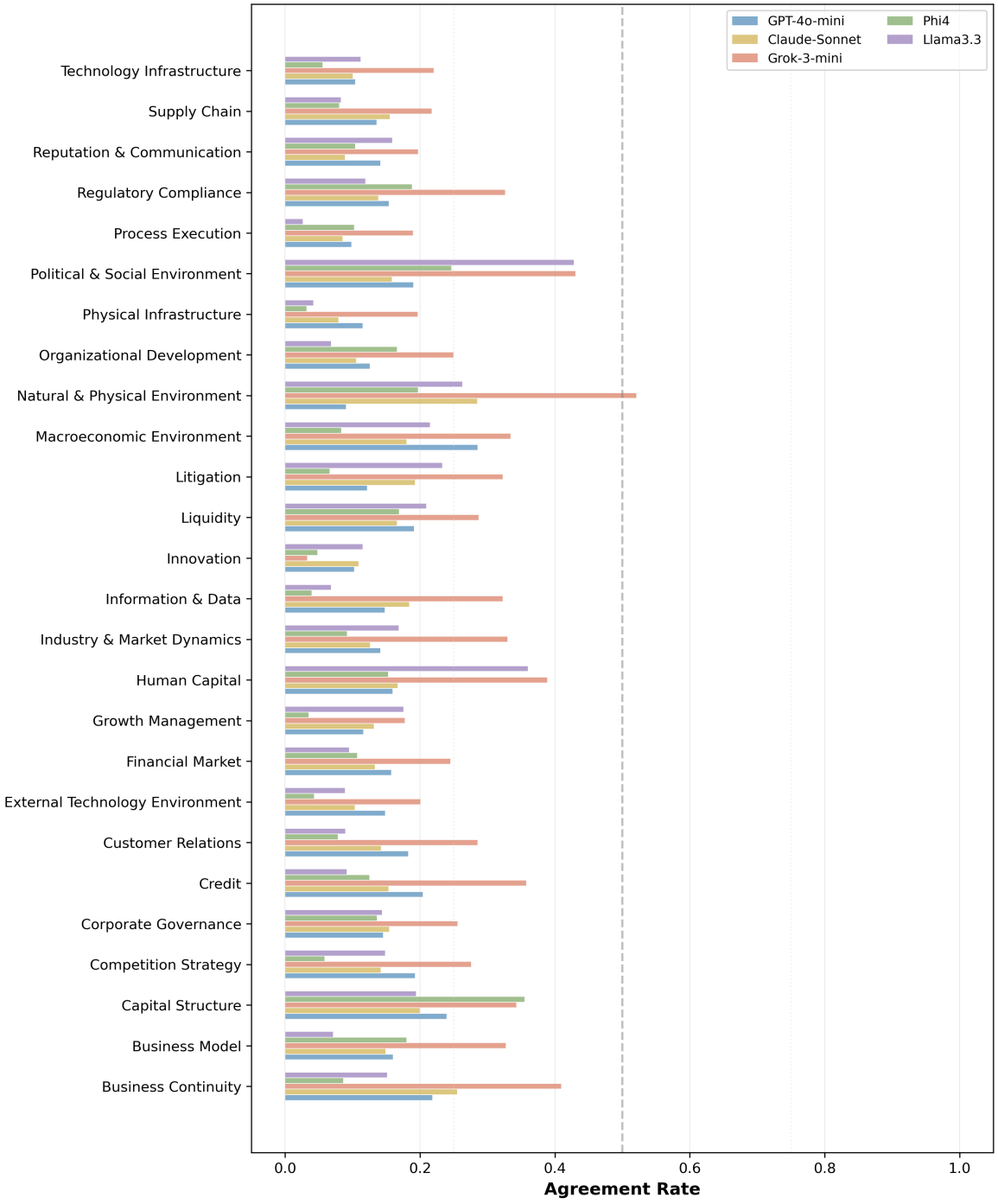
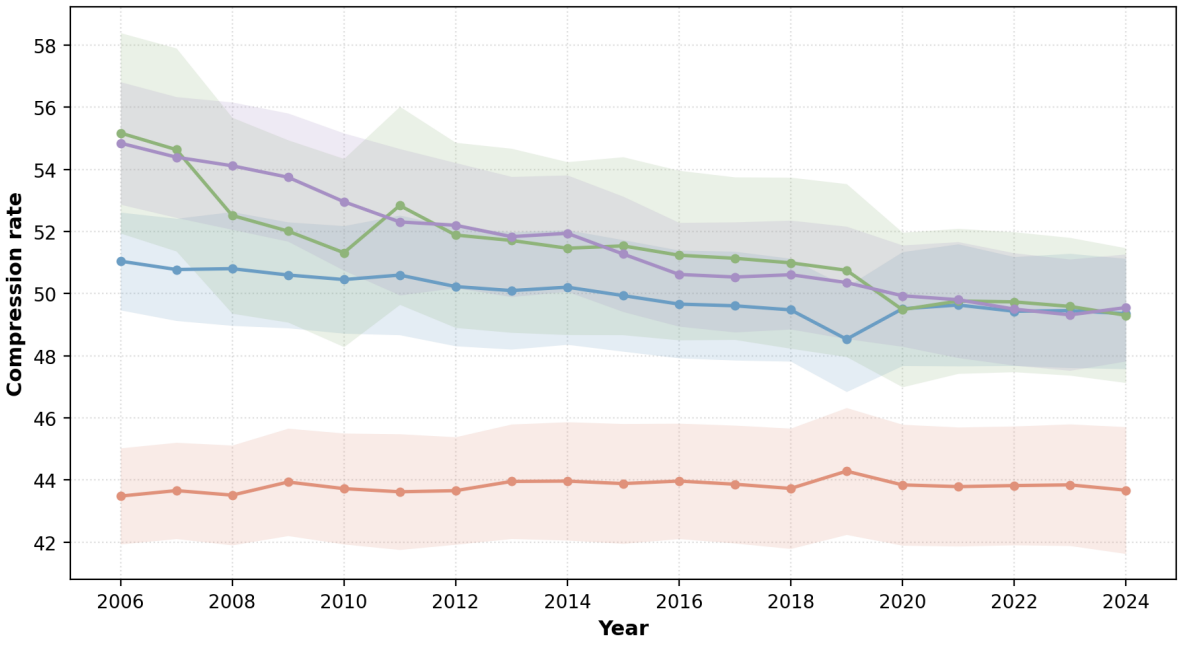


Figure A.4: **Omission and Injection Rates over the Sample Period**

In panels (a) and (b), we plot the Omission and Injection rates for discriminatory terms identified in our S&P 500 sample across all risk categories separately for each model. Omission is defined as $O_{im} = 1 - \frac{|V_{i,\text{source}} \cap V_{im,\text{summary}}|}{V_{i,\text{source}}}$. Injection is defined as $I_{im} = 1 - \frac{|V_{im,\text{summary}} \cap V_{i,\text{source}}|}{V_{im,\text{summary}}}$. We define Human and LLM vocabularies as the count of unique discriminative terms used by companies in their Item 1A and LLM summaries, respectively.

(a) Omission rate



(b) Injection rate

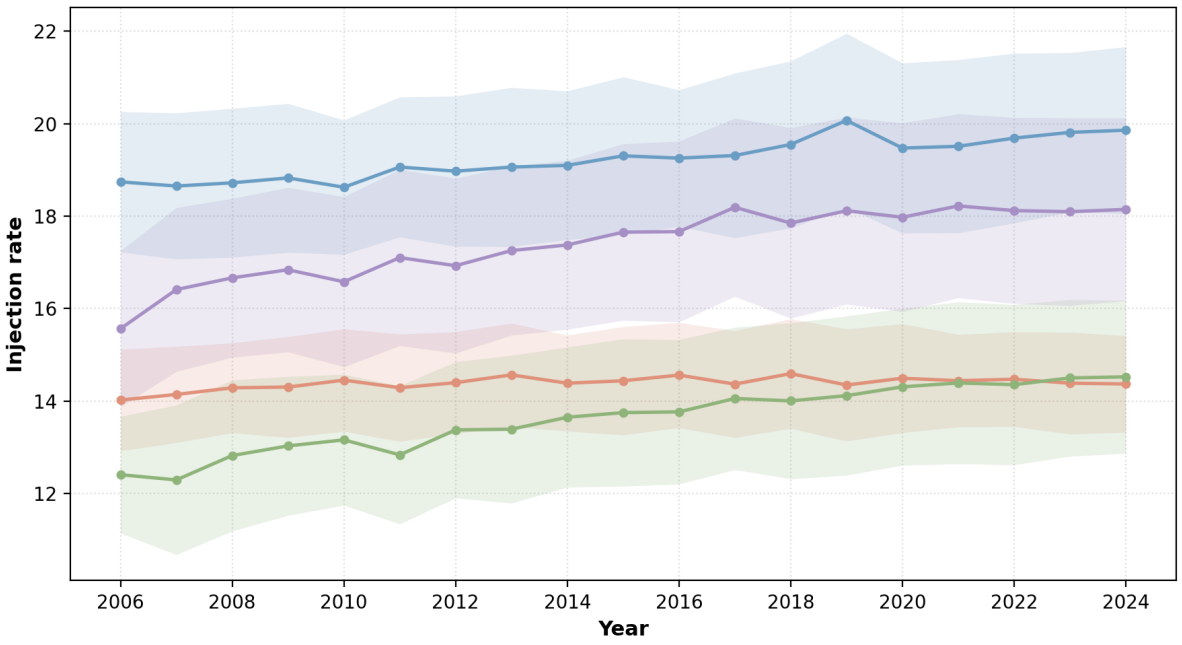
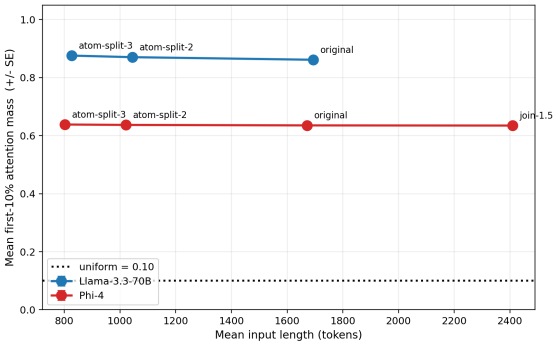
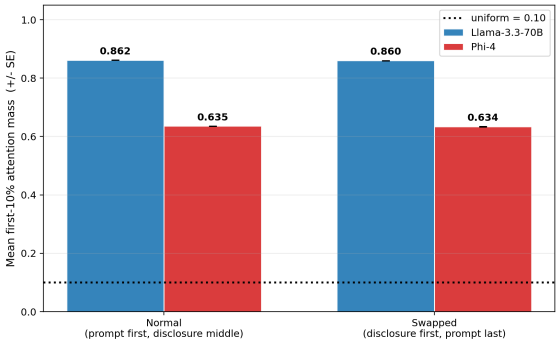


Figure A.5: **Robustness of the Attention Bottleneck**

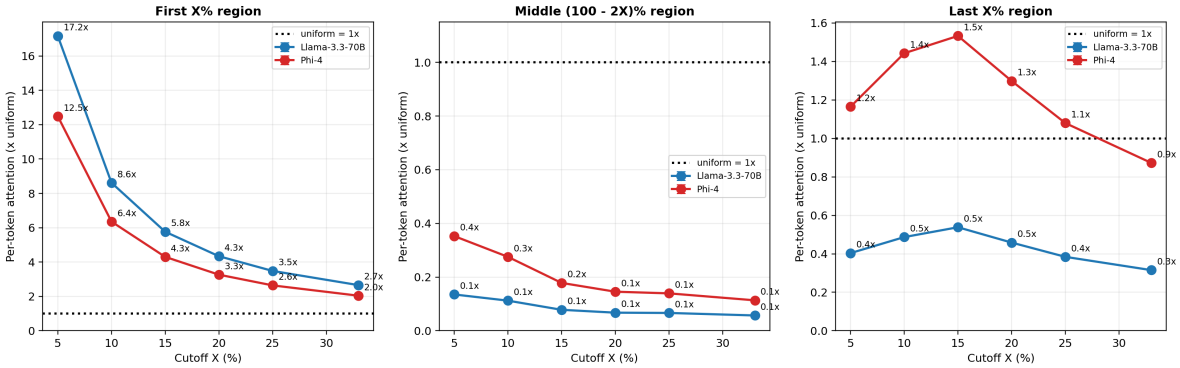
The figure assesses the robustness of the first-10% attention bottleneck reported in Figure 2 to three natural alternative explanations: input length, the position of the user's instructions within the prompt, and the choice of region boundary. All three panels use the score prompt with the severity construct on the same 500-chunk subsample of Item 1A disclosures, run on Llama-3.3-70B and Phi-4; markers, bars, and error bars are means and ± 1 standard error across the 500 chunks; dotted lines mark the values implied by uniform attention. Panel (a) plots mean first-10% attention mass against mean input length. Size variants are constructed from each base chunk's underlying sentence-level "atoms": atom-split-3 (one third of the original text, ≈ 800 tokens), atom-split-2 (one half, ≈ 1,000 tokens), original (≈ 1,700 tokens), and, for Phi-4 only, join-1.5 (adjacent same-firm chunks concatenated, ≈ 2,400 tokens; Llama-3.3-70B's 80-layer attention tensor exceeds 95 GB and does not fit in single-GPU memory). Panel (b) compares the first-10% attention mass under the standard score prompt (system message and user instructions first, disclosure body in the middle) against a swapped variant in which the disclosure body appears first and the user's instructions are appended at the end of the input. Panel (c) re-expresses the per-token attention measurement of Figure 2 for region boundaries ranging from 5% to 33% of the input. The input is divided into twenty 5% bins; for each cutoff X we report the per-token concentration multiplier (mean attention mass divided by the region's token share, so that 1× corresponds to uniform attention) in the first-X%, middle-$(100-2X)$%, and last-X% regions. Across all three panels, first-10% attention mass is essentially unchanged from its Figure 2 value, indicating that the bottleneck is not driven by input length, by the position of the user's instructions within the input, or by the choice of the 10% region boundary.



(a) Input Length Sensitivity



(b) Prompt-Position Swap



(c) Region Cutoff Sensitivity

Figure A.6: **Tone Distribution**

This figure shows the average number words in each of the seven Loughran-McDonald tone dimensions per disclosure, broken out by vocabulary object: Source (black, is the risk sentences from Item 1A), Compressed (red, words dropped by LLM), Retained (green, words retained by LLM), Injected (blue, words added by LLM), Summary (gray, is the LLM summary). Four panels, one per LLM.

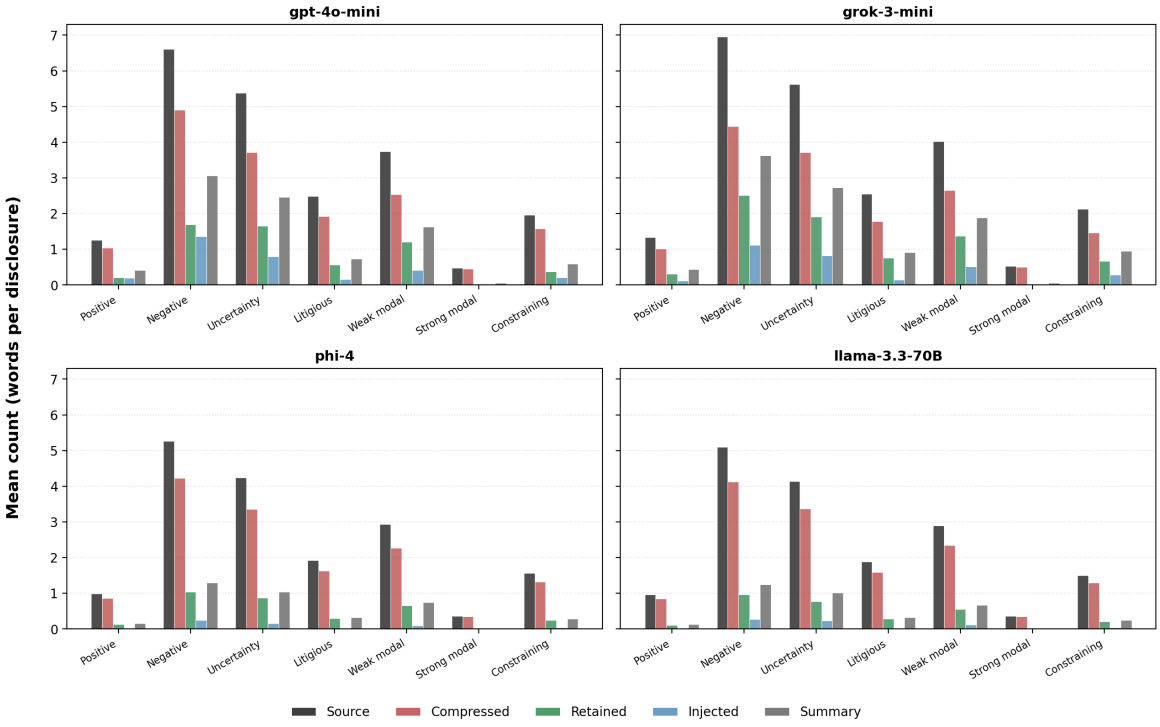


Figure A.7: **LLM Prompt for Risk Factor Summaries and Tasks in Item 1A**

This figure reproduces the prompt issued to each large language model (GPT-4o-mini, Grok-3-mini, Llama-3.3-70B, and Phi-4) to produce several tasks ranging from summary descriptions to risk factor labels, applied to individual Item 1A risk-factor disclosures.

```
instructions = """
You are a corporate risk manager reviewing this item 1A section from
a company's 10-K disclosure. You do not know of future events beyond
the content in this document. Do not use any information, terminology,
or market context beyond the provided financial disclosure and
disclosure date. Analyze the disclosure and present the following:

1. Identify all risk factors and, if available, the associated
mitigation plans to address the identified risks **explicitly
discussed** by the company in the disclosure.
Do not infer or introduce unstated risks.

2. For each risk factor, present the analysis in JSON format
with the following structure:

{
 "risk_factors": [
 {
  "risk_factor": "Bi-gram Label",
  "risk_factor_syn": "Bi-gram synonym of Risk Factor label",
  "risk_severity": "High/Medium/Low/Not Specified",
  "risk_desc": "100-150 word description based ONLY on given text",
  "risk_sentences": "All exact sentences from text about this risk",
  "risk_keyword": "Keyword indicating this risk",
  "mitigation": "Bi-gram Label or 'Not Disclosed'",
  "mitigation_syn": "Bi-gram synonym OR 'Not Disclosed'",
  "mitigation_desc": "100-150 word description
  OR 'No plan discussed'",
  "mitigation_sentences": "All related sentences OR 'None'",
  "mitigation_keyword": "Keyword for mitigation OR 'None'",
  "bus_function": "One from: |'Research & Development'|'Operations'|
  'Sales & Marketing'|'Customer Experience'|'Financing'|
  'Capital Investment'|'Human Capital Management'|
  'Information Technology'|'Supply Chain Management'|
  'Strategy \& Planning'|'Risk & Compliance'"
 }
 ]
}
Important Notes:
- Explicit Information Only
- Bi-gram Labels (e.g., "Market Competition", "Supply Chain")
- Time-Appropriate Language
- Valid JSON format required
"""
```

Figure A.8: **LLM Scoring Prompt for Risk-Disclosure Constructs**

This figure reproduces the scoring prompt issued to each large language model (GPT-4o-mini, Grok-3-mini, Llama-3.3-70B, and Phi-4) used to score Item 1A risk-factor disclosures along the four constructs studied in the paper. Panel A presents the task instructions, the closed lists of evaluable constructs and business functions, and the ordinal and continuous scoring rubrics. Panel B presents the required JSON output schema.

Panel A: Task Instructions, Business Functions, and Constructs

```
[frame=single,framesep=3mm,xleftmargin=2mm,xrightmargin=2mm]
      Evaluate each of the following {n} risk disclosures on four constructs.
      For EACH disclosure, provide:

      PART A: EVALUABILITY
      Which constructs have CLEAR indicators? Select all that apply from:
      Severity, Complexity, Specificity, Boilerplate

      PART B: BUSINESS FUNCTIONS
      Which business functions are explicitly affected? Select all that apply from:
      Research & Development, Operations, Sales & Marketing, Customer Experience,
      Financing, Capital Budgeting, Human Capital Management, Information Technology,
      Supply Chain Management, Strategy & Planning, Risk & Compliance

      PART C: SCORING
      - Ordinal: 1-5 scale (integers only)
      Severity:    1 = negligible       to 5 = catastrophic
      Complexity:  1 = straightforward  to 5 = intractable
      Specificity: 1 = generic          to 5 = unique
      Boilerplate: 1 = entirely original to 5 = pure boilerplate
      - Continuous: 0.0-10.0 scale (decimals allowed)

      CONSTRUCTS:
      1. SEVERITY -- Potential impact if risk materializes
      Ordinal:    1=negligible, 2=minor, 3=moderate, 4=major, 5=catastrophic
      Continuous: 0=no impact,  10=existential threat

      2. COMPLEXITY -- Difficulty of assessing/interpreting this risk
      Ordinal:    1=straightforward, 2=somewhat complex, 3=complex,
      4=highly complex,  5=intractable
      Continuous: 0=trivially simple, 10=impossible to evaluate

      3. SPECIFICITY -- How unique is this risk to this firm
      Ordinal:    1=generic/any firm, 2=industry-wide, 3=sector-specific,
      4=firm-specific,    5=unique
      Continuous: 0=universal risk, 10=completely unique to this firm

      4. BOILERPLATE -- Degree of template/standard language used
      Ordinal:    1=entirely original, 2=mostly original, 3=mixed,
      4=mostly template,   5=pure boilerplate
      Continuous: 0=fully novel, 10=entirely standard language
```

Panel B: Output Format and Required JSON Schema

```
[frame=single,framesep=3mm,xleftmargin=2mm,xrightmargin=2mm]
      CRITICAL INSTRUCTIONS:
      - Return ONLY a valid JSON array with exactly {n} objects, one per
      disclosure in order
      - Each object must have: evaluable, functions, ordinal, continuous
      - Use pipe-delimited strings for evaluable and functions
      - Ordinal scores: integers 1-5. Continuous scores: decimals 0.0-10.0.

      REQUIRED FORMAT (array of {n} objects):
      [
      {
       "evaluable": "Severity|Complexity",
       "functions": "Operations|Risk & Compliance",
       "ordinal":    {"Severity": 4,   "Complexity": 3,
        "Specificity": 2, "Boilerplate": 4},
       "continuous": {"Severity": 7.5, "Complexity": 6.0,
        "Specificity": 4.0, "Boilerplate": 8.0}
      },
      ...
      ]

      ---
      {disclosures}
```

Table A.1: Risk Factor Taxonomy: Detailed

Main Category	**Sub-category**	**Key Elements**
External Environmental	External Technology Environment	Technological disruption, Industry-wide digital transformation, Innovation pace, Tech adoption trends, Emerging technologies, Digital transformation
	Industry and Market Dynamics	Industry-wide consolidation and price trends, Shifts in Industry-wide supply and demand, New market entrants, Competitive intensity, Product market fluidity
	Macroeconomic Environment	Macroeconomic conditions, Economic growth/recession, Interest rates/inflation, Currency fluctuations, Trade conditions, Market liquidity
	Natural/Physical Environment	Natural disasters, Climate change impacts, Weather-related disruptions, Resource scarcity, Environmental degradation
	Political/Social Environment	Government stability, Regulatory changes, Social/demographic shifts, Geopolitical tensions, Public sentiment
Financial	Capital Structure	Leverage levels, Funding mix, Dividend policy, Investment capacity, Solvency, Tax Management, Bankruptcy
	Credit Risk	Counterparty default, Customer credit, Collateral value, Recovery rates, Borrower concentration
	Liquidity Risk	Funding access, Cash flow management, Asset liquidity, Working capital, Emergency funding, Sinking fund, Treasury operations, Capital market access
	Financial Market Risk	Interest rate exposure, Foreign exchange, Asset valuations, Commodity prices, Hedging risks, Value at Risk Analysis, Investment Portfolio management
Operational	Business Continuity	Disaster recovery, Crisis management, Backup systems, Emergency response, Business restoration, Continuity planning, Critical function maintenance
	Human Capital	Talent management, Labor relations, Skills development, Talent acquisition, Performance management, Employee engagement
	Information & Data Risks	Cybersecurity, Data privacy, Information protection, System access, Cloud security, Network security, Third-party data handling
	Organizational Development	Succession planning, Leadership development, Organizational culture
	Physical Infrastructure	Physical assets, Equipment reliability, Facility management, Workplace safety
	Process Execution	Quality control, Efficiency metrics, Capacity utilization, Process integration, Standard operating procedures, Operational efficiency, Transaction processing, Project delivery, Timeline management
	Supply Chain	Supplier management, Logistics network, Inventory control, Distribution systems, Supply chain resilience, Strategic supply chain partnerships, Supplier concentration, Supplier service quality
	Technology Infrastructure	System reliability, Company-specific tech capabilities, System architecture, IT strategy, Technology obsolescence, Digital capabilities, System integration, Enterprise systems
Regulatory and Compliance	Corporate Governance	Board oversight, Internal controls, Ethics/conduct, Decision rights, Accountability structure, Audit risk
	Litigation	Contract compliance, Legal proceedings, Dispute resolution, Liability exposure, Intellectual property
	Regulatory Compliance	Regulations, Compliance standards, Operating licenses, Reporting requirements, Regulatory changes, Environmental compliance, Labor regulations, Capital adequacy requirements
	Reputation & Communication	Media relations, Stakeholder management, Corporate social responsibility, Public relations, Social media risks, Corporate reputation, Community relations, Crisis communication
Strategic	Business Model	Value proposition, Revenue streams, Cost structure, Resource allocation, Business model disruption, Strategic agility, Adaptation capability, Development costs, Strategic initiatives
	Competition Strategy	Firm-level market share, Product differentiation, Firm-specific pricing power, Price-cost margin, Brand value, Product adoption, Competitive advantage, Strategic positioning
	Customer Relations	Customer satisfaction, Customer retention, Service quality, Digital engagement, Customer concentration, Customer trust, Client relations, Customer service
	Growth management	Organic growth, Market development, Market expansion, M&A execution, Post-merger integration, Strategic partnerships other than supply chain
	Innovation Risks	R&D effectiveness, Innovation pipeline, Development timing, Product testing, Patent development, IP protection, Research rights, Technical feasibility

Table A.2: **Generated Regressor Diagnostics: Summary For LLM Classification**

This table reports results from a two-stage LLM wedge diagnostic approach to assess potential bias in LLM-generated risk factor category classifications. Panel A reports the Omission rate, Injection rate, and months since training coefficients from the first-stage multinomial logit regressions after controlling for year fixed effects. We decompose the 26 risk factor category indicators into corresponding parts predicted by the LLM characteristics (Omission, Injection, and training) in the first stage, and orthogonal residuals. In the second stage, we use these predicted and residual variables as regressors along with firm-level controls (log assets, leverage, cash/assets, market-to-book, Operating income/assets) and year fixed effects. Panels B and C report the coefficients on the LLM predicted part and residual part of the category identifier in second-stage regressions, respectively. The last row in Panel A reports p-values for the joint test that Omission rate, Injection rate, and months since training can be excluded from the first stage. The Hausman tests in Panel B test whether the LLM characteristic predicted part of the category identifier has a different effect on y than the residual. * $p < .10$, ** $p < .05$, *** $p < .01$.

Panel A: First Stage (Compression → LLM Classification)

	Compression	Injection	Months Since Cutoff	Joint χ^2 (df=3)	p-val
			Operational Risks		
Business Continuity	-	-	-	-	-
Human Capital	-1.250***	0.627***	0.021***	4001.18	0.000
Information & Data	-0.686***	0.277***	0.027***	1916.30	0.000
Organizational Development	-1.194***	1.475***	0.032***	1990.50	0.000
Physical Infrastructure	-1.919***	1.527***	-0.059***	5543.00	0.000
Process Execution	-2.265***	2.093***	-0.019***	12200.57	0.000
Supply Chain	-1.694***	0.705***	0.031***	7220.85	0.000
Technology Infrastructure	-1.443***	0.372***	-0.012***	3373.22	0.000
			External Environmental Risks		
External Technology Environment	-1.964***	0.717***	0.013***	3271.07	0.000
Industry & Market Dynamics	-1.475***	0.525***	0.008***	5188.01	0.000
Macroeconomic Environment	-1.465***	-0.594***	0.071***	12315.01	0.000
Natural & Physical Environment	-0.986***	0.165***	0.012***	1723.15	0.000
Political & Social Environment	-0.729***	0.401***	-0.017***	1657.36	0.000
			Financial Risks		
Capital Structure	-1.471***	0.793***	-0.023***	7368.56	0.000
Credit	-2.139***	0.312***	0.038***	8673.80	0.000
Financial Market	-1.729***	0.256***	0.040***	8441.14	0.000
Liquidity	-1.568***	0.544***	0.059***	7200.38	0.000
			Strategic Risks		
Business Model	-0.734***	0.789***	0.034***	3802.13	0.000
Competition Strategy	-1.375***	1.500***	0.023***	8197.95	0.000
Innovation	-2.108***	1.280***	0.031***	7538.75	0.000
Customer Relations	-1.824***	1.498***	0.019***	7896.52	0.000
Growth Management	-1.669***	1.465***	-0.068***	13780.18	0.000
			Regulatory & Compliance Risks		
Corporate Governance	-1.890***	0.572***	-0.003***	6314.55	0.000
Litigation	-1.099***	0.700***	0.027***	3902.89	0.000
Regulatory Compliance	-1.051***	1.132***	0.017***	7204.47	0.000
Reputation & Communication	-1.378***	0.619***	0.028***	3162.17	0.000
Joint Wald (comp = inj = months = 0, df = 75)				110000	0.000

Panel B: Second Stage LLM Wedge Predicted Coefficients

Predicted	Outcome Variables NI/Assets	Sales/Assets	R&D/Assets	Analyst Disp.
	Operational Risks			
Business Continuity	-	-	-	-
Human Capital	0.623***	42.579***	1.669***	-25.004***
Information & Data	0.880***	3.581***	0.132***	-0.711**
Organizational Development	-8.505***	-77.215***	4.609***	22.261***
Physical Infrastructure	-5.855***	-7.760***	2.001***	0.515
Process Execution	-1.815***	-6.525***	2.025***	0.358
Supply Chain	-0.092*	12.294***	1.497***	0.215
Technology Infrastructure	-0.800***	-0.660	0.254***	4.226***
	External Environmental Risks			
External Technology Environment	12.430***	48.375***	1.457***	0.377
Industry & Market Dynamics	1.440***	2.129***	-0.647***	-16.167***
Macroeconomic Environment	-0.165***	11.822***	1.609***	-6.373***
Natural & Physical Environment	-4.291***	-19.332***	1.136***	6.673***
Political & Social Environment	-1.415***	20.580***	5.454***	-5.680***
	Financial Risks			
Capital Structure	0.212***	2.640***	0.521***	2.982***
Credit	0.262***	-2.902***	1.105***	-3.096***
Financial Market	-1.964***	-13.889***	1.058***	10.095***
Liquidity	-3.299***	10.919***	1.527***	1.916***
	Strategic Risks			
Business Model	-5.217***	30.962***	6.638***	-9.193***
Competition Strategy	2.423***	26.104***	-0.641***	-16.754***
Innovation	-1.607***	3.739***	2.053***	-18.111***
Customer Relations	0.003	-2.898***	0.839***	6.606***
Growth Management	1.523***	5.532***	0.865***	-7.845***
	Regulatory & Compliance Risks			
Corporate Governance	-7.375***	19.841***	5.595***	-7.379***
Litigation	8.320***	-46.164***	-3.790***	-18.444***
Regulatory Compliance	-1.648***	-2.575***	1.289***	5.184***
Reputation & Communication	5.363***	29.746***	2.719***	-12.517***
Hausman F ($\beta_k = \gamma_k$)	8730.92***	7532.11***	5773.80***	1476.87***
Hausman F (p-val)	0.000	0.000	0.000	0.000
N	2,862,806	2,862,806	2,862,806	2,738,037

Panel C: Second Stage LLM Wedge Residual Coefficients

Residual	Outcome Variables NI/Assets	Sales/Assets	R&D/Assets	Analyst Disp.
		Operational Risks		
Business Continuity	-	-	-	-
Human Capital	0.003***	0.049***	-0.004***	-0.033***
Information & Data	0.004***	-0.018***	0.000	-0.050***
Organizational Development	-0.002*	0.004	0.002***	-0.048***
Physical Infrastructure	-0.003***	-0.075***	-0.009***	0.074***
Process Execution	0.001*	-0.028***	0.001	-0.038***
Supply Chain	0.002***	0.114***	0.005***	-0.018***
Technology Infrastructure	0.005***	0.044***	-0.004***	-0.055***
		External Environmental Risks		
External Technology Environment	0.001	-0.099***	0.003***	-0.031***
Industry & Market Dynamics	0.003***	0.041***	-0.000	-0.008***
Macroeconomic Environment	0.006***	0.014***	-0.005***	-0.043***
Natural & Physical Environment	0.003***	-0.017***	-0.005***	-0.000
Political & Social Environment	0.004***	0.023***	-0.003***	-0.022***
		Financial Risks		
Capital Structure	-0.002***	-0.056***	-0.004***	0.005**
Credit	0.006***	-0.138***	-0.010***	-0.055***
Financial Market	0.003***	-0.075***	-0.004***	-0.039***
Liquidity	-0.001**	-0.089***	-0.007***	0.001
		Strategic Risks		
Business Model	0.000	-0.013***	0.003***	-0.011***
Competition Strategy	0.005***	0.021***	-0.001***	-0.031***
Innovation	0.002***	-0.072***	0.016***	-0.049***
Customer Relations	0.003***	0.063***	-0.000	-0.052***
Growth Management	0.004***	0.044***	-0.005***	-0.057***
		Regulatory & Compliance Risks		
Corporate Governance	0.003***	-0.050***	-0.006***	-0.010***
Litigation	0.002***	0.015***	0.003***	-0.027***
Regulatory Compliance	0.002***	-0.045***	-0.002***	-0.018***
Reputation & Communication	0.005***	0.027***	-0.007***	-0.045***

Table A.3: Jha et al. (2024) Primary Results Replication

This table replicates the three headline regressions of Jha et al. (2024): the firm-quarter ChatGPT Investment Score forecasts capital expenditure at t+2 (their Table 3 col. 4), capital expenditure at t+4 (Table 4 col. 2), and PT-scaled R&D at t+2 (Table 5 col. 8). The regression specification follows their equation (2): $y_{i,t+h} = \beta\,\text{ChatGPT Investment Score}_{i,t} + \beta_2\,\text{Total } q_{i,t} + \beta_3\,y_{i,t} + \gamma' X_{i,t} + \alpha_i + \alpha_t + \varepsilon_{i,t}$, with the control set $X_{i,t}$ = {Size, MTB (proxy for PT Total q), CashFlow, Leverage}, lagged $y_{i,t}$, firm fixed effects α_i, and year-quarter fixed effects α_t. Standard errors are clustered on firm. The predictor is the −1 to +1 ChatGPT Investment Score (page 14 of J-W), averaged across earnings-call chunks. Outcomes are in percent (CapEx/Assets and PT-scaled R&D as pp), matching their reporting convention. The standardized beta $\beta \cdot \text{sd}(X)/\text{sd}(Y)$ is the unit-free magnitude they report in their text. Jha et al. (2024) used `gpt-3.5-turbo`; the replication uses `gpt-4o-mini` and `llama-3.3-70B`. We use the S&P 500 firms with quarterly earnings-call transcripts for the period 2006–2024 in this analyses. * $p < .10$, ** $p < .05$, *** $p < .01$.

	gpt-4o-mini					llama-3.3-70B				
Source	coef	std β	t	R^2	N	coef	std β	t	R^2	N
Panel A: $CapEx_{i,t+2}$ (Table 3 col. 4)										
Jha et al. (2024)	+0.4770 * **	0.055–0.079	13.91	0.575	74,429	+0.4770 * **	0.055–0.079	13.91	0.575	74,429
Replication	+0.2441 * **	+0.0677 * **	10.78	0.804	18,487	+0.1502 * **	+0.0594 * **	9.98	0.802	17,066
Panel B: $CapEx_{i,t+4}$ (Table 4 col. 2)										
Jha et al. (2024)	+0.4740 * **	0.060–0.080	13.99	0.543	73,598	+0.4740 * **	0.060–0.080	13.99	0.543	73,598
Replication	+0.2792 * **	+0.0781 * **	10.05	0.797	18,385	+0.1894 * **	+0.0756 * **	9.74	0.791	16,972
Panel C: $R\&D_{i,t+2}$ (PT-scaled) (Table 5 col. 8)										
Jha et al. (2024)	+0.1350 * **	0.020–0.025	5.37	0.905	34,253	+0.1350 * **	0.020–0.025	5.37	0.905	34,253
Replication	+0.0293	+0.0044	0.91	0.778	16,956	+0.0444*	+0.0081*	1.95	0.788	14,329

Table A.4: **Variable Definitions**

Variable	Definition / Formula	Source
Vocabulary descriptives (Table 1; computed per text source × model)		
Total words	Total word count in the corpus	Authors' calculation
Vocabulary size	Number of distinct words appearing at least once in the corpus	Authors' calculation
Top-N concentration	Percentage of total words accounted for by the N most frequent words: *Top-N* $= \left(\sum_{w \in \text{top-}N} c(w)\right)/T \times 100\%$. where $c(w)$ is the count of word w. We report $N \in \{100, 500, 1000\}$.	Authors' calculation
Entropy (normalized Shannon)	$H_{\text{norm}} = \left(-\sum_w p_w \log_2 p_w\right) \Big/ \log_2 V$, where $p_w = c(w)/T$. Bounded on $[0, 1]$; $H_{\text{norm}} = 1$ indicates a uniform word distribution (maximum lexical diversity for the given V), $H_{\text{norm}} \to 0$ indicates that a small set of words dominates the corpus.	Authors' calculation
LLM-mechanism regressors (computed at the disclosure level)		
Discriminative term	A word identified as informative for distinguishing one risk-factor category from the others. We rank candidate words within each (model, category, text source) cell by lift, retain the top 1,000 with *in category count* ≥ 15 and significant χ^2 statistics, and treat the resulting vocabulary as the discriminative term set for that category. The same procedure is applied to the human (Item 1A) source text and to each LLM's summary, producing $V_{i,\text{source}}$ and $V_{im,\text{summary}}$ respectively.	Authors' calculation
Omission rate	$O_{im} = 1 - \frac{\lvert V_{i,\text{source}} \cap V_{im,\text{summary}} \rvert}{V_{i,\text{source}}}$. Share of human discriminative terms not present in the LLM's discriminative-term set.	Authors' calculation
Injection rate	$I_{im} = 1 - \frac{\lvert V_{im,\text{summary}} \cap V_{i,\text{source}} \rvert}{V_{im,\text{summary}}}$. Share of LLM discriminative terms not present in the human discriminative-term set.	Authors' calculation
Months since training cutoff	Calendar months between the 10-K filing date and the LLM provider's published knowledge cutoff (GPT-4o-mini: October, 2023; Grok-3-mini: November, 2024; Llama-3.3: December, 2023; Phi-4: June, 2024).	Provider documentation
Outcome variables (firm-year level, year t+1)		
NI/Assets	Annual net income divided by total assets	Compustat annual
Sales/Assets	Annual sales revenue divided by total assets	Compustat annual
R&D/Assets	Annual R&D expense divided by total assets	Compustat annual
Analyst Dispersion	Standard deviation of one-year-ahead analyst EPS forecasts issued in the 90 days following the 10-K filing date.	IBES Summary
Firm-level controls (firm-year, year t)		
Leverage	Long-term debt divided by total assets	Compustat annual
Cash/Assets	Cash and short-term investments divided by total assets	Compustat annual
Market-to-Book	Market value of common equity divided by book value: $(csho \times prcc_f) \big/ ceq$	Compustat annual
Operating Income/Assets	Operating income before depreciation divided by total assets: $oibdp/at$	Compustat annual