Out of the Black Box: Uncertainty Quantification for LLMs via Conditional Probabilities *

Hui Chen[†] Antoine Didisheim[‡] Luciano Somoza[§]

March 9, 2026

Abstract

Autoregressive LLMs generate text by sampling from estimated probability distributions over the next token, conditional on prior context. We use these probabilities to construct an entropy-based measure of prediction uncertainty, termed *inner confidence*. In news classification, LLM predictions with higher inner confidence are systematically more accurate. To evaluate the measure's economic relevance, we form long-short portfolios based on LLM predictions. The portfolio based on high-confidence predictions achieves a Sharpe ratio about 20% higher than the unconditional benchmark, while the one based on low-confidence predictions yields no excess returns. In contrast, self-declared confidence exhibits significant decoding biases and provides no comparable performance gains.

Keywords: Large language models (LLMs); inner probabilities; entropy; inner confidence; decoding bias; news sentiment classification; look-ahead bias

*We thank Rohit Allena (discussant), Leland Bybee (discussant), Martina Frachini, Gerard Hoberg, Alejandro Lopez-Lira, Dongyihai Peng, Gordon Phillips, Mohammad Pourmohammadi, Simon Scheidegger, Hanqing Tian, Nitin Yadav, Terry Zhang, and participants at the AFA 2026, WFA 2025, NTU AI for Finance Summer School, and FIRN-UQ Asset Management Meeting for their valuable comments.

[†]MIT Sloan and NBER. Email: huichen@mit.edu

[‡]University of Melbourne. Email: antoine.didisheim@unimelb.edu.au

[§]ESSEC Business School. Email: somoza@essec.edu

> All [LLMs] are doing, after all, is predicting the next word needed to string out a response that will statistically satisfy a prompt.
>
> Geoffrey Hinton

Introduction

Recent breakthroughs in artificial intelligence, particularly Large Language Models (LLMs), have drastically accelerated their adoption in financial markets and academic research.[1] Because these models are highly effective at simulating human language, their most natural use involves prompting and analyzing generated text. While intuitive, we argue in this paper that decision-making based solely on textual outputs is often suboptimal, especially in high-stake situations. Generated text is the outcome of two distinct components of the LLM: a *neural language model* that produces a probability distribution over the next token given a sequence of prior tokens,[2] and a *decoding strategy* that selects the next token (stochastically or deterministically) from this conditional distribution. Focusing exclusively on the generated text therefore ignores the uncertainty associated with the output, discards valuable information contained in the model's internal beliefs, and introduces additional opacity through the decoding process.

Our central premise is that the conditional probabilities produced by LLMs, which we refer to as inner probabilities, constitute economically meaningful information that can be directly analyzed and aid decision-making. These probabilities summarize the model's beliefs prior to decoding and are standard outputs of most private and open-source LLMs. By working directly with these inner probabilities, researchers can abstract from the idiosyncrasies of decoding strategies and obtain a "white-box" measure of predictive uncertainty.

Specifically, we propose an entropy-based framework for measuring the uncertainty of LLM outputs using inner probabilities. The framework is designed to be interpretable, semantics-aware, and scalable for large empirical applications. While this framework generalizes to multi-token uncertainty measurement, we argue that in many settings it is preferable to restrict LLM responses to a small, predefined set of discrete answers (e.g., "Yes" or "No"; "positive", "neutral" or "negative"). Such constraints help eliminate spurious inflation in uncertainty arising from synonyms (where probability mass is dispersed across lexically distinct but semantically identical tokens), mitigate the uncertainty feedback loop inherent in

[1] See for example, Lopez-Lira and Tang (2023); Bybee (2023); Cheng, Lin, and Zhao (2025); Babina, Fedyk, He, and Hodson (2024); Li, Shi, Xia, and Yang (2025); Acikalin, Caskurlu, Hoberg, and Phillips (2026); He, Lv, Manela, and Wu (2025), among others.

[2] A token can represent a single character such as "A" or "3", a fragment of a word like *ing* or *bel*, or even an entire word. Tokens serve as the basic units that AI models use to encode and process text.

multi-token generation, and yield a transparent and tractable uncertainty measure. In the binary setting, the entropy measure is a monotonic transform of a simple inner confidence statistic.

We examine whether our proposed uncertainty measure reflects the objective uncertainty of LLM predictions or merely the subjective beliefs of the model, which could potentially be completely detached from reality. We do so through a financial news classification experiment. For a sample of 100,000 firm-specific news items released during market hours, we prompt an LLM (GPT-4o) to classify each news item as positive or negative for the associated firm's stock price and compute the inner confidence for each prediction. Using same-day open-to-close returns as ground truth, we document a strong and monotonic relationship between inner confidence and classification accuracy. Predictions in the top quintile of inner confidence achieve an accuracy of approximately 64.7%, while those in the bottom quintile have an accuracy of about 51%. The difference in accuracy between the top and bottom groups is highly statistically significant.

To assess the economic significance of the uncertainty measure, we embed inner confidence into portfolio construction. Following Lopez-Lira and Tang (2023), we prompt an LLM to assess the impact of overnight news on the future short-term performance of the stock (positive or negative) and compute the corresponding inner confidence for each prediction. Using a rolling-window procedure, we estimate an inner-confidence threshold that categorizes all LLM predictions into high- and low-confidence groups. We then construct long-short portfolios within each group, compute next-day open-to-close returns, and compare the performances of these portfolios against that of a baseline long-short portfolio formed using the full sample.

In line with Lopez-Lira and Tang (2023), the baseline strategy constructed from the full sample produces an annualized mean excess return of 40% and a Sharpe ratio of 4.24. For the high-inner-confidence portfolio, the mean excess return rises to 50% while the Sharpe ratio rises to 5.15. In contrast, the low-inner-confidence portfolio yields a mean excess return of -10%, with a Sharpe ratio of -0.39, both of which are statistically indistinguishable from zero. These results indicate that inner confidence effectively identifies instances in which LLM forecasts lack economic reliability and should not be used as the basis for investment decisions.

We demonstrate the robustness of these findings to alternative ways for setting the high- vs. low-confidence thresholds. Further analysis shows that the Sharpe ratios for high-inner-confidence long-short portfolios are particularly high in stocks with low market capitalization, low liquidity, and low media or analyst coverage, and that the excess returns primarily

come from the short legs. These cross-sectional patterns are consistent with the interpretation that the performance of the long-short strategy is driven by LLM's ability to identify market underreaction to news, especially in smaller, less liquid, and less media- or analyst-covered stocks. Overall, these findings demonstrate that incorporating inner probabilities, particularly for uncertainty quantification, could substantially enhance the performance of LLM-prediction-based strategies.

While the two experiments above demonstrate that the entropy-based uncertainty measures are informative, they do not imply that LLM beliefs are unbiased or that they accurately reflect the true level of uncertainty in economic outcomes. In fact, recent studies show that modern LLMs exhibit biases similar to those observed in human expectations (Bybee, 2023; Fedyk, Kakhbod, Li, and Malmendier, 2024), are often miscalibrated, and tend to produce overly concentrated probabilities (e.g., Desai and Durrett, 2020; Jiang, Kim, Guan, and Gupta, 2021). Nevertheless, even small differences in inner probabilities and the associated uncertainty measures still carry meaningful information. This is why we focus on the relative ranking of inner confidence rather than its absolute level in our applications.

Moving beyond the empirical connection between inner confidence and predictive accuracy, we examine factors that drive LLMs' inner confidence in order to better understand its belief formation process. First, we examine which firm- and market-level variables are systematically associated with the LLM's confidence in its news classifications. We find that inner confidence is lower (i.e., uncertainty is higher) for news items followed by weaker subsequent firm-level returns or elevated firm-level return volatility over the next 30 days. Confidence is also lower for news accompanied by unusually high abnormal trading volume, consistent with greater investor disagreement. At the market level, news released on days with unusually high aggregate news volume are associated with lower inner confidence.

Second, we examine the relationship between news subjects and inner confidence. We use the published word weights in Bybee, Kelly, Manela, and Xiu (2024) to assign each news item in our sample to one of 180 pre-defined topic categories. We then estimate a Lasso regression in which the topic indicators are used to predict the decile of inner confidence for each article. The tuning parameter is chosen so that the model selects the ten most informative categories. The full-sample results show that the LLM exhibits higher inner confidence when processing news related to lawsuits, options activity, or product development, and lower confidence for items concerning news conferences or research. Applying this procedure to individual firms reveals firm-level heterogeneity in information complexity across news subjects.

Third, we use prompting to manipulate an LLM's prior and perceived signal precision, and examine the resulting effects on inner confidence. In the news classification setting,

the LLM can be viewed as holding a prior belief about the firm, which is formed during pre-training. Based on the news item, which serves as a signal, it then produces a posterior. Benchmarking this process against the Bayesian framework offers a useful way to evaluate the model's economic rationality.

Specifically, we redo our first news classification experiment under three sets of modified prompts. First, we state that analysts agree (or disagree) on the firm's general outlook. This modification is designed to shift the model's prior uncertainty about future returns. Second, we state that analysts agree (or disagree) on the interpretation of the specific news item, thereby altering the model's perceived signal precision. Finally, as a placebo, we introduce a third prompt that appends words commonly associated with high or low human confidence, without explicitly modifying either the prior or the signal.[3] Consistent (at least directionally) with a Bayesian learner, the first two sets of prompt conditioning shift the model's internal confidence in the expected direction. When analysts agree on the firm's prospect or when a news signal is perceived to be precise, the model's confidence in its classification rises. In contrast, confidence-related words added at random have no effect.

Having established the economic relevance and determinants of inner confidence, we next examine whether it can inform an important debate in the LLM-in-finance literature: the presence of look-ahead bias, whereby model outputs inadvertently reflect future information embedded in the pre-training data.[4] The idea is that if an LLM makes a prediction by recalling memorized outcomes, the high predictive accuracy is likely to be accompanied by artificially high inner confidence. To test this idea, we follow Didisheim, Fraschini, and Somoza (2025); Lopez-Lira, Tang, and Zhu (2025) and prompt the LLM to recall realized returns for individual assets without any additional context. This design isolates potential memorization from the confusion effect of added context. In this setting, we uncover a strong positive association between the model's inner confidence and its ability to correctly recall realized returns.

The evidence suggests that unusually high inner confidence can serve as a diagnostic signal of implicit memory retrieval, thereby providing a practical tool for detecting look-ahead bias in LLM-based financial applications. We apply this idea to our news classification setting by comparing the time series of daily average inner confidence across all financial news in our sample during the one-year periods before and after the GPT-4o's training cutoff date. If look-ahead bias were driving predictive performance, one would expect systematically higher

[3]Positive noise conditioning: "Confidence, clarity, conviction, precision, certainty." Negative noise conditioning: "Doubt, anxiety, mistrust, uncertainty, hesitation."

[4]See, for example, Levy (2024); Sarkar and Vafa (2024); Ludwig, Mullainathan, and Rambachan (2025); Engelberg, Manela, Mullins, and Vulicevic (2025).

inner confidence in the pre-cutoff period, when memorized outcomes could potentially be recalled. However, we find no statistically significant difference in inner confidence between the two periods. If anything, the inner confidence is slightly higher in the post-cutoff sample. This pattern is consistent with the findings of Glasserman and Lin (2023), who report that the look-ahead bias of LLMs in financial news analysis is smaller than the effects induced by model confusion.

Finally, we compare the white-box uncertainty measure in this paper with black-box approaches that infer model uncertainty without relying on inner probabilities. One such method measures uncertainty by examining the dispersion of model outputs across repeated queries using the same prompt. This method is not only computationally costly, but also noisy and sensitive to hyperparameter choices (temperature). Another intuitively appealing black-box measure is *declared confidence*, obtained by directly prompting the LLM to report its confidence in its primary output (see, e.g. Bybee, 2023; Fedyk et al., 2024).

We compare the information content of declared confidence and inner confidence in our portfolio experiment. The rank correlation between the two measures is 0.74 (the Pearson correlation is 0.38). Unlike inner confidence, conditioning on declared confidence does not enhance the strategy performance. The mean excess return for the high-declared-confidence long-short portfolio is no higher than that for the baseline portfolio, and its Sharpe ratio is significantly lower.

We further show that declared confidence is systematically biased by the decoding process. Because it is a token generated following the prediction token, its conditional distribution depends on the realized decoding outcome for the primary output. We again use the news classification setting to demonstrate this bias. The prompt asks the LLM to classify a news item as (A) positive or (B) negative and then report a confidence score between 0 and 1. We rerun this prompt multiple times for the same news item to introduce randomness in the classification. While the uncertainty of the LLM belief should depend only on the news item and not the realized prediction token selected, we find that, even for the same news item, declared confidence is significantly higher, by 0.0337 on average, when the prediction token is "A" instead of "B."

Related literature This paper contributes to the rapidly growing literature that studies the use of large language models (LLMs) in economics and finance (see e.g., Korinek, 2023; Eisfeldt and Schubert, 2024; Hoberg and Manela, 2025, for recent surveys). Existing work has primarily followed two approaches. One approach leverages LLMs as representation learners, extracting embeddings that serve as inputs into downstream analysis. See

for example Chen, Kelly, and Xiu (2022). The second and more common approach treats LLMs as answer machines, using them to summarize text, generate predictions, or simulate beliefs and behaviors. For example, Chang, Dong, Martin, and Zhou (2023) use LLM to measure earnings call sentiment, Saunders, Steffen, and Verhoff (2025) for identifying loan covenant violations, and Caragea, Chen, Cojoianu, Dobri, Glandt, and Mihaila (2020) for patent classifications. Bybee (2023) introduces a survey of economic expectations formed by querying LLMs. In the same spirit, Fedyk et al. (2024) investigates whether AI models accurately capture investment preferences across different demographics. Ashokkumar, Hewitt, Ghezae, and Willer (2024) tests the predictive capabilities of LLMs concerning the outcomes of social science experiments.

Our framework advances this second approach by focusing on a largely overlooked component of LLM output: the model's internal probability distribution. Specifically, we show that these probabilities can be used to construct an interpretable and scalable measure of predictive uncertainty for LLM outputs, which can be incorporated into economic analysis. In that sense, this paper is also connected to research that quantifies uncertainties in machine learning model predictions. Allena (2025) derives ex-ante confidence intervals for risk premium forecasts from penalized linear models and neural networks, which are shown to improve investment strategies.

Our uncertainty measures are based on entropy, which is commonly used for quantifying uncertainty in classification problems. Entropy has also been used by Glasserman and Mamaysky (2019) to measure news unusualness. Glasserman, Mamaysky, and Qin (2023) extend this idea and use a self-trained recurrent neural network to construct measures for changes in news novelty, which can predict returns for the market index and macroeconomic outcomes. Their notion of entropy (rate) is about the average uncertainty per token in the long run, while we focus on the uncertainty for a specific prediction associated with a specific prompt.

Finally, our paper contributes to the growing discussion on the interpretability of LLMs. Korinek (2023) emphasizes the need for approaches that shed light on the internal mechanisms of these models in order to strengthen their credibility and adoption in research and practice. The black-box nature of LLMs complicates efforts to understand and evaluate their outputs, particularly in high-stakes applications. Our framework makes progress in opening up the black box by removing the randomness introduced by the decoding process and directly analyzing the model's inner probabilities.

1 LLM and Inner Probabilities

In this Section, we briefly review large language models (LLMs) and introduce the notion of *inner probabilities*. Colloquially, the term "LLM" refers to the entire process that maps a textual prompt into a generated response. This process comprises two distinct components. The first component is a *neural language model* that can be viewed as a probabilistic system. It produces conditional probability distributions over the next token given a preceding context. The second is a *decoding strategy*, which specifies how these conditional distributions are queried and transformed into a realized sequence of tokens through deterministic or stochastic selection rules.

Existing applications of LLMs in economics and finance typically focus on the realized textual output. By contrast, the central premise of this paper is that the model's internal conditional probabilities contain economically meaningful information not captured by the generated text. By analyzing these probabilities directly, we can effectively reduce the opacity introduced by the decoding strategy. In particular, the inner probabilities provide a principled way for quantifying the uncertainty associated with LLM-generated predictions.

We now briefly review the two core components of a large language model: the transformer neural network and the decoding strategy.

The network: The neural network architectures underlying LLMs are exceptionally large, reflecting the scale implied by the first "L" in the acronym. In practice, most state-of-the-art LLMs adopt a decoder-only transformer architecture designed for autoregressive generation. Within this framework, transformer layers (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, and Polosukhin, 2017) compute context-dependent representations using masked self-attention, which directs attention to different parts of an input text sequence and enables the model to capture long-range semantic relationships across the input. For example, the model can infer whether the term "bank" refers to a financial institution or a riverbank based on surrounding context. As information propagates through successive layers, the network constructs increasingly abstract internal representations of the evolving context, enabling it to generate coherent and contextually appropriate text to follow the prompt, thus achieving the goal of completing sentences, answering questions, performing reasoning tasks, etc.

Although LLMs may differ in architectural details, they are fundamentally characterized by their training objective. Modern LLMs are trained using causal language modeling, in which the model learns to predict the next token in a sequence conditional on all preceding tokens (Radford et al., 2019; Brown et al., 2020). During pre-training, contiguous segments

of text are sampled from a large corpus and tokenized. At each position in the sequence, the model observes only the preceding tokens, with future tokens masked to enforce the causal structure. The network outputs a probability distribution over the full vocabulary for the next token, and model parameters are estimated by minimizing the cross-entropy loss between the predicted distribution and the realized token, which is equivalent to maximum likelihood estimation.

Let P denote a fixed prompt (context), and let $\mathcal{V}$ be the LLM's vocabulary of all possible tokens. An autoregressive LLM defines a conditional probability distribution over the next token, s, from a finite vocabulary $\mathcal{V}$. This conditional distribution is given by

$$p(s \mid P), \qquad s \in \mathcal{V}. \tag{1}$$

Here, p denotes the probability distribution produced by the neural network, such that $\sum_{s\in\mathcal{V}} p(s \mid P) = 1$. It is an estimate of the true but unknown distribution $q(s \mid P)$. For this reason, we refer to p as the LLM's *inner probabilities*.

The LLM vocabulary typically consists of between 50,000 (Radford et al., 2019) and 150,000 tokens (Dubey et al., 2024). The high dimensionality of the token space, combined with the normalization constraint (probabilities summing to one), naturally results in a high degree of sparsity in the conditional distribution p. This sparsity plays a central role in enabling tractable analysis of the model's inner probabilities. For any given context P, it often suffices to focus on a small subset of tokens in $\mathcal{V}$ with non-negligible predicted probabilities.

Decoding strategy: After training, the neural network does not autonomously generate text. It requires an algorithm to select tokens from the conditional probability distribution given the current context. For instance, given the prompt "The cat is on the," a well-trained model might assign high probability to "table" and negligible probability to unrelated words like "stock." The decoding strategy applies a specified selection rule to choose the next token, which is then appended to the existing context to form an updated context for predicting subsequent tokens. This iterative process continues until a stopping condition is met, such as reaching a maximum length or generating a special end-of-sequence token.

One possible decoding strategy selects the most probable token at each step:

$$s_1 = \arg\max_{s\in\mathcal{V}} p(s \mid P),$$
$$s_2 = \arg\max_{s\in\mathcal{V}} p(s \mid s_1, P),$$

and so on. Although intuitive, this deterministic decoding strategy has been shown to yield outputs that are repetitive and stylistically monotonous (Holtzman, Buys, Du, Forbes, and Choi, 2019). Consequently, modern LLMs typically employ stochastic sampling, for example, by selecting the next tokens according to their predicted probabilities. The degree of randomness is controlled by a temperature parameter that adjusts the distribution's sharpness: a lower temperature concentrates the distribution around the most probable tokens (with a temperature of zero for the deterministic case), while a higher temperature flattens the distribution, allowing for more diverse token selection.

It is worth emphasizing that the simplified explanation above masks the richness of real-world decoding strategies,[5] which add considerable complexity and opacity atop the probabilistic model. For example, rather than selecting individual tokens sequentially, *beam search* explores multiple candidate sequences simultaneously to find the one that maximizes the joint probability of the entire sequence, while dynamic filtering rules, such as top-k or nucleus sampling, truncate the probability distribution before sampling occurs. While these decoding strategies are commonly used in commercial LLMs, the specific algorithms and hyperparameters are rarely disclosed.

LLM uncertainty. Researchers are rightfully concerned with the opacity of LLMs and often refer to them as black boxes. The two distinct components of LLMs discussed above have important implications for understanding and quantifying uncertainty in their outputs. Just as deep neural networks are generally considered black-box models due to their complexity and scale, the neural language model component of LLMs shares this opacity. State-of-the-art models now contain billions of parameters, rendering the interpretation of individual neurons or parameters infeasible. Decoding strategies further exacerbate the opacity and introduce additional uncertainty in two ways, first due to the lack of transparency in the decoding algorithms themselves, and second due to a feedback loop, whereby each selected token becomes part of the context for subsequent predictions, altering future conditional distributions. As a result, randomness introduced by the decoding strategy can cascade and amplify the unpredictability of the final output.

Nevertheless, even in the absence of structural interpretability, the uncertainty associated with the neural language model's predictions is well-defined given our knowledge of the conditional probability distribution over the next token. In the next section, we show how these inner probabilities provide the foundation for an entropy-based framework for uncertainty quantification. This framework abstracts from the opaque decoding strategy and

[5] These include greedy decoding and beam search (Sutskever, Vinyals, and Le, 2014), as well as stochastic techniques such as top-k (Fan, Lewis, and Dauphin, 2018) and nucleus sampling (Holtzman et al., 2019).

yields principled, interpretable measures of uncertainty for LLM-generated outputs.

2 Methodology

In this Section, we develop an entropy-based framework for measuring the uncertainty of large language model (LLM) output, which is designed to be interpretable, semantics-aware, and feasible in large-scale empirical applications. We also use the framework to analyze the feedback loop embedded in a multi-token generation setting and discuss its implications for uncertainty quantification and prompt engineering. Finally, we compare the proposed entropy-based measures with alternative black-box approaches for uncertainty quantification.

2.1 Token-level Uncertainty

As discussed in the previous section, an autoregressive LLM produces a conditional probability distribution over the next token, s, based on the prompt P. This distribution is denoted by $p(s \mid P)$ in Eq. (1). Predicting the next token is effectively a classification problem over a large but finite set of discrete outcomes (tokens) from a finite vocabulary $\mathcal{V}$. A canonical measure of the predictive uncertainty in this setting is the Shannon entropy of the conditional distribution $p(s \mid P)$:

$$H_{\text{tok}}(P) = -\sum_{s \in \mathcal{V}} p(s \mid P) \log p(s \mid P). \tag{2}$$

Large values of $H_{\text{tok}}(P)$ indicate a diffuse predictive distribution (high uncertainty); small values indicate a concentrated predictive distribution (high confidence).

In practice, computing $H_{\text{tok}}(P)$ can be challenging. Modern LLM vocabularies are very large, often exceeding 10^5 tokens. For most prompts, however, the conditional probability distribution $p(s \mid P)$ is highly sparse: the vast majority of probability mass is concentrated on a relatively small set of tokens. Furthermore, closed-source LLMs typically only provide a limited number of (highest) token probabilities from the model API. Thus, it is more practical to compute a truncated entropy measure in place of $H_{\text{tok}}(P)$.

Specifically, let $\mathcal{V}_K(P) \subset \mathcal{V}$ denote the set of $K > 0$ tokens with the highest conditional probabilities under $p(\cdot \mid P)$. We define the *top-K normalized* conditional probability

distribution over $\mathcal{V}_K(P)$ as

$$p_K(s \mid P) = \begin{cases} \dfrac{p(s \mid P)}{\sum_{t \in \mathcal{V}_K(P)} p(t \mid P)}, & s \in \mathcal{V}_K(P), \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The *top-K token-level entropy* is then given by

$$H^K_{\text{tok}}(P) = -\sum_{s \in \mathcal{V}_K(P)} p_K(s \mid P) \log p_K(s \mid P). \tag{4}$$

When $K \ll |\mathcal{V}|$ while $\sum_{t \in \mathcal{V}_K(P)} p(t \mid P)$ is close to one, $H^K_{\text{tok}}(P)$ will not only closely approximate the full token-level entropy $H_{\text{tok}}(P)$, but also be much easier to compute.

A particularly tractable setting in which Eq. (2) can be implemented directly arises when the set of admissible outputs is explicitly constrained to be small. In certain applications, we may want to instruct the LLM to generate a response from a finite set of discrete labels, for example, choosing among options "A", "B", or "C," or providing a binary "Yes" or "No" answer. Consider the binary case, where the admissible token set is $\mathcal{S} = \{A, B\}$, the token-level entropy $H_{\text{tok}}(P)$ simplifies to

$$H_{\text{bin}}(P) = -p(A \mid P) \log p(A \mid P) - (1 - p(A \mid P)) \log(1 - p(A \mid P)), \tag{5}$$

which attains its maximum when $p(A \mid P) = p(B \mid P) = 0.5$.

Motivated by this observation, we define a simple inner confidence statistic for a given LLM output. With the probability of output s conditional on prompt P being $p(s \mid P)$, the probability that the output is not s is given by $1 - p(s \mid P)$. Given that the binary entropy is maximized when $p(s \mid P) = 0.5$, we measure the model's inner confidence on the output s based on the distance of the inner probability from maximum uncertainty:

$$C(s \mid P) = \left| p(s \mid P) - \tfrac{1}{2} \right|. \tag{6}$$

Notice that this measure of inner confidence assumes that all outputs that are not s can be aggregated into a single complementary outcome. This assumption is valid when the prediction task is indeed binary, or when the non-s outcomes are equivalent from a decision perspective. For example, a trading strategy may only invest in a stock if the LLM predicts a firm-specific news event is "positive," while all other outcomes ("negative," "neutral," or "uncertain") lead to no investment. As the following lemma shows, this inner confidence

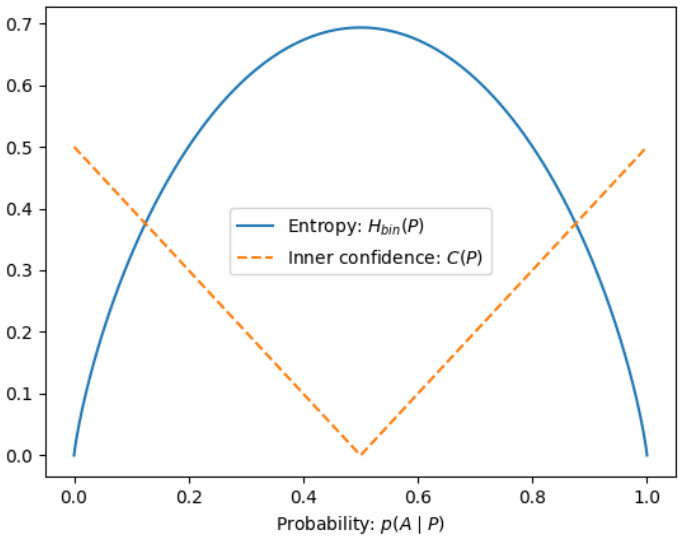

Figure 1: **Entropy and inner confidence in the binary case.** The solid blue curve plots the binary entropy $H_{\text{bin}}(P)$ as a function of $p(A \mid P) \in [0, 1]$. The dashed red curve plots the inner-confidence statistic $C(A \mid P)$.

statistic is monotonically related to the binary entropy.

Lemma 1 (Monotone transform between inner-confidence and binary entropy)**.** *There exists a continuous, strictly decreasing function g such that*

$$C(s \mid P) = g\big(H_{\text{bin}}(s \mid P)\big), \tag{7}$$

where $H_{\text{bin}}(s \mid P)$ is defined for the conditional distribution over $\mathcal{S} = \{s, \neg s\}$. In particular, ranking predictions by $C(s \mid P)$ is equivalent (up to reversal) to ranking them by binary entropy.

Proof. See Appendix A. □

Lemma 1 shows that the inner-confidence statistic $C(s \mid P)$ is an order-preserving (more specifically, order-reversing) transformation of the entropy-based uncertainty measure. This result is also illustrated in Figure 1. Thus, in empirical applications where uncertainty measures are used ordinally, e.g., when ranking the uncertainties of different LLM outputs, sorting by the inner-confidence statistic $C(s \mid P)$ is equivalent to sorting by $H_{\text{bin}}(s \mid P)$.

2.2 Synonyms and Semantic Labels

The token-level entropy defined in Eq. (2) treats all tokens as distinct outcomes, even when multiple tokens are semantically equivalent (synonyms). When probability mass is distributed across such synonymous tokens, the entropy measure can become inflated by capturing lexical variation rather than focusing on economically meaningful uncertainty. For example, following the prompt "The stock market is expected to," an LLM might assign similar probabilities to tokens like "rise" and "increase." Treating these tokens as distinct outcomes would inflate the uncertainty of output.

To better align uncertainty quantification with economic decisions, we would ideally like to group tokens into equivalence classes based on their semantic interpretations in a given context, and then compute the entropy with respect to the conditional distribution over these semantic classes. Formally, denote a finite label space $\mathcal{Y}$ and a deterministic mapping $f : \mathcal{V} \to \mathcal{Y}$ that identifies tokens that are semantically equivalent (e.g., "rise" and "increase" will be assigned to the same label class). The induced conditional distribution in the label space is given by

$$p_Y(y \mid P) = \sum_{s \in f^{-1}(y)} p(s \mid P), \qquad y \in \mathcal{Y}. \tag{8}$$

We can then compute the corresponding semantic-label entropy:

$$H_{\text{sem}}(P) = -\sum_{y \in \mathcal{Y}} p_Y(y \mid P) \log p_Y(y \mid P). \tag{9}$$

It is easy to show that $H_{\text{sem}}(P) \leq H_{\text{tok}}(P)$. Intuitively, H_{sem} leaves out the dispersion across synonyms within each label class, while H_{tok} counts all variation. In practice, we can apply the mapping f to the top-K tokens with the highest conditional probabilities, and compute a top-K semantic-label entropy in an analogous manner.

Implementing the semantic-label entropy requires specifying the mapping f. There are several possibilities. First, we can use a curated synonym dictionary to create the mapping. Second, the mapping can be learned via embedding-based clustering. Specifically, we first map tokens into a high-dimensional semantic embedding space using pre-trained language model embeddings, and then group them into semantic classes using unsupervised clustering algorithms (e.g., using k-means clustering). This approach has the advantage of being flexible and adaptable to different contexts. For example, embeddings can be trained or adapted for domain-specific corpora, such as financial texts, while clustering can be performed locally at the prompt level to capture context-dependent semantic equivalences. Finally, similar to the binary entropy discussed in Section 2.1, we can avoid the synonym issue altogether

by constraining the LLM to generate outputs from a small controlled token set through prompting. Our empirical study in Section 3 will be built on this last approach.

2.3 Multi-token Uncertainty

The token-level entropy measures discussed above can be extended to multi-token outputs, e.g., a sentence. Let a multi-token sequence be denoted by $S = (s_1, \ldots, s_n)$. We can compute its conditional probability using the chain rule for the autoregressive model:

$$p(S \mid P) = \prod_{i=1}^{n} p(s_i \mid s_{<i}, P). \tag{10}$$

where $s_{<i} \equiv (s_1, \ldots, s_{i-1})$ denotes the sequence of previously-generated tokens up to but not including token i. Importantly, these previously-generated tokens become part of the augmented prompt for predicting token s_i.

Conceptually, we can measure the uncertainty of the entire sequence using the mean token entropy:

$$H_{\text{seq}}(P) = -\frac{1}{n} \sum_{S} p(S \mid P) \log p(S \mid P), \tag{11}$$

which is the sequence-level entropy normalized by sequence length n. The summation in Eq. (11) is over all possible sequences of length n, and the normalization ensures that the measure is comparable across sequences of different lengths. Without normalization, longer sequences would naturally have higher entropy.

However, $H_{\text{seq}}(P)$ is generally computationally intractable: the number of possible sequences, $|\mathcal{V}|^n$, grows exponentially with sequence length n. Even if we restrict the possible sequences based on top-K tokens at each step, we will only be able to compute the measure for short sequences.

For practical applications, we propose a computationally feasible implementation of Eq. (11) based on sampling. Specifically, we first draw M independent sample sequences $S^{(1)}, \ldots, S^{(M)}$ based on the same prompt P and use stochastic decoding (with temperature ≥ 1.0) to ensure diversity. For each generated sequence, we then compute its conditional probability according to Eq. (10) and renormalize it:

$$p_M(S^{(j)} \mid P) = \frac{p(S^{(j)} \mid P)}{\sum_{\ell=1}^{M} p(S^{(\ell)} \mid P)}. \tag{12}$$

Finally, we can compute the mean token entropy as:

$$H^M_{\text{seq}}(P) = -\frac{1}{n}\sum_{j=1}^{M} p_M(S^{(j)} \mid P) \log p_M(S^{(j)} \mid P). \tag{13}$$

As in the case of single-token outputs, lexically different sentences could have similar semantic meanings. To deal with this issue, we can extend the semantic-label entropy from Eq. (9) to multi-token outputs. Similar to the procedure described in Section 2.2, we can first assign sequences into semantic clusters, then aggregate conditional probabilities for the M sequences at the cluster level, and finally compute the entropy over the cluster distribution. For sequence clustering, Kuhn, Gal, and Farquhar (2023) have proposed the bidirectional entailment clustering algorithm. Alternatively, we can also use embedding-based clustering as described in Section 2.2. In Appendix B we provide an example to illustrate how our method can be expanded to the multi-token setting.

The feedback loop. As mentioned earlier, the autoregressive decoding process creates a feedback loop: each selected token becomes part of the context for predicting subsequent tokens; therefore, the randomness with each token selection will influence the conditional distributions of subsequent tokens, especially when the selected token has significant and persistent influence on the distribution of subsequent tokens. This feedback mechanism has important implications for uncertainty quantification and prompt engineering of a multi-token sequence.

To formalize this feedback loop, consider the cases of one-token versus two-token generation. Let S_1 denote the first token generated by the model given a prompt P, and let S_2 be the subsequent token. While the conditional distribution of S_1 only depends on the prompt P, the marginal distribution of S_2 integrates over the randomness in S_1,

$$p(S_2 \mid P) = \sum_{s_1 \in \mathcal{V}} p(s_1 \mid P)\, p(S_2 \mid s_1, P). \tag{14}$$

The entropy for this marginal distribution of S_2 admits the decomposition

$$H(S_2 \mid P) = H(S_2 \mid S_1, P) + I(S_2; S_1 \mid P) \geq H(S_2 \mid S_1, P). \tag{15}$$

The term $H(S_2 \mid S_1, P)$ captures the average conditional uncertainty of the second token given a realized first token. The mutual information term, $I(S_2; S_1 \mid P)$, measures the conditional dependence between S_2 and S_1 given the prompt P. It quantifies the additional

uncertainty about S_2 induced by the randomness of the initial decoding step. Intuitively, if a particular realization of S_1 has significant influence on the conditional distribution of S_2, then $I(S_2; S_1 \mid P)$ will be large, resulting in larger marginal entropy for S_2. Finally, the inequality in Eq. (15) follows because $I(S_2; S_1 \mid P)$ is nonnegative; $I(S_2; S_1 \mid P) = 0$ if and only if S_2 is conditionally independent of S_1 given P.

Extending Eq. (15) to a sequence with tokens $S_1, \ldots, S_L$, the marginal entropy of the final token admits the decomposition

$$H(S_L \mid P) = H(S_L \mid S_{<L}, P) + \sum_{i=1}^{L-1} I(S_L; S_i \mid S_{<i}, P), \tag{16}$$

where $S_{<i} \equiv (S_1, \ldots, S_{i-1})$. The term $H(S_L \mid S_{<L}, P)$ represents the average conditional uncertainty of the L-th token given a realized prefix, $S_{<L}$. Each mutual information term $I(S_L; S_i \mid S_{<i}, P)$ is nonnegative and captures the incremental uncertainty about the L-th token arising from the randomness of an earlier token S_i.

This result clearly shows that the realization of each decoding step alters the conditional distributions of the subsequent tokens, with the resulting dependence propagating forward through the sequence. Consequently, the marginal entropy of later tokens is weakly increasing in sequence length. Longer generated sequences therefore embed a larger uncertainty feedback loop, as randomness introduced at early stages persists and amplifies the unpredictability of subsequent outputs.

As a concrete example, consider the prompt:

```
Based on the recent 25 bps cut in the federal funds rate, provide a
concise forecast for the S&P 500 performance in the next month. End your
sentence with either the token ‘positive’ or ‘negative’.
```

If the model generates the final prediction directly, and the training data show that the market generally responds positively to rate cuts, then the model may assign a high probability to the token "positive," and the uncertainty will be low. However, in a multi-token generation, the first token S_1 drawn could be a conjunction like "while," which could then lead to a "narrative drift" that significantly alters the conditional distribution for the final token. An example could be:

```
While rate cuts often lead to short-term gains, market participants had
anticipated a more substantial reduction, suggesting that the overall
outlook for the index may be negative.
```

Implications for Prompt Engineering. The feedback loop provides guidance for how to design LLM prompts to minimize uncertainty amplification. Unconstrained or verbose responses mechanically increase uncertainty through the cumulative dependence on earlier decoding choices. To mitigate this effect, we recommend prompts be structured so that the payoff-relevant output is produced early and compactly, ideally as a single token or from a small, predefined set of labels. When multi-token responses are needed, ask for short and direct answers, with reasoning or explanatory text separated from the final output.

At first glance, this recommendation may appear to conflict with the widespread use of chain-of-thought (CoT) prompting to improve LLM performance on complex tasks (Wei and al., 2022). The apparent tension reflects a distinction between unconstrained token generation and structured intermediate reasoning aimed at extracting task-relevant information. Under CoT prompting, intermediate reasoning tokens are part of the conditioning context for the final output. If accurate and informative, they can effectively reduce the conditional entropy $H(S_L \mid S_{<L}, P)$ and improve the accuracy of the output, even though the additional generated tokens mechanically increase the marginal entropy $H(S_L \mid P)$ through the autoregressive feedback loop (see Eq. (16)). This is why we recommend separating reasoning from final answers rather than avoiding CoT altogether. In practice, uncertainty amplification can be further limited by eliciting structured, task-specific intermediate steps instead of open-ended CoT, and by verifying intermediate outputs before producing the final prediction.

2.4 White-box vs. Black-box Uncertainty Quantification

We conclude this section by comparing the entropy-based measures developed in this section with alternative approaches to uncertainty quantification. The entropy-based measures rely on direct access to the LLM's conditional probabilities. We refer to them as white-box measures, in contrast to black-box approaches that infer uncertainty without observing the model's internal probabilities.

One type of black-box methods treats an LLM as a stochastic mapping from prompts to outputs and infer uncertainty by measuring the dispersion across model outputs generated from repeated queries. Besides the obvious computational challenges, there are several issues

associated with these sampling-based methods. First, these measures are built on a limited number of sampled realizations from a high-dimensional output space and therefore conflate intrinsic model uncertainty with randomness induced by the decoding strategy. As a result, these black-box measures are inherently noisy, sensitive to hyperparameter choices, and not readily comparable across models. Second, due to its lack of a clear probabilistic meaning, dispersion-based measures are difficult to interpret theoretically and compare across different prompts. Finally, sampling-based methods scale poorly when uncertainty is driven by rare but economically important outcomes, as low-probability events are systematically underrepresented in finite samples.

Another black-box approach for uncertainty quantification uses model-declared confidence (Lin, Hilton, and Evans, 2022). Under this approach, the LLM is prompted to generate a confidence score following its main output. For example, the model may be asked to rate its confidence on a scale from 1 to 10, which is then used as a proxy for uncertainty. While simple and intuitive, it is unclear how well declared confidence aligns with the model's internal uncertainty, especially in light of the feedback loop discussed in Section 2.3. We compare the empirical relationship between entropy-based measures and declared confidence and document a distinct bias associated with declared confidence in Section 5.

3 Economic Value of Uncertainty Quantification

In this Section, we examine the economic value of quantifying uncertainty in LLM outputs. We conduct two empirical experiments. First, we examine whether LLM predictions regarding the impact of news on stock prices are more accurate when the predictions are associated with lower uncertainty (i.e., higher internal confidence). Second, we evaluate the economic significance of uncertainty quantification by assessing its effect on the profitability of trading strategies constructed using LLM predictions.

In addition, to study the sources of uncertainty, we investigate which topics in the news are most strongly associated with model confidence. Finally, we apply a Bayesian framework to assess the drivers and stability of the inner confidence measure.

3.1 The Relationship Between Confidence and Accuracy

The inner probability distributions can be viewed as an LLM's subjective beliefs. If these beliefs are detached from actual outcomes, quantifying their uncertainty would have little practical value. Here, we use a news classification experiment to examine whether the

Table 1: **Summary Statistics of the News Classification Sample**
The table below presents the summary statistics for the dataset of news articles used in the exercise presented in Section 3.1, sorted by firm size (market equity) deciles. The breakpoints for the size portfolios are based on the NYSE universe (Fama and French, 1993). For each decile, we report the number of firms, number of news items, the median firm size, the average and median number of news items per firm, and the percentage of news items with positive returns.

Size Deciles	**Small**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Large**
Firms	2,402	991	796	675	528	466	399	362	297	256
News Items	29,093	10,039	8,014	6,855	5,173	5,098	5,478	7,706	7,341	15,203
Med ME ($bn)	0.1	0.6	1.2	2.2	3.5	5.5	8.3	15.1	32.4	135.2
Avg News/Firm	12.1	10.1	10.1	10.2	9.8	10.9	13.7	21.3	24.7	59.4
Med News/Firm	8	6	6	5	6	6	8	12	15	37
% Positive Ret.	46.2	50.5	50.7	51.5	51.4	52.4	52.6	54.0	52.5	53.9

proposed uncertainty measure reflects objective predictive uncertainty. Specifically, we study whether higher inner confidence is associated with greater out-of-sample predictive accuracy.

We feed financial news articles published during regular market hours to an LLM and ask it to classify the news as positive or negative for the associated firm's stock price. We then use same-day realized returns as ground truth to assess classification accuracy. Our data are from Reuters.com and consist of a randomly selected sample of articles meeting the following criteria:

1. Following Chen et al. (2022), we only select news that mentions a single firm.[6] This filter ensures that the firm is directly affected by the news rather than only mentioned alongside others. We provide the details of the data-cleaning procedure in Appendix C.
2. To increase the likelihood that contemporaneous daily returns reflect the news item impact, we select only those news items with a timestamp between 10 a.m. and 2 p.m. ET.
3. The news article must contain between 100 and 5,000 characters.
4. To avoid look-ahead bias, we only consider news items published after GPT-4o's training cutoff in October 2023.

Following the criteria above, we obtain a sample of 100,000 financial news items released between October 2023 and December 2024 from 5,142 unique tickers. We also obtain daily return and market capitalization data from CRSP. Table 1 presents the main summary statistics for the sample by NYSE market capitalization decile (Fama and French, 1993).

[6]Reuters attaches every mention of a firm to a ticker marker, which allows a clean identification.

Notice that a firm can have multiple news items on the same day, in which case each news item is treated as a distinct observation. Approximately 31.8% of firm-day observations in our sample have more than one news item in a day.[7]

We classify news using Prompt 1, which elicits the LLM's assessment of the news' impact on the firm's stock price. As discussed in Section 2, issues related to synonymy/semantics and the effect of feedback loops on entropy can be mitigated by designing prompts that elicit concise responses in which, ideally, a single token encodes complex meaning. Accordingly, Prompt 1 is constructed to produce a single-letter output: "A" for positive news and "B" for negative news. We deliberately adopt a forced-choice framework that restricts responses to a binary set, excluding any neutral category. By compelling a directional classification, we aim to isolate the informational content of the underlying probabilities and magnify the relationship between the model's confidence and its objective predictive accuracy.

```
Based on the news below, please say if you think the return for the stock
 {target} will be A (positive) or B (negative).

Please write only a letter A or B. Add no other formatting or bolding.

{headline}
{body}
```

Prompt 1: **Classifying the News**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

As a technical consideration, note that despite the prompt instruction, an LLM may occasionally produce tokens other than "A" or "B." For a sufficiently capable transformer model, however, these two tokens should account for the majority of the probability mass in the distribution for the next token. Empirically, this is indeed what we find for GPT-4o. To restore the binary case, we define the normalized inner probability of "A" over the acceptable token set as:

$$\tilde{p}(s_1 = A \mid P) = \frac{p(s_1 = A \mid P)}{p(s_1 = A \mid P) + p(s_1 = B \mid P)}. \tag{17}$$

Under this normalized distribution, both the binary entropy $H_{bin}(P)$ defined in Eq. (5) and

[7] When constructing portfolios, we aggregate the predictions for multiple news items for the same firm-day by taking the majority vote.

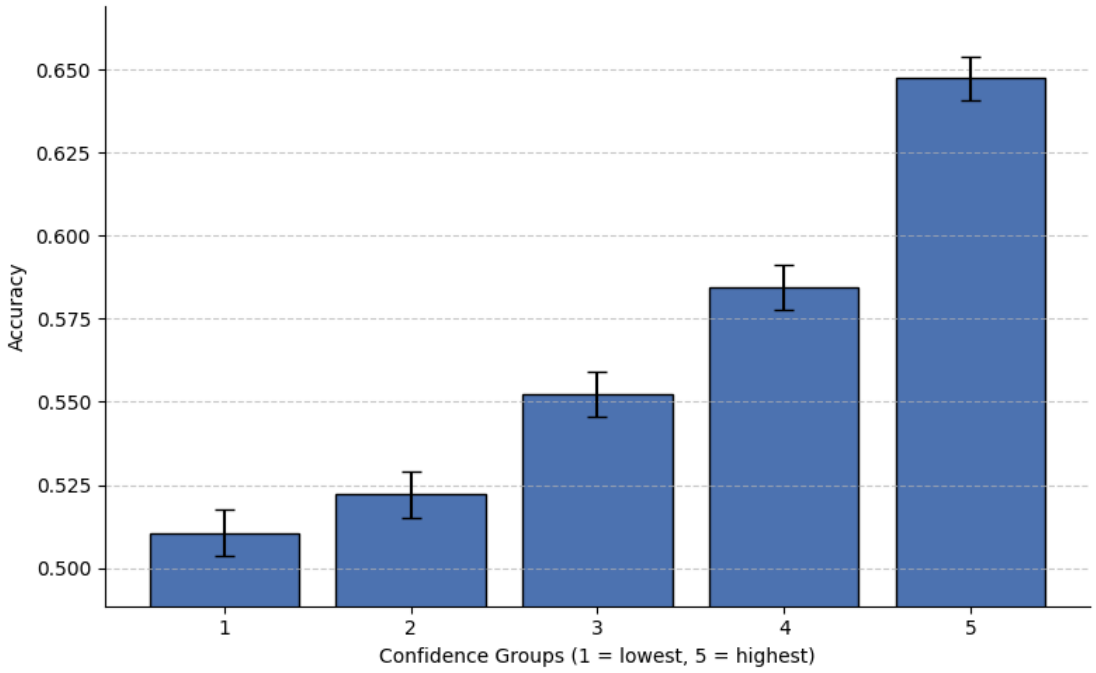

Figure 2: **Baseline Accuracy by Confidence Groups.** This figure shows the results of using prompt 1 to estimate sentiment and the associated inner probability for 100,000 randomly selected news articles. Sentiment is classified as positive when the probability of generating the first token as "A" exceeds 0.5 ($\tilde{p}(s_1 = A \mid P) \geq 0.5$). Confidence is defined as $c = |\tilde{p}(s_1 = A \mid P) - 0.5|$, and its distribution is divided into 5 quintiles from 1 (low confidence) to 5 (high confidence). For each quintile group, we report the average accuracy and 95% confidence intervals. The sample period is from October 2023, following the training cutoff of GPT-4o, to December 2024.

the inner confidence statistic $C(A \mid P)$ defined in Eq. (6) can be readily computed.

We extract the inner probabilities from GPT-4o for each news article in our sample. The technical details of this extraction are explained in Appendix D. We treat the LLM's classification of a news item as positive (having a positive impact on stock return) if the normalized conditional probability of "A" as the first token is at least 0.5, i.e., $\tilde{p}(s_1 = A \mid P) \geq 0.5$. Otherwise, the news item is classified as negative by the LLM. Next, we use the contemporaneous daily open-to-close return as ground truth, defining a news item as positive if the corresponding return is positive.[8] We then measure the accuracy of LLM predictions by comparing the sign of the realized return with the LLM's classification.

We sort the 100,000 financial news items into quintiles based on inner confidence, with

[8]With intra-day price data, one can measure the impact of news more precisely by using the stock return from the release of news article to market-closing. We opt for open-to-close return for simplicity, which also avoids the potential issues with timestamp inaccuracies and intra-day news delays.

Portfolio 1 having the lowest inner confidence and Portfolio 5 the highest. For each portfolio, we compute the average classification accuracy. Figure 2 reports the results. Accuracy increases monotonically with inner confidence: in the lowest-confidence portfolio, the LLM's predictions are correct approximately 51% of the time, almost indistinguishable from a random guess, whereas in the highest-confidence portfolio, accuracy reaches 64.7%. The difference in average accuracy between Portfolio 5 and 1 is highly statistically significant, with a t-statistic of 28.07.

The monotonic relationship between inner confidence and classification accuracy documented above indicates that the uncertainty implied by the LLM's inner probabilities is informative about objective predictive uncertainty. This connection can be formalized through a bias-variance decomposition. For inner confidence to be positively associated with predictive accuracy, it is sufficient that the subjective uncertainty of LLM beliefs be positively related to the variance of the LLM predictor around the true conditional mean, and that any systematic bias in LLM beliefs does not increase too rapidly as subjective uncertainty declines. Under these conditions, higher inner confidence (lower uncertainty) corresponds to lower expected prediction error in the mean-squared-error sense, making LLM uncertainty quantification economically meaningful.

Note that the purpose of this experiment is not to test whether neural language models' subjective beliefs are unbiased in their mean forecasts or properly calibrated in their confidence.[9] Indeed, several recent studies have documented various biases for modern LLMs (Bybee, 2023; Fedyk et al., 2024). Rather, our findings indicate that, despite such biases, quantifying the uncertainty of LLM beliefs provides useful information about the model's comparative reliability across observations, which can enhance decision-making based on LLM forecasts. We explore this implication in the next section.

3.2 Confidence-Based Portfolio Strategies

In this Section, we further evaluate the economic significance of quantifying LLM uncertainty using inner probabilities by examining whether they enhance the profitability of trading strategies built on LLM predictions. Following Lopez-Lira and Tang (2023), we prompt an LLM to evaluate whether overnight news is likely to have a positive or negative short-term impact on a stock's performance and construct long-short portfolios based on these predictions. Different from Lopez-Lira and Tang (2023), we examine how strategy performance varies with the inner confidence of the LLM's predictions.

[9] For that purpose, in the binary setting one could run regression of actual outcomes on the inner probabilities, $1_{\{r_i>0\}} = \alpha + \beta\tilde{p}(s_1 = A \mid P) + \epsilon_i$, and test whether $\alpha = 0$ and $\beta = 1$.

Table 2: **Summary Statistics of the Portfolio Exercise Sample**
The table below presents the summary statistics for the dataset of news articles used in the exercise presented in Section 3.2, sorted by firm size (market equity) deciles. The breakpoints for the size portfolios are based on the NYSE universe (Fama and French, 1993). For each decile, we report the number of firms, number of news items, the median firm size, and the average and median number of news items per firm.

Size Deciles	**Small**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Large**
Firms	2,852	1,236	1,019	878	685	609	506	458	344	279
News Items	68,439	35,063	29,918	30,055	24,219	23,737	26,159	30,051	33,914	60,193
Med ME ($bn)	0.1	0.6	1.2	2.1	3.3	5.2	7.9	13.9	30.8	122.9
Avg News/Firm	24.0	28.4	29.4	34.2	35.4	39.0	51.7	65.6	98.6	215.7
Med News/Firm	13	15	17	21	20	24	32	41	70	147

As in the previous experiment, we use news articles from *Reuters* for this experiment. Instead of intra-day news, we follow Lopez-Lira and Tang (2023) and select all overnight news (with a timestamp after 4pm on day $t-1$ and before 9am on day t). As in Section 3.1, we require that each news article must be related to a single firm, and that the article must contain between 100 and 5,000 characters. Table 2 presents the sample summary statistics by NYSE market capitalization decile (Fama and French, 1993). We then ask GPT-4o to classify each news item as either "positive" or "negative" for the stock price in the short term. An important difference lies in the prompt — not only do we ask for the sentiment classification, but we also elicit the model's self-declared confidence in its prediction.

Specifically, we use Prompt 2, which produces text structured as a sentiment label, followed by the delimiting character "|" and a numerical value representing the model-declared confidence (e.g., "A|0.7"). In addition to being compatible with the principle proposed in Section 2, this format enables a direct comparison, within a single prompt, between the model's inner confidence, computed from the distribution of the first token to obtain $\tilde{p}(s_1 = A \mid P)$, and its declared confidence, obtained by parsing the number after the "|" symbol. The latter, used in prior work (see, e.g., Bybee, 2023; Fedyk et al., 2024), offers an intuitive means of eliciting model confidence without accessing token-level probabilities. Here, we compare inner confidence and declared confidence through their effects on portfolio performance. We further study *declared confidence* and the potential biases introduced by the feedback loop in Section 5.

Next, we construct five long-short strategies based on daily open-to-close returns:[10]

- Baseline Portfolio: An equal-weight portfolio in which the long leg on day t includes all stocks with a net positive signal from the overnight news released after market close

[10] Appendix E presents a more comprehensive description of the portfolio construction methodology.

```
Forget all your previous instructions.

You are a financial expert with stock recommendation experience.
Is this financial news good or bad for the stock price of {target} in the
 short term?
Answer A if good news, B if bad news. After you answer with A or B,
please write the symbol | and then a number between 1 (very confident)
and 0 (not confident) to indicate how confident you are in your answer.
Please only write the symbol "|" and the number, nothing else.

ARTICLE:
{headline}
{body}
```

Prompt 2: **Joint Declared and Inner Confidence.**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm and to assess its own confidence in the prediction. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

on day $t-1$, while the short leg includes all stocks with a net negative signal. Since there can be multiple overnight news items for the same firm, we aggregate them into a net signal for each firm based on the majority class of LLM assessments (positive or negative) of the news items.

- Confidence-based Portfolios: For both *inner* and *declared* confidence measures, we construct two portfolios:
  - High-Confidence Portfolio: An equal-weighted portfolio formed in a similar way as the baseline portfolio, but only using news items whose inner or declared confidence exceeds a specified threshold. The threshold is recalibrated monthly via a rolling-window procedure, selecting the value that maximizes the Sharpe ratio over the preceding three months.
  - Low-Confidence Portfolio: An equal-weighted portfolio formed using news items whose inner or declared confidence falls below the same dynamic threshold.

Table 3 reports the results of the portfolio performances. The *high-inner-confidence* portfolio earns an average annualized excess return of 49.62% and a Sharpe ratio of 5.15, substantially higher than those for the *Baseline* portfolio, which have average return of 39.56%

Table 3: **Performance of Confidence-Based Portfolios**
This table reports out-of-sample performance of three long-short investment strategies based on LLM news classification, inner confidence, and declared confidence. The *Baseline* portfolio includes all news; the *High-Confidence* portfolios are based only on news associated with either high inner or declared confidence, while the opposite is true for *Low-Confidence* portfolios. We report annualized mean returns (%), volatility (%), Sharpe ratio, alpha (%) based on FF5 plus momentum (Mom) as well as FF5 plus momentum (Mom) and short-term reversal (ST-Rev), and average number of stocks. The sample is from October 2023, following the training cutoff of GPT-4o, to December 2024. All portfolios are equally weighted and rebalanced daily.

	Inner Confidence		**Declared Confidence**		**Baseline**
	High Conf	Low Conf	High Conf	Low Conf	
Mean	49.62	−9.80	40.83	−3.55	39.56
Std	9.63	25.19	11.01	32.83	9.34
Sharpe ratio	5.15	−0.39	3.71	−0.11	4.24
α: FF5+Mom	46.78	−10.26	37.58	−10.63	37.17
α: FF5+Mom+ST-Rev	48.66	−5.12	39.08	−9.83	38.98
Avg # stocks	481.50	111.82	452.42	130.82	525.71

and Sharpe ratio of 4.24.[11] A test of the difference between the two Sharpe ratios, using the approach suggested by Jobson and Korkie (1981) and Memmel (2003), is statistically significant, with t-statistic of 2.84. This improvement results from filtering out on average 18% of the news articles with low inner confidence.[12] In contrast, the *low-inner-confidence* portfolio earns a negative average annualized return (-9.80%, which is sizable in magnitude but statistically insignificant) and a Sharpe ratio of -0.39, which is insignificantly different from zero (t-statistic is 0.41). The same patterns also hold for the risk-adjusted returns based on the alpha from Fama and French (2015) five-factor model (FF5) plus momentum, as well as the five-factor model plus momentum and short-term reversal (see Table A2 in Appendix F.1 for details).

Unlike inner confidence, the declared confidence obtained through LLM prompting actually weakens the portfolio performance. The *high-declared-confidence* portfolio earns an average annualized excess return of 40.83%, about the same as that for the *Baseline* portfolio (the same pattern also holds for the alphas from the two different factor models), but the

[11] To make our results comparable to Lopez-Lira and Tang (2023), we follow their approach and retain all stocks in the main specification. As a robustness check, Appendix F.3 reports portfolio performance under different market-capitalization filters ranging from no restriction to above the 20th NYSE size percentile. Across all configurations, the high-inner-confidence portfolio maintains a higher Sharpe ratio than both the high-declared-confidence and baseline portfolios.

[12] Since the threshold is re-calibrated monthly, the share of excluded news items varies over time.

Sharpe ratio drops from 4.24 to 3.71, although the difference is not statistically significant (t-statistic is -1.28). These results are consistent with the interpretation that declared confidence is randomly excluding news items, which does not improve the average returns based on the remaining signals but does make the portfolio less diversified. The cutoff threshold, which is calibrated using the same rolling-window methodology, is on average at the 5th percentile for the full sample. As a result, the average return for the *low-declared-confidence* portfolio, while negative, is very noisy.[13] It is also worth noting that an important reason the average cutoff threshold for declared confidence is so low is due to the fact that the distribution of declared confidence tends to be highly concentrated and non-smooth. We examine this issue in more detail in Section 5.

Note that the LLM may occasionally fail to follow the instruction to produce a parsable declared confidence (it occurred for about 2% of our sample). In such cases, we remove the observation from the declared confidence portfolios. By contrast, the inner confidence measure can be computed regardless of whether "A" or "B" appears among the tokens with the highest conditional probabilities. This reliability represents another advantage of our inner-probability-based methodology over declared confidences.

While the average return and Sharpe ratio of the high-inner-confidence portfolio are strikingly high, transaction costs will also be high due to daily rebalancing. To further examine the sources of superior performance of the high-inner-confidence portfolio, we sort firms in the high-inner-confidence portfolio into two halves based on several characteristics including firm size, media and analyst coverage, and liquidity, and examine the conditional long-short returns.

Table 4 reports the results. The Sharpe ratio of the long-short strategy is higher among firms that are smaller, less liquid (as proxied by higher bid-ask spreads, higher Amihud (2002) measure, and lower turnover), and receive less media or analyst coverage. Moreover, across all six splits, the majority of the long-short return appears to be generated by the short leg. These findings are consistent with mispricing arising from market underreaction to news, which is sustained by trading and informational frictions.

An interesting finding is that, among stocks with high media or analyst coverage, both the long and short legs generate negative returns. The existing literature has documented that stocks with high media coverage tend to earn lower returns at longer horizons (see e.g.,

[13] One subtle detail about Table 3 is that the baseline portfolio average return is not a simple weighted average of the returns of the high- and low-confidence portfolios. This is because the partition between high and low confidence (both inner and declared) is performed at the news item level and not firm level. Since a firm can have multiple news articles in a day, that firm could appear in the high- and low-confidence subsets simultaneously. This also explains why the sum of the average number of firms in the high- and low-confidence portfolios may exceed the average number of firms in the baseline portfolio.

Table 4: **Sub-Sample Sharpe Ratios of High-Confidence Portfolios**
This table provides a breakdown of the "high-inner-confidence" portfolio performance presented in Table 3. We partition the daily sample based on analyst coverage, illiquidity, and market capitalization. "Higher" and "Lower" refer to the split based on the indicated criterion. For each criterion and split, we report the Sharpe ratios of portfolios constructed using the top and bottom halves of the respective split. The first two columns present results for the Long-Short strategy, the next two for the short leg, and the final two for the long leg. The sample period is from October 2023 to December 2024. All portfolios are equally weighted and rebalanced daily.

	Long-Short		Long Leg		Short Leg	
Split Criterion	High	Low	High	Low	High	Low
Size (market cap)	2.350	5.019	1.563	0.335	0.122	-3.247
Bid-ask spread	4.368	3.183	0.087	1.619	-3.059	-0.459
Amihud illiquidity	5.542	1.961	1.025	0.472	-3.202	-0.832
Turnover	2.728	5.563	-0.227	1.568	-2.140	-2.375
# News articles	1.202	3.759	-2.181	1.245	-4.631	-1.601
Analyst coverage	1.916	5.261	-3.973	0.869	-4.293	-2.365

Fang and Peress, 2009, who study monthly rebalanced strategies). Our setting differs in that we focus on the open-to-close return on the trading day following the news release. If attention-driven buying (Barber and Odean, 2008) following good news leads to overreaction in the opening price, this could explain the negative return on the long leg. At the same time, the results suggest that the effects of attention-driven trading may be asymmetric: opening prices appear to underreact following bad news, potentially reflecting short-selling constraints and the disposition effect (Shefrin and Statman, 1985).

Robustness checks for portfolio results. The portfolio sample period is necessarily short because we begin the main analysis after the GPT-4o training cutoff to eliminate potential look-ahead bias. To assess the robustness of the documented effects of inner confidence, we conduct several additional exercises.

First, we expand the portfolio analysis to include the year immediately preceding the training cutoff (October 2022 to September 2023), thereby increasing the time-series variation while acknowledging the possibility of look-ahead biases. The results are reported in Table A3 in Appendix F.2. The ranking of strategies is unchanged: portfolios conditioned on high inner confidence continue to outperform the baseline portfolios, which in turn outperform portfolios based on declared confidence. Moreover, low-inner-confidence portfolios do not generate statistically significant abnormal returns.

These results are consistent with our findings in the post-cutoff sample. Still, we should interpret the in-sample results with caution. Higher inner confidence may be associated with higher probability of recall, i.e., the LLM remembering the news and its market impact from the training data, which could be responsible for the superior performance. We examine the link between inner confidence and recall specifically in Section 4, where we show that there is no strong evidence that the LLM is able to recall the market impact of specific news items in our pre-training-cutoff sample.

Second, our cross-sectional results in Table 4 show that the profitability of high-inner-confidence signals concentrates in economically plausible subsamples, including smaller, less liquid, and less media- or analyst-covered firms. These patterns are consistent with limits to arbitrage and slower information diffusion, suggesting that the predictive gains are not random but instead align with established mechanisms in the anomaly literature. At the same time, Table A4 in Appendix F.3 shows that the performance of the high-inner-confidence portfolio is not entirely driven by microcap stocks, as the Sharpe ratio remains sizable and higher than those of the baseline and declared-confidence portfolios even when we restrict to firms with market capitalization above $100 million or above the 20th NYSE size percentile.

Third, we test the sensitivity of results to the confidence-threshold choice. In the main analysis, thresholds are selected dynamically based on past performance. To ensure that the findings are not driven by this choice, Appendix F.4 reports results using a range of fixed thresholds for both inner (Table A5) and declared confidence (Table A6). Across these alternative specifications, the qualitative conclusions remain unchanged: conditioning on inner confidence improves classification accuracy and portfolio performance relative to both the unconditional strategy and strategies based on declared confidence.

Taken together, these exercises suggest that the documented performance of portfolios conditioned on inner confidence is not an artifact of a short post-cutoff sample, threshold tuning, or a narrow subset of securities.

3.3 Confidence Determinants

Having examined the economic value of quantifying the uncertainty of LLM outputs using inner probabilities, we now turn to the determinants of LLM confidence, which provide an alternative lens for assessing whether the uncertainty of LLM's beliefs reflects underlying predictive uncertainty. Using the same sample as in the portfolio trading experiment in

Section 3.2, we estimate the following logistic model:

$$\begin{aligned}\text{Logit}(\Pr(C_{i,j,t} \geq c_k)) = \gamma_j + \beta_1\, r_{j,t\to t+30} + \beta_2\, \sigma_{j,t\to t+30} + \beta_3\, \text{abn\_volume}_{j,t} \\ + \beta_4\, \text{nb\_news}_{j,t} + \beta_5\, \text{nb\_news}_t + \beta_6\, \text{vix}_t. \end{aligned} \tag{18}$$

Here, $C_{i,j,t}$ denotes the inner confidence for LLM's assessment on news item i for firm j on day t, and c_k denotes the kth percentile of the inner confidence distribution in the full sample. Thus, we examine what firm and market variables associated with each news item are linked to high inner confidence (with a positive β coefficient) in the LLM's assessment. Note that we opt for a logistic regression specification to focus on the rankings and not the levels of inner confidence.

Among the firm and market variables considered, $r_{j,t\to t+30}$ denotes the average daily return for firm j over the 30 days following the news release, which we use as proxy for market sentiment about the news. As a proxy for the market's reflection of news complexity, we include $\sigma_{j,t\to t+30}$, the realized daily volatility of firm j over the 30 days following the news. News release followed by elevated price volatility may indicate that the directional impact of the news is uncertain, and we examine whether LLM inner confidence is correspondingly lower in such cases. Related, $\text{abn\_volume}_{j,t}$ measures abnormal trading volume for firm j on day t, defined as the deviation of raw volume from its 60-day trailing mean, scaled by the corresponding standard deviation. Elevated abnormal trading volume may reflect heightened investor disagreements, which could also be a sign that the news is difficult to interpret.

Next, we include two measures for the intensity of information flow: $\text{nb\_news}_{j,t}$, the number of Reuters headlines mentioning firm j on day t, and nb_news_t, the total number of Reuters headlines across all firms on day t. These variables allow us to examine how LLM uncertainty varies during periods of high information volume. We also include the CBOE VIX Index on day t, vix_t, as a proxy for market-wide uncertainty. Finally, we incorporate firm fixed effects, γ_j, so that the estimates reflect within-firm variation in inner confidence over time. For ease of interpretation, all independent variables are standardized to have unit standard deviation. Standard errors are computed using two-way clustering at the firm and date levels.

Table 5 presents the results. The LLM is more likely to assign high confidence to news articles that are followed by more positive returns. Consistent with intuition, LLM inner confidence is likely to be higher when the news is followed by lower volatility and lower abnormal trading volume, suggesting that news that is less ambiguous in its price impact or results in less disagreement among investors are easier for the LLM to interpret. The number of firm-specific news items on the same day is positively associated with inner confidence,

Table 5: **Determinants of High Inner-Confidence Classification**
The table reports logit regression coefficients from Eq. (18), where the dependent variable is an indicator equal to one if the LLM's inner confidence for a news item exceeds the 90$^{\text{th}}$, 80$^{\text{th}}$, or 70$^{\text{th}}$ percentile of the distribution. Explanatory variables include 30-day forward average daily return and realized volatility, abnormal trading volume, firm-specific Reuters headline account, market-wide headline count, and the VIX index. Explanatory variables are all normalized by their standard deviation. All regressions include firm fixed effects, and standard errors are clustered by firm-date. Asterisks indicate statistical significance at the 10% (*), 5% (**), and 1% (***) levels.

High-Confidence Threshold (c_k)	**90th**	**70th**	**50th**
$r_{j,t\to t+30}$	0.023*	0.034***	0.046***
	(0.011)	(0.009)	(0.011)
$\sigma_{j,t\to t+30}$	-0.034**	-0.051***	-0.078***
	(0.013)	(0.015)	(0.016)
abn_volume	-0.038**	-0.064***	-0.106**
	(0.012)	(0.018)	(0.036)
$nb\_news_{j,t}$	0.010	0.018	0.046**
	(0.012)	(0.015)	(0.017)
nb_news_t	-0.032**	-0.076***	-0.078***
	(0.012)	(0.012)	(0.011)
vix_t	-0.010	-0.023	-0.013
	(0.012)	(0.013)	(0.013)
Firm FE	Yes	Yes	Yes
Observations	327,625	340,792	344,266
Squared Correlation	0.0586	0.1376	0.1629
Pseudo R^2	0.0795	0.1191	0.1276

although the coefficient is only significant at the 50th percentile threshold. This could be in part due to limited sample size: With a high percentile cutoff, only a small number of firms in the high-confidence group would have multiple news items on the same day. In contrast, the total number of news items across all firms on the same day is significantly negatively associated with inner confidence. Days with large amount of news flows are likely those with market- or economy-wide events. Firm level news during such periods may be more difficult for the LLM to interpret due to their complexity and the need for broader context.

Next, we examine the relationship between inner confidence and the topical content of news articles. To assign topics to each news item, we follow Bybee, Kelly, Manela, and Xiu (2020), who employ a Latent Dirichlet Allocation (LDA) model trained on the *Wall Street Journal* to identify 130 distinct topics. Their publication of "word weights per topic" enables efficient topic assignment to new texts. While these weights are insufficient for precise probabilistic assignment across all topics, they are adequate for identifying the most

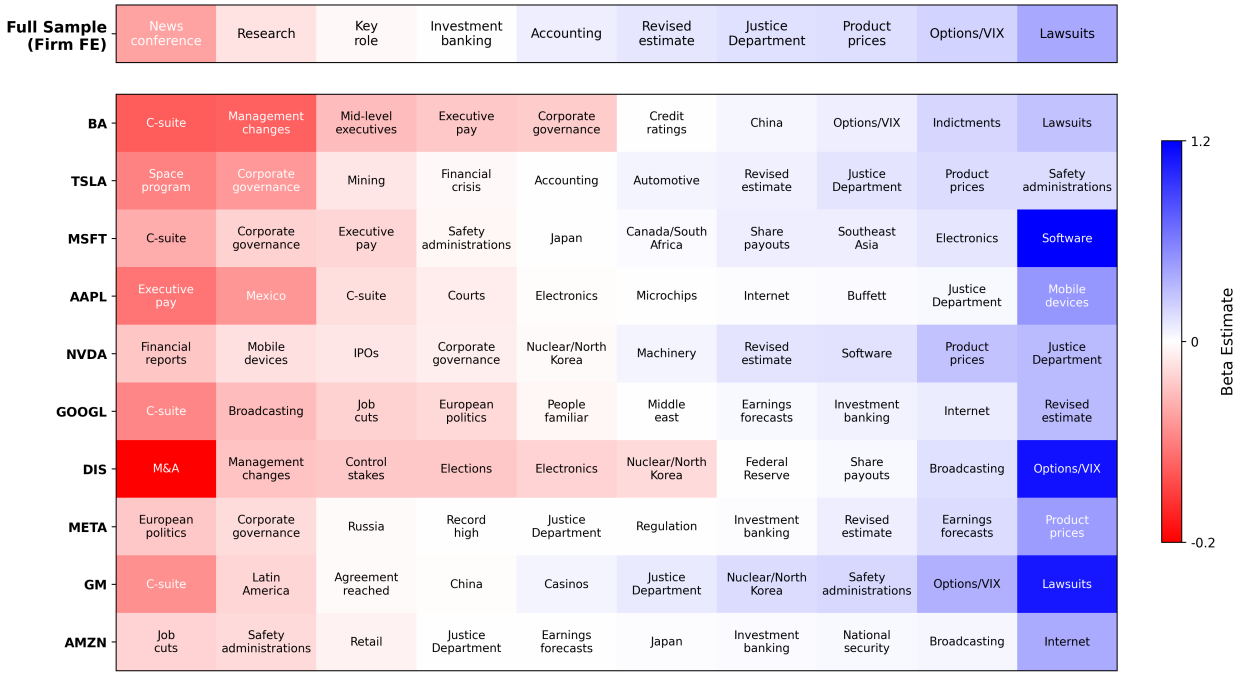

Figure 3: **Lasso-Selected Topic Coefficients vs. Confidence Decile.** This figure presents the topics most strongly associated with the LLM's inner confidence. We estimate the penalized least squares model in Eq. (19), predicting the confidence decile y_i using topic dummies X_i, and tune the penalty parameter λ to ten selected topics. The top panel displays the full-sample coefficient estimates $\hat{\beta}$, while the lower panels show firm-specific estimates for the ten firms with the largest number of news articles. Blue tones indicate larger positive coefficients (associated with increased model confidence), while red tones represent larger negative coefficients (associated with decreased model confidence).

dominant topic in a given article.

Specifically, each news item is associated with a sparse 130-dimensional vector X, where the k^{th} component equals 1 if topic k is the most strongly associated with that article, and 0 otherwise. We then perform a Lasso regression for the full panel of news articles, regressing normalized inner confidence $\widehat{C}_{i,j,t}$, which measures the decile of $C_{i,j,t}$ ranging from 1 (lowest confidence) to 10 (highest confidence), on a firm fixed effect and the topic vector $X_{i,j,t}$:

$$min_{\beta\in\mathbb{R}^p}\frac{1}{N}\sum_{i=1}^{N}\left(\widehat{C}_{i,j,t}-\alpha_j-X_{i,j,t}^{\top}\beta\right)^2+\lambda\|\beta\|_1, \tag{19}$$

where λ is a regularization parameter that we raise gradually until the model retains only 10 non-zero coefficients for $\hat{\beta}$, corresponding to the ten topics most predictive of confidence levels. We also run similar Lasso regressions at firm level, which help identify the top ten topics that are most relevant for firm-level LLM uncertainty.

The top row of Figure 3 summarizes the results for the full sample of news. Red hues denote topics associated with low confidence (negative $\hat{\beta}_j$), while blue hues indicate topics associated with high confidence (positive $\hat{\beta}_j$). The results show that news topics related to "lawsuits" and "options/VIX" tend to be associated with higher levels of inner confidence, while topics about "news conference" and "research" tend to be associated with low inner confidence. These patterns are economically intuitive: discrete legal events often carry clearer directional implications, whereas conference commentary and research updates may be more nuanced or ambiguous.

Next, we apply the analysis above to the ten firms with the highest amount of news coverage in our sample. This analysis reveals firm-level heterogeneity in information complexity across news subjects. For example, news about *software* was strongly associated with high inner confidence for Microsoft, which was aggressively integrating Copilot into its product ecosystem during our sample period. News about *lawsuits* for GM was also assessed with high confidence, as the firm faced class-action suits from its investors. Conversely, other topics show a strong negative association with inner uncertainty. These include "C-Suite" and "management changes" for Boeing, stemming from the leadership instability during its prolonged safety crises, and "M&A" for Disney, which went through complex negotiations to merge its Indian media assets with Reliance Industries in 2024. Given the results from regression (18), it would be interesting to explore whether cross-sectional and time-series variation in exposure to ambiguous news topics can help explain differences in realized price volatility and trading volume across firms and over time. We leave a formal analysis of this channel to future research.

3.4 Precision of the Prior and Signal

Neural language models, through their pre-training, are implicitly endowed with a set of prior beliefs, including beliefs about firm performances. In the context of news classification, we can view a news article as a signal, and the conditional distribution based on the news and prompt can be viewed as the posterior belief. Heuristically, we can view the LLM output and conditional entropy as the mean (or mode) and variance of this posterior distribution, respectively. While the updating process is opaque and unlikely to conform to the Bayes rule in a strict sense, does it at least exhibit qualitative traits of Bayesian updating?

This question motivates the following experiment. We try to alter either the model's prior precision about a firm or the perceived precision of the signal, and examine how the uncertainty of the resulting posterior belief is affected. The intuition comes from Bayesian updating under Gaussian prior and signal. The Bayes rule implies $1/\sigma^2_{post} = 1/\sigma^2_{prior} + 1/\sigma^2_s$,

Table 6: **Aggregated Inner Confidence Across Eight Prompting Conditions**
This table reports the mean and variance of the LLM's inner confidence for three semantic contexts (Analyst Prior, Analyst Signal, and Noise), each evaluated under a Consensus and a Disagreement variant. For each experiment, we report the LLM's inner confidence variance, mean and its "corresponding percentile," which indicates the percentile rank of the mean confidence within the unconditional baseline distribution. The final row lists the t-stats from two-sided tests that compare the Consensus and Disagreement means within each context.

	Analyst Prior		**Analyst Signal**		**Noise**	
Conditioning	consensus	disagreement	consensus	disagreement	consensus	disagreement
Mean	0.480	0.476	0.481	0.478	0.485	0.484
Percentile	65.31%	58.57%	64.60%	55.88%	60.96%	57.25%
Variance	0.005	0.005	0.005	0.005	0.003	0.003
t-stat for difference	7.250		7.860		1.698	

with σ^2_{post}, σ^2_{prior}, and σ^2_s denoting the posterior variance, the prior variance, and the variance of the signal, respectively. In that case, when either the prior precision or the precision of the signal rises (falls), so does the precision of the posterior.

Specifically, we design three pairs of prompts intended to either positively or negatively perturb the precision of the prior or signal.

- The first pair of prompts aims at modifying the model's prior uncertainty. One asserts that analysts mostly agree on the firm's prospects, while the other asserts that there is significant disagreement.
- The second pair aims at modifying the variance of the signal by asserting that analysts either agree or disagree on the interpretation of the recent news.
- The third set serves as a placebo. Here, we prepend Prompt 1 with words associated either with high confidence, "confidence, clarity, conviction, precision, certainty," or low confidence, "doubt, anxiety, mistrust, uncertainty, hesitation." to assess whether semantically vacuous keywords can influence the model's internal confidence.

The specific prompts are detailed in Appendix G. We employ these prompts in the news classification exercise in Section 3.1. Table 6 summarizes the results. We report the mean and variance of the inner confidence under perturbed prompts. In addition, we also report their corresponding percentiles in the original baseline distribution without conditioning. This is because the inner confidence distribution is clustered near 0.5, making the difference in mean confidence harder to interpret. As we have shown in Section 3, despite the clustering, the ordinal ranking carries valuable economic information.

We find that providing the model with information regarding analyst consensus, whether pertaining to prior beliefs or posterior interpretation, induces a significant shift in the model's

inner confidence. When informed that analysts are in agreement, without reference to the direction of the consensus, the model exhibits greater confidence in its classification. Conversely, when informed of analyst disagreement, the model's inner confidence diminishes. The difference in inner confidence under the two prompts is highly statistically significant.

This behavior is qualitatively consistent with that of a Bayesian learner: the model integrates the additional information and modulates its internal belief structure, treating consensus as an informative signal about the underlying classification difficulty. By contrast, introducing affectively charged but semantically irrelevant terms, serving as either positive or negative noise, has no measurable effect on the model's confidence. While such cues might influence human judgment, their irrelevance to the classification task renders any reaction to them irrational. In this respect, the model's behavior aligns with rational decision-making principles.

4 Confidence and Look-Ahead Bias

Look-ahead bias is a crucial methodological concern in economic and financial analyses involving LLMs (see, e.g., Sarkar and Vafa, 2024; Lopez-Lira et al., 2025). Commercial LLMs trained on extensive datasets may inadvertently "recall" information unavailable at the prediction time. Such recall can bias results and produce false positives, inflating performance in classification and prediction tasks. In this Section, we examine the connection between model memory and internal probabilities, focusing on months near the training cutoff.[14]

To identify this relationship clearly, we first conduct a test in a controlled setting. In the spirit of Didisheim et al. (2025) and Lopez-Lira et al. (2025), we query the LLM with Prompt 3. The prompt evaluates the LLM's recall capability in a parsimonious way. In the absence of contextual cues for predicting returns, any accurate recall must arise solely from look-ahead bias. Following the methodology of Didisheim et al. (2025), we query the LLM about daily returns for eleven major indices between 2018 and 2023 with temperature set to zero.

Treating the "recalled" return as a forecast and the true return as the ground truth, we use the absolute error as a proxy for pure look-ahead bias. Since the LLM receives no additional information, a low absolute error implies that it successfully retrieved the true return from its internal "memory." Such memorization capacity could inadvertently introduce look-ahead bias in downstream applications.

[14] All other tests presented in the rest of the paper are conducted out of sample, using only observations that occur after the model's training cutoff date.

```
What was the daily return of index {index name} on {date}?
Answer with only a number.
```

Prompt 3: **Pure Look-Ahead Bias**
The prompt above is designed to extract the LLM's pure look-ahead bias, defined as the ability to recall the specific data point without any context. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the index's name, ticker, and date.

For each recollection, the API returns the probability that the model assigns to the token it selects, $\tilde{p}(s^* \mid P)$. This effectively reduces the recall task to a binary classification: either s^* matches the true return or it does not. The model's confidence in its selection can therefore be measured using the same binary framework as in the rest of the paper, with the confidence measure defined in Eq. (6):[15]

$$C(\text{“Recall”}) \;=\; \left|\tilde{p}(s^* \mid P) - \tfrac{1}{2}\right|. \tag{20}$$

Figure 4 illustrates the relationship between absolute recall error and average confidence. Specifically, we sort the observations by absolute error scaled by the daily volatility of the index, and group them into buckets representing 4% of the sample for visualization. For instance, a value of 1.37 means that the absolute recall error is on average equal to 1.37% of the index's daily volatility. The x-axis reports the average absolute error per group, while the y-axis shows the corresponding average C within each group.

The figure shows a clear difference between high recall (absolute error below 0.03% of daily volatility), which is associated with average confidence exceeding 45%, and an average confidence of around 25% for more inaccurate recalls. The sharp angle in the plot, and the fairly stable distribution for the high and low absolute errors, suggest a clear association between look-ahead bias and inner confidence.

The previous exercise shows a relationship between pure look-ahead and inner confidence. We next analyze the relationship between inner confidence derived from processing news and the look-ahead bias. Specifically, we compute a daily average confidence score obtained by processing all news items in our sample using Prompt 1, following the procedure described in Section 3. Figure 5 displays this average daily confidence over the one-year period before

[15] A recollection may span multiple tokens when the tokenizer splits the response into sub-units. For instance, the value 0.1 could be generated as a single token or as three separate tokens (“0″, “.″, “1″). In the multi-token case, we replace the single-token probability with the joint probability of the full sequence, $\tilde{p}(S^* \mid P) = \prod_{i=1}^{L} \tilde{p}(s_i^* \mid P, s_{<i}^*)$, and apply the same formula.

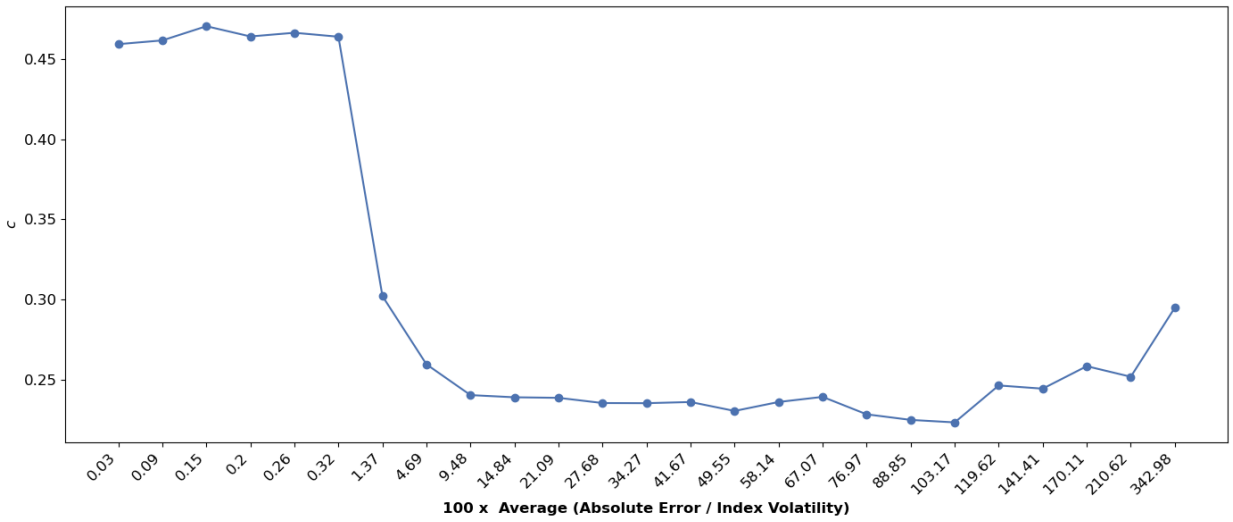

Figure 4: **Recall Confidence by Accuracy.** The horizontal axis shows observation buckets sorted by recall accuracy, which is defined as the absolute recall error scaled by the daily volatility of the index (left = most accurate, right = least accurate). The vertical axis indicates the corresponding average inner confidence for each bucket. Accuracy measures the LLM's ability to correctly retrieve a specific daily index return occurring before its training cutoff, without receiving any related information. Accurate recalls indicate the presence of look-ahead bias (Didisheim et al., 2025).

and after the GPT-4o learning cutoff in October 2023.[16]

If a strong look-ahead bias were present, implying substantial reliance on memorization of news or associated returns, we would expect higher average confidence in the pre–learning-cutoff period relative to the post-cutoff period. Instead, we observe a small but statistically insignificant *increase* in inner confidence after the learning cutoff. This finding aligns with Didisheim et al. (2025), who report that the look-ahead bias is relatively minor for daily-frequency recollection of individual firm returns, and with Glasserman and Lin (2023), who document that the *distraction effect*—where general knowledge of the companies mentioned interferes with sentiment measurement—can be stronger than the look-ahead bias.

A natural question is whether this pattern differs for firms more likely to appear in the LLMs' training data. To examine this, in Appendix H we replicate the analysis from Figure 5 for the top 10% of firms ranked by annual news coverage and market capitalization. Intuitively, these subsets are more likely to be represented in the LLMs' training corpus and therefore more exposed to potential look-ahead contamination (Didisheim et al., 2025). However, the resulting patterns closely resemble those observed for the full sample shown in

[16]The learning cutoff is defined as the date corresponding to the last observation included in the model's training dataset. This information is typically disclosed by the provider.

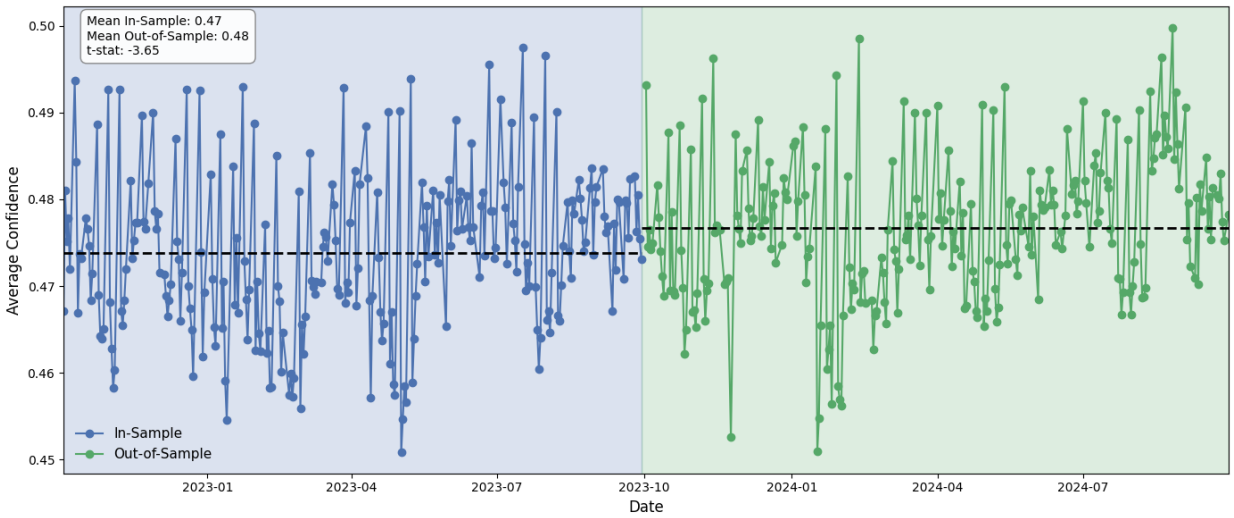

Figure 5: **Daily Average Inner Confidence In- and Out-of-Sample.** The figure shows the average daily inner confidence one year before and after GPT-4o training sample cutoff in October 2023. Shaded areas indicate in-sample (blue) and out-of-sample (green) periods. Dashed lines show mean confidence per period, with top-left inset reporting means and t-statistic.

Figure 5. This finding is consistent with Lopez-Lira et al. (2025), who report no systematic relationship between look-ahead bias and firm characteristics or news coverage.

Finally, we remove daily averaging and compare the distribution of individual inner-confidence values in the year before and after the learning cutoff. Figure 6 presents the distribution of news-level inner confidence for the in-sample period (left) and the out-of-sample period (right). Once again, the in- and out-of-sample distributions do not exhibit patterns consistent with a strong influence of look-ahead bias. These plots also highlight that inner probabilities tend to cluster near 0% and 100%, leading inner confidence to concentrate around 0.5. This clustering complicates the cardinal interpretation of inner probabilities. Nevertheless, these probabilities remain economically meaningful when interpreted ordinally. For example, although the difference between 0.490 and 0.499 may appear negligible, the relative ranking carries important economic information, as discussed in Section 3.

Overall, the results in this section indicate that the internal confidence can serve as a proxy for the risk of look-ahead bias contamination. Moreover, the findings suggest that this risk remains relatively low for current LLMs when applied to processing daily information from financial news about individual stocks.

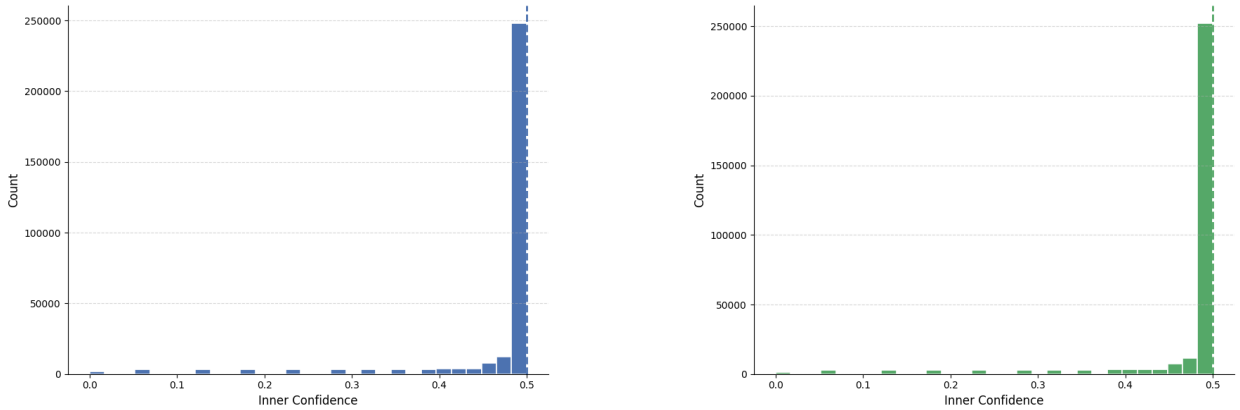

Figure 6: **Inner Confidence Distributions (Non-Ranked).**
The figure shows the distribution of inner confidence in absolute terms. Left: in-sample distribution (blue) with median dashed line. Right: out-of-sample distribution (green) with median dashed line.

5 Declared Confidence

In Section 3, we run a horse race between inner confidence and declared confidence and show that only the former enhances the performance of the LLM-prediction-based portfolio strategy. In this section, we further examine the properties of declared confidence as an uncertainty measure.

While declared confidence and inner probabilities may appear similar at first glance, the extraction method introduces two key sources of discrepancy. First, the LLM cannot directly "observe" its inner probabilities or inner confidence. When asked about its declared confidence, it simply performs the exercise of generating another token following the primary output. Second, inner confidence is computed from the conditional distribution $p(s \mid P)$, which only depends on the news prompt P. In contrast, the conditional distribution of the token S_2 for declared confidence, $p(S_2 \mid s_1, P)$, depends not only on the news prompt P, but also on which prediction token s_1 is drawn. As detailed in Section 2, the feedback loop can amplify the uncertainty about S_2 in multi-token generation. Moreover, it can also systematically bias the prediction for declared confidence depending on the decoding outcome of the primary output.

We test for this bias with an experiment that isolates the impact of the additional conditioning. Using Prompt 2, we construct a dataset of matched inner probabilities, answers, and declared confidence levels, and proceed as follows:

- We randomly select 500 news items, half of which have positive contemporaneous return, $r_{i,t} > 0$, the other half with negative return, $r_{i,t} \leq 0$, and we require that

the inner probability of the news item being positive news falls within $0.4 \leq \tilde{p}(s_1 = A \mid P) \leq 0.6$. These are cases where the LLM has low inner confidence (with inner confidence $C \leq 0.1$).

- We rerun the prompt 100 times for each of the 500 news articles, generating a total of 50,000 observations.[17]
- We estimate the following regression:

$$D_{i,j} = \gamma_j + \beta A_{i,j} + \epsilon_{i,j}, \tag{21}$$

where $D_{i,j}$ represents the declared confidence, i.e., the generated token representing the model's confidence. $A_{i,j}$ is a binary variable equal to 1 if the first generated token is "A". As explained in Section 1, this token does not necessarily correspond to the most likely token based on the inner probability but it is the outcome of the decoding strategy. The index i denotes the individual observation, while j refers to the specific prompt. The key coefficient, β, captures whether the LLM systematically declares higher confidence when the first generated token is "A", given the same inner probabilities. As a robustness check, we perform sample splits based on realized returns and inner probabilities. In all regressions, we cluster standard errors at the prompt level.

Table 7 presents the results. For the same news item, the declared confidence is significantly higher, by 0.0337 on average, when the prediction token is "A" instead of "B." The economic magnitude of this bias is also significant. For reference, the standard deviation of declared confidence in the full sample is 0.106. Moreover, the same bias persists in cases where the inner probability is actually higher for B (column 3) or when the market reaction is negative (column 5). These results demonstrate a distinct bias within the declared confidence metric, introduced by the feedback loop. Unlike inner confidence, which is conditional only on the original prompt, declared confidence is also influenced by the generated response. In this case, the bias may stem from the LLM arbitrarily favoring the letter "A" over B or from a general tendency to express higher confidence when predicting positive outcomes.

Next, we examine the distribution of declared confidence, shown in Figure 7. Two main patterns emerge. First, the distribution is highly discrete, as the LLM overwhelmingly favors confidence values reported with a single decimal digit, even though the prompt does not specify any formatting requirement. Nearly all declared confidence estimates take this form, with only a few exceptions such as 0.75 and 0.85. Second, for no obvious reasons, the

[17] The final sample size is slightly lower than the theoretical maximum because we drop observations where the API failed to process the input or produced unparsable output. We apply this data cleaning procedure consistently throughout the paper. In this experiment, the temperature is set to 1 to induce variation while maintaining coherence.

Table 7: **Identifying the Effect of Conditioning**
This table presents the estimation results for Eq. (21). The dependent variable is the declared confidence, while the independent variable is an indicator variable that equals to 1 if the classification is A, 0 otherwise. The first column reports the estimated coefficient for the full sample. The next two columns present results for subsamples in which the LLM assigns a probability to A above or below 0.5, respectively. The final two columns show results for the subset of news articles corresponding to firms whose return on the day of the news was above or below zero, respectively. Standard errors are clustered at article level. For each estimation we report the point estimate of the parameters and its standard error (SE), and the sample size. *, **, and *** denote statistical significance at the 10%, 5%, and 1% level, respectively.

	(1)	(2)	(3)	(4)	(5)
	Full Sample	$\tilde{p}(A) > 0.5$	$\tilde{p}(A) \leq 0.5$	$r_{i,t} > 0$	$r_{i,t} \leq 0$
$A_{i,j}$	0.0337***	0.0432***	0.0216***	0.0367***	0.0307***
	(0.0038)	(0.0047)	(0.0048)	(0.0055)	(0.0052)
Article FE	✓	✓	✓	✓	✓
Observations	49,999	22,944	27,055	25,000	24,999
R^2	0.5684	0.5154	0.5993	0.5695	0.5673

model reports a confidence of exactly 0.8 in more than 35% of cases. These distributional features likely reflect regularities in the training data. LLMs are trained to predict the most likely next token from large text corpora, and tokens representing words and numbers are treated identically in this process. The training objective does not endow the model with a mathematical understanding of numbers. For example, in a simple random number generation task—where one might expect the model to approximate a uniform distribution—the realized output probabilities exhibit a clear bias. Rather than sampling uniformly, GPT-4o shows a pronounced tendency to favor the number 7 (see Figure 8). These findings caution against attributing inherent numerical meaning to LLM's declared confidence or interpreting them as statistically calibrated quantities.

6 Conclusion

This paper shows that the conditional probabilities produced by autoregressive LLMs contain economically meaningful information beyond the generated text. By leveraging these inner probabilities, we develop an entropy-based framework for uncertainty quantification and construct a simple inner-confidence measure that is interpretable, scalable, and directly linked to Shannon entropy in binary settings.

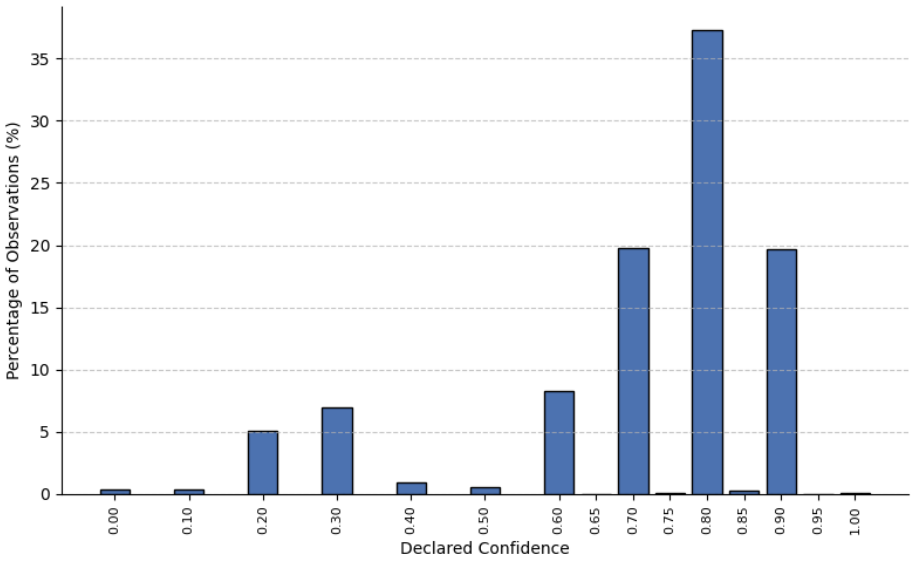

Figure 7: **Distribution of Declared Confidence.** The figure above shows the distribution of declared confidence values across observations, expressed as a percentage of the total sample.

Empirically, inner confidence is strongly related to objective predictive accuracy. In a large-scale news classification exercise, we find that LLM prediction accuracy rises monotonically with inner confidence. Conditioning on inner confidence materially improves portfolio performance: high-confidence signals generate substantially higher Sharpe ratios, while low-confidence signals produce no reliable excess returns. These results demonstrate that LLM uncertainty is economically informative, even if these beliefs are not perfectly calibrated. In contrast, black-box alternatives such as declared confidence fail to deliver comparable performance enhancement and is significantly biased by the decoding process.

We further show that inner confidence varies systematically with information complexity and prompt-induced shifts in priors and signal precision, and that unusually high confidence can help diagnose potential memorization and look-ahead biases. In the cross section, news assessed with low inner confidence is followed by higher subsequent return volatility and elevated abnormal trading volume, and we document rich heterogeneity across firms in how inner confidence varies by news subjects.

Overall, our paper reframes LLMs as probabilistic forecasting systems rather than text generators. Access to inner probabilities allows researchers to quantify and rank model reliability, opening a practical path toward more transparent and reliable use of generative AI in finance.

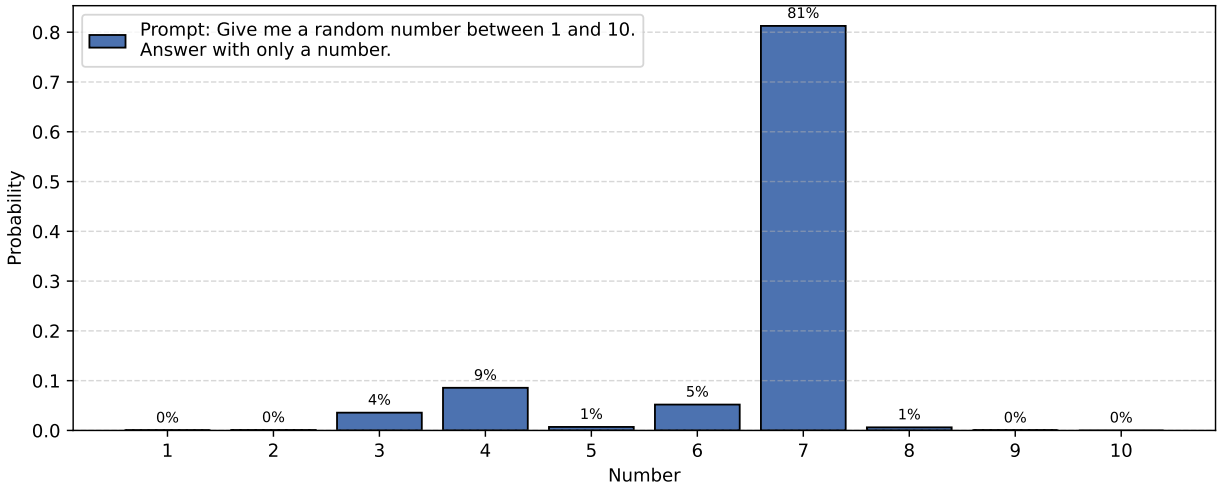

Figure 8: **Random Number Generation.** The figure shows the inner probabilities generated by the model when prompted to select a random number between 1 and 10.

References

Acikalin, Utku, Tolga Caskurlu, Gerard Hoberg, and Gordon M Phillips, 2026, Intellectual property protection lost and competition: An examination using large language models, *Journal of Financial Economics* Forthcoming.

Allena, Rohit, 2025, Confident risk premiums and investments using machine learning uncertainties, *The Review of Financial Studies, forthcoming* .

Amihud, Yakov, 2002, Illiquidity and stock returns: Cross-section and time-series effects, *Journal of Financial Markets* 5, 31–56.

Ashokkumar, Ashwini, Luke Hewitt, Isaias Ghezae, and Robb Willer, 2024, Predicting results of social science experiments using large language models, arXiv preprint arXiv:2504.01167.

Babina, Tania, Anastassia Fedyk, Alex He, and James Hodson, 2024, Artificial intelligence, firm growth, and product innovation, *Journal of Financial Economics* 151, 103745.

Barber, Brad M, and Terrance Odean, 2008, All that glitters: The effect of attention and news on the buying behavior of individual and institutional investors, *The review of financial studies* 21, 785–818.

Brown, Tom B, et al., 2020, Language models are few-shot learners, *Advances in Neural Information Processing Systems (NeurIPS)* 33, 1877–1901.

Bybee, Leland, 2023, Surveying generative ai's economic expectations, arXiv preprint arXiv:2305.02823.

Bybee, Leland, Bryan Kelly, Asaf Manela, and Dacheng Xiu, 2024, Business news and business cycles, *The Journal of Finance* 79, 3105–3147.

Bybee, Leland, Bryan T Kelly, Asaf Manela, and Dacheng Xiu, 2020, The structure of economic news, Available at SSRN 3522305.

Caragea, Doina, Mark Chen, Theodor Cojoianu, Mihai Dobri, Kyle Glandt, and George Mihaila, 2020, Identifying fintech innovations using bert, Proceedings of the 2020 IEEE International Conference on Big Data (Big Data).

Chang, Anne, Xi Dong, Xiumin Martin, and Changyun Zhou, 2023, AI democratization, return predictability, and trading inequality, Available at SSRN 4543999.

Chen, Yifei, Bryan T Kelly, and Dacheng Xiu, 2022, Expected returns and large language models, Available at SSRN 4416687.

Cheng, Qiang, Pengkai Lin, and Yue Zhao, 2025, Does generative ai facilitate investor trading? early evidence from chatgpt outages, *Journal of Accounting and Economics* 101821.

Desai, Nikita, and Greg Durrett, 2020, Calibration of pretrained transformers, in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 295–302.

Didisheim, Antoine, Martina Fraschini, and Luciano Somoza, 2025, AI's predictable memory and its implications for finance, *Economics Letters* 256.

Dubey, Abhimanyu, et al., 2024, The llama 3 herd of models, arXiv preprint arXiv:2407.21783.

Eisfeldt, Andrea L, and Gregor Schubert, 2024, AI and finance, Available at SSRN 4988553.

Engelberg, Joseph, Asaf Manela, William Mullins, and Luka Vulicevic, 2025, Entity neutering, Available at SSRN 5182756.

Fama, Eugene F, and Kenneth R French, 1993, Common risk factors in the returns on stocks and bonds, *Journal of Financial Economics* 33, 3–56.

Fama, Eugene F., and Kenneth R. French, 2015, A five-factor asset pricing model, *Journal of Financial Economics* 116, 1–22.

Fan, Angela, Mike Lewis, and Yann Dauphin, 2018, Hierarchical neural story generation, Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).

Fang, Lily, and Joel Peress, 2009, Media coverage and the cross-section of stock returns, *The Journal of Finance* 64, 2023–2052.

Fedyk, Anastassia, Ali Kakhbod, Peiyao Li, and Ulrike Malmendier, 2024, Chatgpt and perception biases in investments: An experimental study, Available at SSRN 4787249.

Glasserman, Paul, and Caden Lin, 2023, Assessing look-ahead bias in stock return predictions generated by gpt sentiment analysis, arXiv preprint arXiv:2309.17322.

Glasserman, Paul, and Harry Mamaysky, 2019, Does unusual news forecast market stress?, *Journal of Financial and Quantitative Analysis* 54, 1937–1974.

Glasserman, Paul, Harry Mamaysky, and Jimmy Qin, 2023, New news is bad news, arXiv preprint arXiv:2309.05560.

He, Songrun, Linying Lv, Asaf Manela, and Jimmy Wu, 2025, Chronologically consistent large language models, arXiv preprint arXiv:2502.21206.

Hoberg, Gerard, and Asaf Manela, 2025, The natural language of finance, *Foundations and Trends in Finance* 14, 244–365.

Holtzman, Ari, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi, 2019, The curious case of neural text degeneration, arXiv preprint arXiv:1904.09751.

Jiang, Heinrich, Been Kim, Melody Guan, and Maya Gupta, 2021, To trust or not to trust a classifier, in *Advances in Neural Information Processing Systems (NeurIPS)*.

Jobson, J Dave, and Bob M Korkie, 1981, Performance hypothesis testing with the sharpe and treynor measures, *The Journal of Finance* 889–908.

Korinek, Anton, 2023, Generative ai for economic research: Use cases and implications for economists, *Journal of Economic Literature* 61, 1281–1317.

Kuhn, Lorenz, Yarin Gal, and Sebastian Farquhar, 2023, Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, in *International Conference on Learning Representations (ICLR)*.

Levy, Bradford, 2024, Caution ahead: Numerical reasoning and look-ahead bias in ai models, Available at SSRN 5082861.

Li, Shuting, Zhen Shi, Yusen Xia, and Baozhong Yang, 2025, The value of stock analysis in news, Available at SSRN 5754262.

Lin, Stephanie, Jacob Hilton, and Owain Evans, 2022, Teaching models to express their uncertainty in words, *Transactions on Machine Learning Research* .

Lopez-Lira, Alejandro, and Yuehua Tang, 2023, Can chatgpt forecast stock price movements? return predictability and large language models, arXiv preprint arXiv:2304.07619.

Lopez-Lira, Alejandro, Yuehua Tang, and Mingyin Zhu, 2025, The memorization problem: Can we trust llms' economic forecasts?, arXiv preprint arXiv:2504.14765.

Ludwig, Jens, Sendhil Mullainathan, and Ashesh Rambachan, 2025, Large language models: An applied econometric framework, Technical report, National Bureau of Economic Research.

Memmel, Christoph, 2003, Performance hypothesis testing with the sharpe ratio, Available at SSRN 412588.

Radford, Alec, et al., 2019, Language models are unsupervised multitask learners, OpenAI blog post.

Sarkar, Suproteem K., and Keyon Vafa, 2024, Lookahead Bias in Pretrained Language Models, Available at SSRN 4754678.

Saunders, Anthony, Sascha Steffen, and Paulina M Verhoff, 2025, CovenantAI-new insights into covenant violations, Working paper.

Shefrin, Hersh, and Meir Statman, 1985, The disposition to sell winners too early and ride losers too long: Theory and evidence, *The Journal of Finance* 40, 777–790.

Sutskever, Ilya, Oriol Vinyals, and Quoc V Le, 2014, Sequence to sequence learning with neural networks, *Advances in Neural Information Processing Systems* 27, 3104–3112.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, 2017, Attention is all you need, *Advances in neural information processing systems* 30.

Wei, Jason, and al., 2022, Chain-of-thought prompting elicits reasoning in large language models, in *Advances in Neural Information Processing Systems*, volume 35, 24824–24837.

Yuan, Jiayi, et al., 2025, Understanding and mitigating numerical sources of nondeterminism in LLM inference, in *Advances in Neural Information Processing Systems*.

Appendix

A Proof of Lemma 1

Proof. Both $H_{\text{bin}}(s \mid P)$ and $C(s \mid P)$ can be expressed as a function of $p \equiv p(s \mid P)$: $H_{\text{bin}}(s \mid P) = H(p)$ and $C(s \mid P) = C(p)$. It is easy to see that both $H(p)$ and $C(p)$ are symmetric about $p = 0.5$, i.e. $H(p) = H(1-p)$ and $C(p) = C(1-p)$, so it suffices to study $p \in [1/2, 1]$. In this region, Eq. (6) simplifies to $C(p) = p - 0.5$.

From Eq. (5),

$$H'(p) = \log \frac{1-p}{p}.$$

For $p \in (1/2, 1)$, we have $H'(p) < 0$, i.e., H is strictly decreasing on $(1/2, 1)$, while the opposite is true for $C(p)$. This already implies that ranking by $H(p)$ is equivalent to ranking by $C(p)$ for $p \in [1/2, 1]$.

Next, the function g can be constructed through the inverse function of H subject to a change of variable. Given $C = p - 0.5$, define $\Phi(C)$ for $C \in [0, 1/2]$ as:

$$\Phi(C) \equiv H(C + 0.5).$$

Differentiating Φ gives

$$\Phi'(C) = \log \frac{\frac{1}{2} - C}{\frac{1}{2} + C},$$

and $\Phi'(C) < 0$ for $C \in (0, 1/2)$. Therefore, Φ is continuous on $[0, 1/2]$ and strictly decreasing on $(0, 1/2)$, and thus a bijection between $[0, 1/2]$ and $[H(1), H(1/2)] = [0, \log 2]$. The inverse function theorem (applied to a continuous strictly monotone map on a compact interval) guarantees that Φ admits a continuous inverse $\Phi^{-1} : [0, \log 2] \to [0, 1/2]$. Setting $g = \Phi^{-1}$ yields $C = g(H(p))$ for all $p \in [1/2, 1]$. By symmetry the same results hold for $p \in [0, 1/2]$. □

B Multi-token Uncertainty

To illustrate the multi-token approach discussed in Section 2.3, we use Prompt 4 to query GPT-4o 1,000 times with a temperature of 2, thereby generating 1,000 multi-token sequences $S^{(1)}, S^{(2)}, \cdots, S^{(1000)}$. Setting a high temperature ensures substantial diversity across the generated sequences $S^{(j)}$. At the same time, the OpenAI API provides the default *inner*

probabilities prior to the temperature transformation.

```
In five words or fewer, what is the most likely cause of a U.S. recession
 in 2030?
```

Prompt 4: **Multi-token illustration:**
Simple prompt designed to generate short answer on economic topics to illustrate multi-token uncertainty measure.

We use these probabilities to compute the conditional probability of each sequence, $p(S \mid P)$, via the chain rule (Equation (10)). We then focus on the 20 most likely sequences according to this conditional probability and compute the normalized conditional probability $p_M(S^{(j)} \mid P)$, as defined in Equation (12).

Table A1 reports the results. While this sequence-level approach offers flexibility and generality, it introduces semantic ambiguity: identical meanings expressed using different phrasings are treated as distinct sequences. In most applications, addressing this issue would require the construction of semantic clusters (see, e.g. Kuhn et al., 2023), which in turn introduces additional modeling opacity and computational overhead.

Consequently, as argued in Section 2.3, in applications where predefined categories or semantic clarity are paramount, token-level responses are often preferable. Moreover, careful prompt design can bridge the gap between open-ended generation and structured classification. For example, rather than eliciting free-form responses, one may prompt the model to enumerate candidate answers in a structured list, each associated with a token-level identifier such as a letter. The model can then be instructed to select the most plausible option via a single-token response (e.g., "—b"), thereby preserving interpretability while retaining the generative richness of the initial prompt. This hybrid strategy combines the semantic granularity of generation with the analytical clarity of classification.

C Data Cleaning

Our news data are sourced from the Thomson Reuters Real-Time News Feed, following the data construction procedure described in Chen et al. (2022). We apply the following filters sequentially.

1. **Single-stock association.** We retain only news articles tagged with a single stock, ensuring that each article pertains directly to one firm rather than mentioning it in-

Table A1: **Multi-Token Illustration**
This table displays the top 20 completions sampled from GPT-4o in response to the prompt *"In five words or fewer, what is the most likely cause of a U.S. recession in 2030?"*, ranked by normalized likelihood ($p_M(S^{(j)} \mid P)$).

message	$p_M(S^{(j)} \mid P)$
Economic policy missteps	0.2811
Technological disruption and job displacement	0.2213
Economic policy mismanagement	0.0631
Technological disruption or geopolitical tensions	0.0590
Technological disruption and job loss	0.0506
Debt crisis or technological disruption	0.0460
Monetary policy or geopolitical tensions	0.0447
Global economic slowdown	0.0327
Rising interest rates and inflation	0.0306
Tight monetary policy	0.0290
Global economic downturn	0.0272
Interest rate hikes	0.0178
Monetary policy or economic imbalance	0.0177
Global economic instability	0.0167
Rising debt and inflation pressures	0.0140
Debt crisis or geopolitical tensions	0.0132
High interest rates or inflation	0.0102
Technological disruption or economic imbalance	0.0101
Technological disruption or policy missteps	0.0087
Monetary tightening or geopolitical tensions	0.0064

cidentally. Articles referencing multiple firms are excluded to avoid ambiguity in the stock-return mapping.

2. **Return availability.** We match each article to the corresponding firm's equity return data from CRSP. Articles for which no valid return observation is available on the relevant date are dropped.
3. **Article length.** We remove excessively short articles (fewer than 100 characters) that are unlikely to contain meaningful information. While Chen et al. (2022) exclude extremely long reports (exceeding 100,000 characters) that typically correspond to regulatory filings or comprehensive annual reviews, we impose a stricter upper limit of 5,000 characters to keep experimentation costs down.
4. **Redundancy filtering.** To ensure content novelty, we compute pairwise cosine similarity scores based on bag-of-words representations between each article and all articles published in the preceding five business days. An article is classified as redundant and removed if its cosine similarity with any prior article exceeds 0.8. This step substan-

tially reduces repetitive content and improves the signal-to-noise ratio in our sample.

In addition to these standard filters from Chen et al. (2022), we impose experiment-specific criteria described in the main text. For the nowcasting exercise, we restrict to articles published between 10 a.m. and 2 p.m. ET, with a length between 100 and 5,000 characters. For the portfolio exercise, we focus on overnight news with timestamps after 4 p.m. on day $t-1$ and before 9 a.m. on day t. Both exercises only use articles published after GPT-4o's training cutoff in October 2023.

D Technical Considerations

This Appendix outlines the practical implementation of our methodology, detailing the API configurations required to extract token-level log-probabilities directly from the model along with technical considerations.

Extracting Inner Probabilities A brute-force estimation of inner probabilities–sampling numerous generated texts from the same prompt–would be computationally expensive and yield only an approximate estimate. Fortunately, such an approach is unnecessary. Most LLM providers, including all major open-source implementations, offer functionality to return both the generated text *and* the associated token-level probabilities. In the case of OpenAI's GPT models, these probabilities can be retrieved with minimal modification to the API call–typically by adding two lines of code to the prompt processing request (see Figure A1).

Without Logprobs

```
client.chat.completions.create(
    messages=[{
        "role": "user",
        "content": prompt,
    }],
    model="chatgpt-4o-latest"
)
```

With Logprobs

```
client.chat.completions.create(
    messages=[{
        "role": "user",
        "content": prompt,
    }],
    model="chatgpt-4o-latest",
    logprobs=True,
    top_logprobs=10
)
```

Figure A1: The figure shows the necessary code for extracting inner probabilities via the OpenAI API. The 'logprobs=True" parameter enables access to token-level probabilities for the next predicted tokens, returned as log-probabilities. This is essential for obtaining $\tilde{p}(s_1 \mid P)$ from model outputs.

Parsing and Failed Prompts Throughout the paper, we present various tests using different prompts. To facilitate token parsing, we sometimes request that results be generated within special symbols, such as $[\cdot]$ or $< \cdot >$. This structure helps automate the processing of large volumes of generated text.

However, some prompts occasionally fail to produce parsable text. This can occur due to a rare token draw or a poor conditional distribution of the model. When this happens, we simply discard the observation. As a result, the total number of observations may not always sum precisely to the declared sample size. These discrepancies are minimal and unlikely to affect the significance of our results.

Reproducibility and Hardware Non-determinism For all our tests, we use the API provided by OpenAI. Our default model is *gpt-4o*, which we compare to GPT-3.5 using *gpt-3.5-turbo-0125*. As mentioned above, we obtain the inner probabilities, $\tilde{p}(s_1 \mid P)$, directly from the API. However, obtaining the exact same probability values across repeated runs is often impossible, even with identical settings (e.g., temperature set to 0). This is caused by the non-associative nature of floating-point arithmetic on GPUs (Yuan et al., 2025). Large-scale inference relies on massive parallelism, where the order of operations in summation (such as parallel reductions) varies based on microscopic timing differences between GPU threads. Consequently, $(a+b)+c$ may not equal $a+(b+c)$ at the level of machine precision. This hardware-level noise introduces slight variations in the model's output logits. However, this variation does not affect any of the experiments or arguments in this paper; it merely adds noise, which we mitigate by conducting sufficiently large tests.

Quantization LLMs require immense computational resources, particularly for inference, due to the vast number of parameters and operations involved. To reduce computational cost and memory usage, many providers implement *quantization*–a technique that reduces the precision of model weights and activations.

In essence, quantization approximates high-precision floating-point numbers (typically 32-bit floats) using lower-precision representations, such as 16-bit or even 8-bit integers. For example, a model weight originally represented as a float32 value of 0.1234567 may be rounded to an int8-equivalent level such as 0.12 or 0.125. This process significantly decreases storage requirements and accelerates inference by enabling the use of faster, low-precision arithmetic.

However, quantization introduces rounding error and numerical imprecision. While these effects may be negligible for some applications, they can materially affect analyses–such as

ours–that rely on subtle differences in token-level probabilities. As we show later in this paper, small variations in the model's inner probabilities can translate into economically meaningful differences. If such distinctions are flattened or distorted by quantization, the resulting analyses may be biased or less informative. Therefore, whenever precision matters (as is the case when evaluating $\tilde{p}(s_1 = A \mid P)$) quantization must be treated not merely as a technical footnote but as a first-order concern.

E Portfolio Construction Methodology

In this Appendix we provide a comprehensive formalization of the long-short strategies and the signal aggregation techniques used in the trading experiments in Section 3.

Benchmark The benchmark portfolio is an equally weighted strategy that buys companies with good news based on the LLM's predictions and shorts those with bad news. On any given day, a company may have more than one news item; we aggregate the signals at the firm level. Specifically, we define the signal as positive if a company receives more positive signals than negative ones.

Formally, let each news item for firm i on day t be represented by a discrete generated token $S_{i,k,t} \in \{A, B\}$, where A represents good news and B represents bad news.

We define the daily signal for firm i on day t by a single letter $Z_{i,t}$ taking values 1 for a "good" signal and -1 for a "bad" signal. The signal is good if the number of good new for firm i on day t is larger than the number of bad news for the same firm on the same day:

$$Z_{i,t} = \begin{cases} 1, & \text{if } \sum_k \mathbf{1}\{S_{i,k,t} = A\} > \sum_k \mathbf{1}\{S_{i,k,t} = B\}, \\ -1, & \text{otherwise.} \end{cases} \tag{22}$$

We construct our benchmark portfolio by placing \$1 in the long ("good") leg and \$1 in the short ("bad") leg. Let $N_t^+ = \sum_{i=1}^N \mathbf{1}\{Z_{i,t} = 1\}$ and $N_t^- = \sum_{i=1}^N \mathbf{1}\{Z_{i,t} = -1\}$ denote the number of long and short positions on day t, respectively. If $r_{i,t+1}$ is the open-close return of firm i at $t+1$, the benchmark portfolio return on day t is:

$$r_t^{\text{bench}} = \frac{1}{N_t^+} \sum_{\{i:Z_{i,t}=1\}} r_{i,t+1} - \frac{1}{N_t^-} \sum_{\{i:Z_{i,t}=-1\}} r_{i,t+1}. \tag{23}$$

Confidence portfolios We now exploit the full distributional information from $\tilde{p}\big(S_{i,k,t} = A \mid P_{i,k,t}\big)$ rather than relying on the discrete token $S_{i,k,t}$. Define the confidence for each news

item k of firm i on day t as

$$C_{i,k,t} \;=\; \Big|\tilde{p}\big(S_{i,k,t} = A \mid P_{i,k,t}\big) \;-\; 1/2\Big|. \tag{24}$$

We select only those items whose confidence exceeds a threshold $c(\nu_t)$ where ν_t is a hyperparameter defined later this section:

$$\overline{\mathcal{N}}_{i,t} \;=\; \Big\{k : \Big|\tilde{p}\big(S_{i,k,t} = A \mid P_{i,k,t}\big) - 1/2\Big| > c(\nu_t)\Big\}. \tag{25}$$

For those selected items, we average the probabilities:

$$\overline{p}_{i,t} \;=\; \frac{1}{\big|\overline{\mathcal{N}}_{i,t}\big|} \sum_{k \in \overline{\mathcal{N}}_{i,t}} \tilde{p}\big(S_{i,k,t} = A \mid P_{i,k,t}\big). \tag{26}$$

We then define the high-confidence signal $\overline{Z}_{i,t}$:

$$\overline{Z}_{i,t} \;=\; \begin{cases} +1, & \text{if } \overline{p}_{i,t} > 0.5, \\ -1, & \text{otherwise.} \end{cases} \tag{27}$$

As with the benchmark, we form two equally weighted long and short legs to build a portfolio, based on $\overline{Z}_{i,t}$. Define

$$N_t^{+,h} \;=\; \sum_{i=1}^{N} \mathbf{1}\{\overline{Z}_{i,t} = +1\}, \quad N_t^{-,h} \;=\; \sum_{i=1}^{N} \mathbf{1}\{\overline{Z}_{i,t} = -1\}. \tag{28}$$

The daily return of this high-confidence portfolio is

$$r_t^{\text{high-conf}} \;=\; \frac{1}{N_t^{+,h}} \sum_{\{i:\overline{Z}_{i,t}=+1\}} r_{i,t+1} \;-\; \frac{1}{N_t^{-,h}} \sum_{\{i:\overline{Z}_{i,t}=-1\}} r_{i,t+1}. \tag{29}$$

We also construct a low-confidence portfolio by selecting the news items whose confidence is at or below the threshold $c(\nu_t)$:

$$\underline{\mathcal{N}}_{i,t} \;=\; \Big\{k : \Big|\tilde{p}\big(S_{i,k,t} = A \mid P_{i,k,t}\big) - 0.5\Big| \le c(\nu_t)\Big\}. \tag{30}$$

Define

$$\underline{p}_{i,t} \;=\; \frac{1}{\big|\underline{\mathcal{N}}_{i,t}\big|} \sum_{k \in \underline{\mathcal{N}}_{i,t}} \tilde{p}\big(S_{i,k,t} = A \mid P_{i,k,t}\big). \tag{31}$$

Then the low-confidence signal $\underline{Z}_{i,t}$ is

$$\underline{Z}_{i,t} \;=\; \begin{cases} +1, & \underline{p}_{i,t} > 0.5, \\ -1, & \underline{p}_{i,t} \le 0.5. \end{cases} \tag{32}$$

We form the low-confidence portfolio analogously:

$$N_t^{+,\ell} \;=\; \sum_{i=1}^{N} \mathbf{1}\{\underline{Z}_{i,t} = +1\}, \quad N_t^{-,\ell} \;=\; \sum_{i=1}^{N} \mathbf{1}\{\underline{Z}_{i,t} = -1\}, \tag{33}$$

$$r_t^{\text{low-conf}} \;=\; \frac{1}{N_t^{+,\ell}} \sum_{\{i:\underline{Z}_{i,t}=+1\}} r_{i,t+1} \;-\; \frac{1}{N_t^{-,\ell}} \sum_{\{i:\underline{Z}_{i,t}=-1\}} r_{i,t+1}. \tag{34}$$

Threshold Selection for the High-Confidence Portfolio We determine the threshold parameter ν_t for $c(\nu_t)$ using a 3 months rolling-window. This approach ensures that ν_t is chosen exclusively using historical data (up to each recalibration date), thus avoiding look-ahead bias and preserving out-of-sample integrity. Concretely, we implement the following steps:

1. **Define Grid:** For each recalibration date, we consider a grid of values for ν_t ranging from 0 to 0.5 in increments of 0.01.
2. **Quantile Threshold:** Given a chosen ν_t, the quantity $c(\nu_t)$ is defined as the ν_t-quantile of the confidence on the full expanding window. For instance, if $\nu_t = 0.25$, then $c(\nu_t)$ is the first quartile in our *training* sample.
3. **Compute High-Conf Returns:** For each ν_t on the grid, we construct the High-Confidence portfolio (using data up to day t) and compute its Sharpe ratio over the expanding window.
4. **Select ν_t:** We choose the ν_t that maximizes the Sharpe ratio of the High-Confidence portfolio over the expanding window. This selection is based solely on past data, preventing any form of "cheating" for out-of-sample analysis.
5. **Monthly Recalibration:** We repeat this procedure at the start of each month. The selected ν_t is then held fixed for that month and recalibrated again at the next month's start using the updated expanding window.

F Robustness Checks for Portfolio Results

F.1 Risk-adjusted performances

Table A2 reports the risk-adjusted performance of the three portfolios (Baseline, high-declared-confidence, and high-inner-confidence) using three different factor models: CAPM, FF5 augmented with momentum (Mom), and FF5 augmented with momentum and short-term reversal (ST-Rev). The results show that all three portfolios generate significant positive alphas. The alphas of the baseline portfolio and the high-declared-confidence are comparable in magnitude across all specifications, while those for the high-inner-confidence portfolio are significantly higher. The portfolios load negatively on SMB and St-Rev, while positively on HML and Mom. These factor loadings suggest that the LLM-prediction-based portfolios tend to short small-caps, past losers (both one-month and one-year), and growth stocks.

F.2 In-sample performance of confidence-based portfolios

We replicate our portfolio exercise from Table 3 on a pre-training cutoff sample period spanning October 2022 to September 2023. These results serve as an in-sample robustness check, confirming that the relative performance ordering remains stable even when the model may have been exposed to the underlying data during training.

F.3 Portfolios with minimum size threshold

To ensure the robustness of our findings to market-cap-driven frictions, we replicate the main exercise from Section 3.2 under varying size filters and present the results in Table A4. We report Sharpe ratios and asset counts for inner-confidence and declared-confidence strategies across four exclusion levels, ranging from an unrestricted sample to a $1 billion minimum market capitalization requirement.

F.4 Fixed-threshold portfolios

This section provides a robustness check for the exercise in Table 3 by presenting the performance of portfolios constructed using static confidence quantile thresholds. We report the annualized returns and volatility for both inner-confidence (Table A5) and declared-confidence (Table A6) strategies at fixed 10%, 20%, and 30% split levels to confirm that our findings are not dependent on the dynamic recalibration.

Table A2: **Risk-Adjusted Performance of Confidence-Based Portfolios**
This table reports factor model regressions of daily long–short portfolio returns over the out-of-sample period (October 2023 to November 2024). Columns (1)–(3) correspond to: CAPM, FF5 augmented with the momentum factor (Mom), and FF5 augmented with momentum and short-term reversal (ST-Rev). The alpha ($\hat{\alpha}$) is annualized (daily $\times$ 252) and reported in percent. Factor data are from Kenneth French's data library. t-statistics based on heteroskedasticity-robust (HC1) standard errors are reported in parentheses. $^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$.

	baseline			**high-declared-confidence**			**high-inner-confidence**		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
$\hat{\alpha}$	41.72^{***}	37.17^{***}	38.98^{***}	43.40^{***}	37.58^{***}	39.08^{***}	51.59^{***}	46.78^{***}	48.66^{***}
	(4.75)	(4.45)	(4.64)	(4.19)	(3.87)	(3.94)	(5.72)	(5.49)	(5.64)
Mkt−RF	-0.081^{*}	−0.045	−0.042	-0.096^{**}	−0.070	−0.068	-0.074^{*}	−0.039	−0.036
	(−1.86)	(−0.87)	(−0.79)	(−2.01)	(−1.20)	(−1.15)	(−1.69)	(−0.78)	(−0.70)
SMB		-0.141^{**}	-0.124^{**}		-0.185^{***}	-0.170^{**}		-0.141^{**}	-0.123^{*}
		(−2.33)	(−1.99)		(−2.62)	(−2.35)		(−2.32)	(−1.95)
HML		0.141^{***}	0.118^{**}		0.201^{***}	0.182^{**}		0.151^{***}	0.127^{**}
		(2.70)	(2.23)		(2.86)	(2.52)		(2.84)	(2.37)
RMW		0.069	0.087		0.002	0.017		0.076	0.095
		(0.85)	(1.09)		(0.02)	(0.18)		(0.88)	(1.11)
CMA		−0.015	0.031		−0.025	0.013		−0.034	0.014
		(−0.16)	(0.32)		(−0.24)	(0.12)		(−0.39)	(0.15)
Mom		0.172^{***}	0.155^{***}		0.234^{***}	0.220^{***}		0.184^{***}	0.166^{***}
		(2.93)	(2.70)		(3.28)	(3.13)		(3.34)	(3.09)
ST-Rev			-0.118^{**}			-0.098^{*}			-0.122^{**}
			(−2.35)			(−1.72)			(−2.35)
R^2	0.013	0.128	0.145	0.013	0.136	0.144	0.010	0.131	0.148
N	284	284	284	284	284	284	284	284	284

Table A3: **In-Sample Performance of Confidence-Based Portfolios**
This table reports performance of three long-short investment strategies based on the model confidence in the period between October 2022 and September 2023 (before GPT-4o training cutoff). Inner Confidence refers to the classification obtained following the method presented in Section 2, while Declared Confidence refers to the self reported confidence obtained with Prompt 2. The *Baseline* portfolio includes all news, the *High-Confidence* portfolio includes only news for which the model generated a confident classification, and the *Low-Confidence* portfolio includes non-confident classifications only. The rows indicate the Annualized return (in percent), volatility (in percent), Sharpe ratio, and average number of stocks. All portfolios are equally weighted and rebalanced daily.

	Inner Confidence		**Declared Confidence**		**Baseline**
	High Conf	Low Conf	High Conf	Low Conf	
Mean (%)	72.42	7.80	68.96	80.51	59.51
Std (%)	11.16	22.70	12.99	36.14	10.41
Sharpe ratio	6.49	0.34	5.31	2.23	5.71
Avg # stocks	452.58	129.95	404.12	158.28	505.98

Table A4: **Confidence-Based Portfolios with Different Size Thresholds**
This table reports annualized Sharpe ratios and the average number of assets for confidence-based long-short portfolios under different market capitalization filters. For each filter, we report results for the High-Confidence and Low-Confidence portfolios constructed using inner confidence and declared confidence, alongside the Baseline portfolio that includes all news. The confidence threshold is selected using the rolling-window procedure described in Appendix E. The size filters are: no filter, market capitalization above $100 million, and market capitalization above the NYSE 20th percentile. The sample period is from October 2023 to December 2024. All portfolios are equally weighted and rebalanced daily.

	Inner Confidence		**Declared Confidence**		**Baseline**
	High Conf	Low Conf	High Conf	Low Conf	
	A. Full Sample				
Sharpe ratio	5.15	−0.39	3.71	−0.11	4.24
Avg # assets	481.50	111.82	452.42	130.82	525.71
	B. Market Cap > $100M				
Sharpe ratio	4.06	0.20	3.01	−0.15	3.36
Avg # assets	444.66	96.65	416.53	112.80	483.15
	C. Market Cap > NYSE 20th Percentile				
Sharpe ratio	2.42	−1.48	2.26	0.51	2.04
Avg # assets	370.22	68.22	379.86	47.05	395.84

Table A5: **Performance of Inner Confidence-Based Portfolios with Fixed Threshold**

This table reports out-of-sample performance of two long-short investment strategies based on the model's *inner* confidence: the *High-Confidence* portfolio including only news for which the model generated a confident classification, and the *Low-Confidence* portfolio including non-confident classifications only. The first column indicates the percentage of the total sample that is allocated to the *Low-confidence* portfolio. We report the annualized return, volatility, and Sharpe ratio of each portfolio. The sample period is from October 2023 to December 2024. All portfolios are equally weighted and rebalanced daily.

Quantile	Confidence	Position	Mean Return	Volatility	Sharpe Ratio	# of Assets
10%	High	Long	0.0937	0.1416	0.6613	256.58
		Short	-0.3709	0.1649	-2.2491	252.96
		Long–Short	0.4645	0.0954	4.8706	254.77
	Low	Long	-0.4427	0.2807	-1.5771	34.41
		Short	-0.2603	0.2433	-1.0701	39.77
		Long–Short	-0.1885	0.3294	-0.5721	36.82
20%	High	Long	0.0884	0.1418	0.6232	248.91
		Short	-0.4075	0.1678	-2.4283	230.69
		Long–Short	0.4959	0.0987	5.0254	239.80
	Low	Long	-0.3501	0.2279	-1.5364	50.96
		Short	-0.2088	0.1893	-1.1032	71.41
		Long–Short	-0.1308	0.2180	-0.6001	61.10
30%	High	Long	0.0961	0.1406	0.6835	233.19
		Short	-0.3987	0.1706	-2.3379	220.62
		Long–Short	0.4948	0.1043	4.7432	226.90
	Low	Long	-0.1429	0.2162	-0.6611	73.41
		Short	-0.3203	0.1786	-1.7936	83.60
		Long–Short	0.1867	0.1951	0.9569	78.25

Table A6: **Performance of Declared Confidence-Based Portfolios with Fixed Threshold**

This table reports out-of-sample performance of two long-short investment strategies based on the model's *declared* confidence in its classification: the *High-Confidence* portfolio including only news for which the model reported a high confidence score, and the *Low-Confidence* portfolio including low-confidence classifications only. The first column indicates the percentage of the total sample that is allocated to the *Low-confidence* portfolio. Because declared confidence is heavily clustered at round values (e.g., 0.70, 0.80, 0.90), ties are broken by adding uniform random noise of order 10^{-6} to ensure exact quantile splits. We report the annualized mean return, volatility, and Sharpe ratio of each portfolio. The sample period is from October 2023 to December 2024. All portfolios are equally weighted and rebalanced daily.

Quantile	Confidence	Position	Mean Return	Volatility	Sharpe Ratio	# of Assets
10%	High	Long	0.0335	0.1407	0.2381	264.50
		Short	-0.3873	0.1665	-2.3262	233.26
		Long–Short	0.4208	0.0985	4.2727	248.88
	Low	Long	-0.0572	0.2038	-0.2809	34.72
		Short	-0.2379	0.2191	-1.0859	53.88
		Long–Short	0.2249	0.2085	1.0788	44.30
20%	High	Long	0.0252	0.1419	0.1777	249.03
		Short	-0.3620	0.1679	-2.1567	221.93
		Long–Short	0.3873	0.1024	3.7812	235.48
	Low	Long	-0.1721	0.2087	-0.8248	65.77
		Short	-0.3230	0.1876	-1.7219	71.32
		Long–Short	0.1467	0.1886	0.7777	68.55
30%	High	Long	0.0424	0.1427	0.2967	222.21
		Short	-0.3897	0.1715	-2.2724	203.14
		Long–Short	0.4320	0.1111	3.8881	212.67
	Low	Long	-0.1834	0.1923	-0.9538	101.64
		Short	-0.3322	0.1692	-1.9629	94.42
		Long–Short	0.1659	0.1631	1.0170	98.03

G Confidence Conditioning Prompts

This appendix details the full text of the experimental prompts used to assess the model's rational belief updates in Section 3.4. These include the specific semantic variants designed to perturb the model's prior uncertainty and signal precision, as well as the affectively charged "noise" prompts used for our placebo tests.

```
Yesterday, most analysts agreed on the prospects of the firm {target}.
Today, new news has emerged. Based on this news, please assess the
expected return for the stock. Answer A if the news is good, B if the
news is bad. Is this financial news good or bad for the stock price of {
target} in the short term? Please start by writing only the letter A or B
. Add no other formatting. ARTICLE: {headline} {body}
```

Prompt 5: **Prior Consensus**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm while introducing an initial conditioning on the model's prior. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

```
Yesterday, most analysts disagreed on the prospects of the firm {target}.
 Today, new news has emerged. Based on this news, please assess the
expected return for the stock. Answer A if the news is good, B if the
news is bad. Is this financial news good or bad for the stock price of {
target} in the short term? Please start by writing only the letter A or B
. Add no other formatting. ARTICLE: {headline} {body}
```

Prompt 6: **Prior Disagreement**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm with an initial conditioning on its prior. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

```
Most analysts agree about the consequence of the news below for stock
with ticker {target}. Based on this news, please say if you think the
return for the stock with ticker {target} will be positive or negative in
 the short term. Answer A if good news, B if bad news. Is this financial
news good or bad for the stock price of {target} in the short term?
Please start by writing only a letter A or B. Add no other formatting.
ARTICLE: {headline} {body}
```

Prompt 7: **Analyst Consensus**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm with an initial conditioning on the market interpretation. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

```
Most analysts disagree about the consequence of the news below for stock
with ticker {target}. Based on this news, please say if you think the
return for the stock with ticker {target} will be positive or negative in
 the short term. Answer A if good news, B if bad news. Is this financial
news good or bad for the stock price of {target} in the short term?
Please start by writing only a letter A or B. Add no other formatting.
ARTICLE: {headline} {body}
```

Prompt 8: **Analyst Disagreement**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm with an initial conditioning on the market interpretation. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

```
Confidence, clarity, conviction, precision, certainty. Based on the news
below, please say if you think the return for the stock with ticker {
target} will be positive or negative in the short term. Answer A if good
news, B if bad news. Is this financial news good or bad for the stock
price of {target} in the short term? Please start by writing only a
letter A or B. Add no other formatting. ARTICLE: {headline} {body}
```

Prompt 9: **Positive Noise**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm with an an initial conditioning with positive noise. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

```
Doubt, anxiety, mistrust, uncertainty, hesitation. Based on the news
below, please say if you think the return for the stock with ticker {
target} will be positive or negative in the short term. Answer A if good
news, B if bad news. Is this financial news good or bad for the stock
price of {target} in the short term? Please start by writing only a
letter A or B. Add no other formatting. ARTICLE: {headline} {body}
```

Prompt 10: **Negative Noise**
The prompt asks the LLM to predict the market reaction (positive or negative) to a news item related to a specific listed firm with an an initial conditioning with negative noise. The brackets $\{\cdots\}$ indicate dynamic inserts where we place the *target* (firm's ticker), *headline* (article's headline), and *body* (main text of the news article).

H Robustness Check for Look-ahead Bias

In this Appendix, we provide further sub-sample analyses to isolate potential look-ahead contamination among firms most likely to be represented in the LLM's training corpus. We replicate the daily average confidence analysis presented in Figure 5 for firms in the top 10% of news coverage (Figure A2 Top) and market capitalization (Figure A2 Bottom), demonstrating that the lack of a significant confidence shift around the training cutoff is consistent across these highly visible subsets.

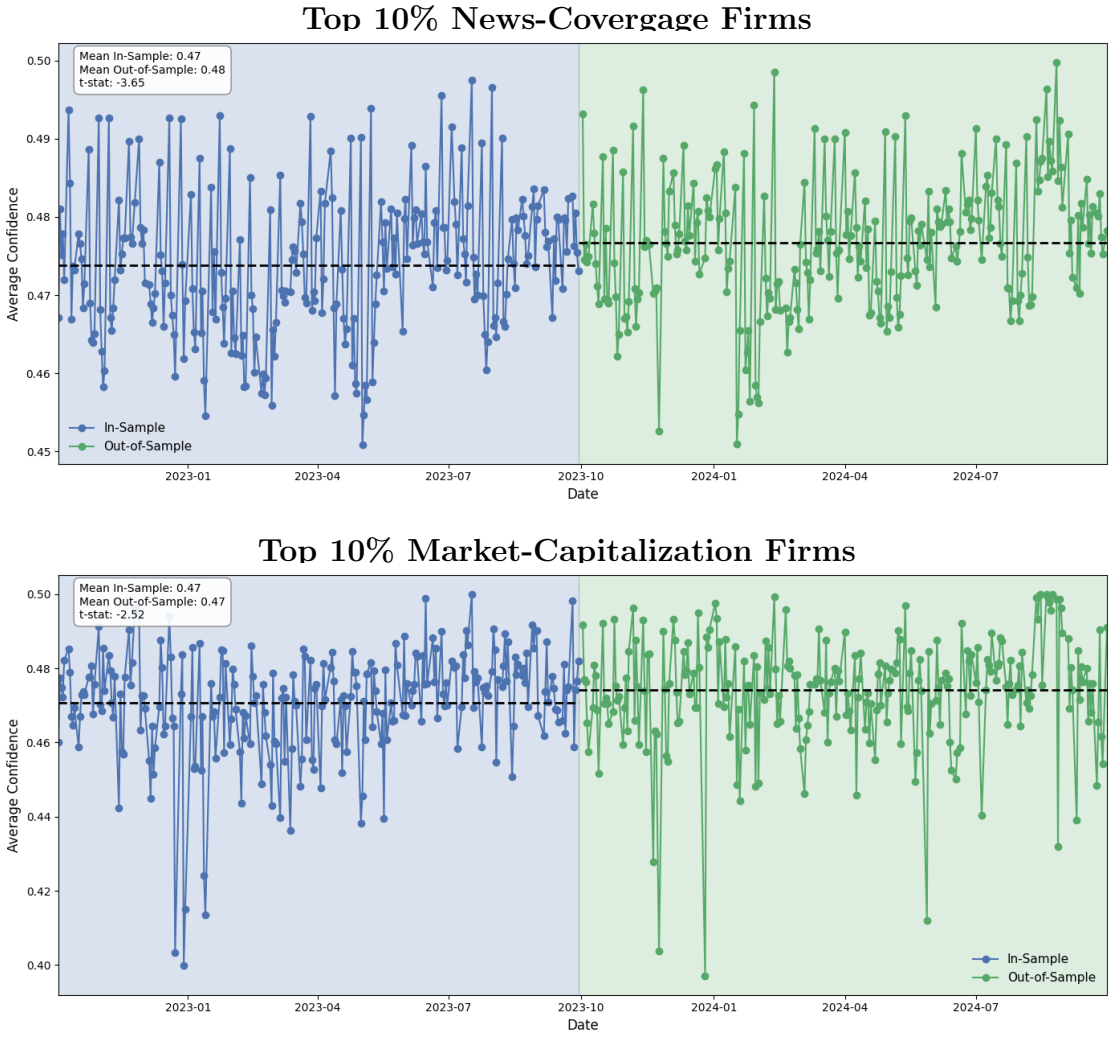

Figure A2: **Daily Average Inner Confidence — Robustness Checks.** This figure replicates Figure 5 for two sub-samples: firms in the top 10% of (a) news coverage and (b) market capitalization, computed annually so that thresholds are defined within each year. The patterns are consistent with the baseline, showing similar shifts in inner confidence around the GPT-4o training sample cutoff in October 2023.