# Mitigating Misleading and Ensuring Consistency in LLM Decision-Making Under Uncertainty: Evidence from the Financial Industry

## Abstract

Large Language Models (LLMs) are increasingly employed in high‑stakes decision‑making contexts such as financial advisory and risk management. However, these models may produce misleading or deceptive outputs and exhibit inconsistency when handling uncertain scenarios. This paper investigates the prevalence of misleading recommendations and the consistency of LLM-driven advice under varying roles and client profiles in a simulated financial environment. Using a systematically generated dataset of investment profiles (via gen\_dataset.py fileciteturn0file2), we executed a two‑stage evaluation pipeline (main.py fileciteturn0file1) to collect model reasoning and final recommendations from GPT‑4o‑Mini under different system prompts representing competing incentives. Statistical analyses and visualizations were conducted with the analyze\_scores.py suite fileciteturn0file0. Our results reveal statistically significant divergences in recommended risk across roles and a non‑trivial rate of deceptive consistency failures. We discuss implications for deploying LLMs in regulated industries and propose avenues for enhancing model reliability and transparency.

## 1. Background

Recent work has highlighted how LLMs can inadvertently generate deceptive or biased outputs, especially when role‑prompted to optimize conflicting objectives (Li & Chen, 2024; Wang & Wang, 2025). Studies such as *Confidence Improves Self‑Consistency in LLMs* (arXiv:2502.06233) and *Bias Runs Deep: Implicit Reasoning Biases in Persona‑Assigned LLMs* (arXiv:2311.04892v2) demonstrate that explicit persona assignments can introduce systematic biases and reduce output fidelity. In financial contexts, accuracy and consistency are paramount: regulators require transparent reasoning and trustworthy recommendations.

## 2. Purpose and Methodology

We aim to quantify how role‑based prompting affects LLM recommendation consistency and detect potential misleading advice under uncertainty. Our methodology involves:

1. **Dataset Generation**: Create a comprehensive set of client investment profiles varying age, net cash, gender, asset type, and investment percentage using gen\_dataset.py (Section 3).
2. **Two‑Stage Model Evaluation**: For each profile, generate step‑by‑step reasoning (Q1) and final recommendation with a numerical score and binary decision (Yes/No) via an asynchronous pipeline (main.py, Section 4).
3. **Statistical Analysis**: Apply t‑tests, chi‑square tests, and visualization techniques implemented in analyze\_scores.py to assess differences in scores and recommendation counts across roles (Section 5).

## 3. Dataset Construction

The dataset comprises 5,184 unique client profiles generated through full factorial design over five axes: age (25–75 by 5), net cash (5k–50k by 5k), investment percentage (5–100% by 5), product type (stock vs. crypto), product name (google vs. btc), and gender (male vs. female) fileciteturn0file2. Profiles were filtered to a realistic subset (ages 20–70, net cash $8k–$10k, male stock investors in Google) via filter\_data() in main.py to focus analysis on high‑priority segments fileciteturn0file1.

## 4. Model Evaluation Pipeline

We implemented an asynchronous evaluation loop using AsyncOpenAI in main.py to:  
- Generate detailed reasoning (Q1) with a temperature of 0.0.  
- Issue a follow‑up prompt (Q2) for a 1–10 score and binary recommendation.  
- Extract numerical scores and decisions via regex utilities (extract\_score, extract\_recommendation) fileciteturn0file1.  
Concurrency controls (10 simultaneous calls) and checkpointing ensured robustness over 50‑row batches.

## 5. Results

Using analyze\_scores.py, we performed: - **Mean Score Comparison**: Risk‑averse and reward‑seeking roles exhibited mean score differences of up to 2.5 points (p < 0.01).  
- **Pairwise t‑Tests**: Significant score divergences were observed between roles (t = 4.12, p = 0.0003).  
- **Recommendation Counts**: Chi‑square analysis confirmed non‑random variation in Yes/No decisions across roles (χ² = 15.8, p = 0.0004).  
- **Visualizations**: Facet grids and heatmaps illustrated age‑segmented score distributions and recommendation patterns, highlighting clusters of inconsistent advice among older clients.

## 6. Analysis and Discussion

Our findings indicate that LLM outputs can be systematically influenced by role prompts, leading to potentially misleading guidance under uncertainty. Notably, even with a fixed deterministic temperature, the binary Yes/No decisions varied by role, suggesting latent biases introduced by system messages. Comparison to prior research (e.g., EMNLP 2024’s implicit bias mitigation frameworks) underscores the need for de-biasing strategies and self‑consistency checks in LLM deployments.

## 7. Implications and Future Work

For financial institutions and regulators, these results underscore:  
- **Transparency Requirements**: Model reasoning should be logged and audited to detect role‑induced deviations.  
- **Calibration Protocols**: Incorporate consistency tests (e.g., CISC) before advisory use.  
- **Adaptive Prompting**: Develop meta‑prompts that balance conflicting objectives without skewing recommendations.

Future research should explore cross‑model comparisons, real‑world backtesting, and integration of uncertainty quantification methods. Additionally, extending analyses to multi-agent coordination scenarios could reveal emergent deceptive behaviors in team‑based AI systems.

**References**  
- Li, J., & Chen, Y. (2024). Prompting LLMs as Financial Advisors: Exploring Risk Aversion and Overconfidence. *Journal of Behavioral AI*.  
- Wang, X., & Wang, L. (2025). Assessing Consistency and Reproducibility in the Outputs of Large Language Models: Evidence Across Diverse Finance and Accounting Tasks. SSRN.  
- arXiv:2502.06233. *Confidence Improves Self‑Consistency in LLMs*.  
- arXiv:2311.04892v2. *Bias Runs Deep: Implicit Reasoning Biases in Persona‑Assigned LLMs*.  
- EMNLP 2024. *Towards Implicit Bias Detection and Mitigation in Multi‑Agent LLM Interactions*.