ABSTRACT

Title of Dissertation: Creating and Evaluating Human-grounded AI Tools for Challenging and Trustworthy Text

Yoo Yeon Sung
Doctor of Philosophy, 2025

Dissertation directed by: Naeemul Hassan
College of Information
Philip Merrill College of Journalism

Dissertation co-directed by: Jordan Boyd-Graber
Department of Computer Science
College of Information
Language Science Center
Institute of Advanced Computer Studies

Natural language processing (NLP) enables computers to understand and interact with human language, and it is increasingly deployed in Artificial Intelligence (AI) applications like chatbots and voice-operated GPS systems. Although NLP models often *claim* super-human performance in these services, the models often struggle to handle the complexity and variability of real-world data. They typically lack the flexibility in handling vagueness and understanding different contexts as users expect, which limits their reliability in assisting or collaborating with humans in daily lives. This unreliability often stems from models being evaluated primarily on narrow metrics and benchmarks that do not capture the complexities of real-world interactions. As a result, these systems may perform well under controlled conditions but fail in open-ended,

ambiguous, or dynamically shifting situations. To address this limitation, it is essential to develop robust evaluation methods and prioritize AI trustworthiness before deployment. Thus, this dissertation introduces a set of human-grounded frameworks that incorporate human input—user responses, annotations, or human-created artifacts—to evaluate and enhance the robustness and trustworthiness of AI systems in real-world deployment settings.

We begin by examining human-grounded approaches that enhance the evaluation methodologies of NLP systems. First, it enriches benchmark datasets with human inputs. It supports the design of benchmark examples that reflect real user queries, increasing realism. Second, it allows us to measure human baseline performance on a specific task—capturing their skill level—which facilitates direct comparison with the model's performance. Third, it includes subjective human judgments from the real-world users, capturing diverse user interpretations of NLP tasks. These dimensions are often overlooked in automated evaluation. Moreover, human-grounded methods can integrate user standards and values directly into the model development process as a reference point to understand model's intended use from a user perspective. For example, incorporating human-designed criteria that resonate with human standards helps to refine and guide the erroneous model behaviors. When models fail, these methods can promote interpretability, allowing users to understand the cause of failures and offer tailored suggestions. Importantly, as models respond to users' suggestions, such dynamic interaction forms a feedback loop: users guide the model, and the model, in turn, refines its outputs based on that user input. Rooted in user norms and values, this loop not only improves model performance but also enhances transparency and controllability by facilitating continuous adjustment between human and machine.

Building on the premise that human-grounded evaluation is essential for user-centric NLP systems, the first two chapters of this dissertation introduce pipelines and metrics to generate

challenging benchmark datasets that better reflect real-world complexity. The first chapter introduces **a human-in-the-loop (HITL) metric that quantifies the adversarial robustness—how consistently the examples are more difficult for models than humans—of a benchmark**. The proposed metric goes beyond standard accuracy or F1 scores by incorporating measures of human difficulty, example ambiguity, and response diversity, thereby capturing aspects of task realism and user perception that are often overlooked in conventional benchmarks. In this measurement process, we account for varying skill levels across expert humans and models while ensuring benchmark examples are well-posed. This metric offers a practical way to track benchmark robustness over time. The second chapter introduces another pipeline to create challenging artifacts that capture **natural adversarialness, directly reflecting real-world tasks that are inherently difficult and subjective for models, and even for humans**. We designed an annotation scheme that effectively elicits real-world user subjective judgments as labels for training and evaluation. Contemporary model results show a critical gap between the current model capabilities and real-world performance demands.

Expanding beyond robustness evaluation via challenging benchmarks, the final two chapters turn to evaluation of how trustworthy the model is; we focus on model calibration and interpretability. The third chapter aims to tackle human mistrust in AI models by evaluating **how well NLP models are calibrated compared to humans, using both humans and models' confidence response data**. We propose a HITL benchmark creation pipeline and metric designed to evaluate models' correctness and confidence accounting for human performance. The models were generally more overconfident when they were incorrect, in contrast to humans. Finally, the fourth chapter introduces **a user-grounded evaluation framework of multi-agent systems. This enables a granular, user-informed assessment based on user-defined standards**, and

thereby enhances the transparency and diagnostic clarity of agent behaviors. This framework can make agent failures interpretable and provide actionable feedback for users, encouraging a more controllable and trustworthy interaction.

In sum, these works take an integrated approach to human-grounded evaluation and development of NLP systems. It centers on creating challenging and naturally adversarial datasets, and proposes user-informed metrics and methods to measure model limitations. They contribute to the advancement of more robust, trustworthy, and interpretable language technologies, which ultimately can lead to better alignment with human needs.

CREATING AND EVALUATING HUMAN-GROUNDED AI TOOLS FOR CHALLENGING AND TRUSTWORTHY TEXT

by

Yoo Yeon Sung

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2025

Advisory Committee:
Professor Naeemul Hassan, Chair/Advisor
Professor Jordan Boyd-Graber, Co-chair/Co-advisor
Professor Cody Buntain
Professor Hal Daumé III
Professor Ernesto Calvo

Acknowledgments

This research journey has been both demanding and rewarding, and reaching the end of it would not have been possible without the encouragement and support of many people. I am truly thankful to all who have guided and helped me along the way.

Above all, I am grateful to my parents and my brother: Habum Sung, Kyungok Lee, and Hansang Sung. They have supported me in every way imaginable, always ready to lift me up when things were difficult, and consistently reminding me to trust myself, stay confident, and aim high. Their strength and love have been, and will continue to be, my foundation, and I know their faith in me will remain unshakable.

I would also like to extend my gratitude to my advisors, Dr. Jordan Boyd-Graber and Dr. Naeemul Hassan. I am especially grateful to Jordan for his constant willingness to engage with my research, for the many QANTA game exhibitions we organized together, and for the hands-on guidance he provided through our regular weekly meetings. These experiences were not always easy, but they shaped me into a researcher who can take real-world problems, frame them as research questions, and work toward the most impactful answers. I believe that his unique advisory style has pushed me to grow as a researcher who seeks well-founded solutions that not only advance scientific understanding but also address pressing practical problems.

I am equally thankful to Naeemul, whose encouragement and insights have supported me through some of the most difficult moments of my Ph.D. journey. His thoughtful feedback consistently pushed me to think more deeply and refine my ideas, while his mentorship helped me

stay focused on what truly matters in research. Beyond the academic guidance, his patience and steady support gave me the confidence to persevere whenever I doubted myself, and I am deeply grateful for the time and energy he invested in my growth as a researcher.

I also want to express my sincere gratitude to my dissertation committee members, Dr. Cody Buntain, Dr. Hal Daumé III, and Dr. Ernesto Calvo. Their generous support and constructive feedback have contributed greatly to the development of my dissertation. I am especially grateful to Dr. Cody Buntain for meeting with me regularly and offering countless suggestions that helped refine my research direction and strengthen the logic of my work. Dr. Hal Daumé III has been a research mentor, offering thorough feedback and guidance on my projects, which pushed me to think deeply about the essential values that must be studied in human-centered AI research. Dr. Ernesto Calvo has guided me to think about misinformation and the integration of human values into technical research questions from a broader, more societal and philosophical perspective. It has truly been a privilege to learn from such inspiring teachers and mentors on my dissertation committee.

My deepest gratitude goes to my fellow labmates in CLIP, whose warmth and support created a welcoming environment throughout my Ph.D. journey, as well as to the CJ Lab members and the iSchool community. I am especially grateful to my dearest CLIP labmates: Eleftheria Briakou, Sweta Agrawal, Pranav Goel, Michelle Yuan, Trista Cao, and Pedro Rodriguez, who shared the early years of my Ph.D. with me and helped me adjust to the group. I would also like to thank Yu Hou, Nishant Balepur, Dayeon Ki, Dang Nguyen, Connor Baumler, Navita Goyal, Hyojung Han, Maharshi Gor, Atrey Desai, and all the other CLIP labmates who joined later and stood by my side for the rest of my Ph.D. journey. I am also grateful to the members of the Computational Journalism Lab: Rony Main Uddin, Mahfuzul Haque, and Mohammad Ali, as well

as to Nitzan Koren, my dearest iSchool cohort.

Outside of research, I was fortunate to be surrounded by a wonderful community of Korean friends in Maryland. I am deeply grateful to my friends: to Hyunki Kim, who was a constant source of encouragement and care; to Hongjun Kim, whose steady support and thoughtful advice gave me strength; to Jaehoon Choi, with whom I shared countless fun and memorable social moments; and to Kyungjoon Lee, who became the beloved "CS dad" for many Korean international students. They not only gave me an outlet to relieve stress but also reminded me how to find joy and balance during the most difficult times of my Ph.D. journey. I am also thankful to Yesop Lee, Dakyung Yang, Yoonkyung Sohn, Kyungyeon Lee, Jiyeon Min, and Geonsun Lee, whose kindness made this journey far more fruitful.

I am also deeply thankful to my long-term friends, who have been a constant source of support and encouragement over the past decade. Jinhee Yoon, Soojin Jung, Ganghyun Kim, Jungmin Kwon, Eungyung Ju, Nayeon Kim, Emilene Ann Badalamenti, Soomin Hong, Seungyeon Lee, Soyeon Lee, and Miya Yoon have remained by my side through different stages of life, and their enduring friendship has sustained me beyond the boundaries of research and academia.

Finally, I am grateful for the many small but meaningful moments: conversations, collaborations, and acts of generosity that shaped my Ph.D. journey in ways that were clear at the time and in ways that revealed themselves afterward. I will always carry the lasting influence of these experiences with me.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Motivation

Since the release of ChatGPT in 2022, large language models (LLMs) have been widely adopted in various AI applications (OpenAI, 2023): service chatbots (Graham et al., 2025), virtual assistants (Tetteh et al., 2025), and voice-based navigation systems (Akanfe et al., 2025). This trend flourished through 2025, as tasks previously performed by humans have been increasingly delegated—or even entirely replaced by—to AI models (Zhuang, 2025). As a result, AI models are more deeply embedded into daily human communications and lifestyles (Brandtzaeg et al., 2025). However, in practice, they fail to cope with the complexity (Ma et al., 2021a), ambiguity (Min et al., 2022), and variability inherent in human language and interaction (Bowman and Dahl, 2021a). On the other hand, benchmark datasets often indicate that LLMs outperform human baselines; high performance on these datasets do not always guarantee effectiveness in real-world applications. This growing disconnect between benchmark success and real-world utility underscores the need for more robust evaluation methods. These methods should not only target accuracy, but also ensure aligning with human expectations and practical end-user demands.

One typical approach to building robust evaluation methods is to develop challenging adversarial datasets: datasets composed of challenging artifacts that reveal model weaknesses. These datasets are carefully designed to test model capabilities before it is deployed, thereby

improving its robustness in real-world applications. These datasets include examples that simulate unexpected and challenging situations for models, including examples that elicit harmful (Perez et al., 2022), unsafe (Quaye et al., 2024), or incorrect (Goodfellow et al., 2015) outputs.

Among these adversarial datasets, we first focus on creating adversarial examples that humans can easily solve but that models cannot, due to fundamental discrepancies in reasoning, contextual understanding, or commonsense inference (Ilyas et al., 2019; Tsipras et al., 2019; Engstrom et al., 2020; Biggio et al., 2012). These types of examples help isolate human and model differences, revealing critical failure modes of models. Additionally, as LLMs advance rapidly, many previous challenging adversarial datasets have become obsolete. Tasks that once revealed failure cases for models are now easily solved (Kiela et al., 2021; Recht et al., 2019; Bowman and Dahl, 2021a). A significant challenge in this space is the lack of principled methods for determining when adversarial datasets become outdated, as well as absence of a standardized metric for assessing which datasets best reveal the persistent gap between human and model performance. In response to these challenges, Chapter 3 of this dissertation thus explores how to evaluate and design challenging human-grounded datasets that can remain meaningful over time, by reflecting realistic human language and human performance. By grounding dataset difficulty in human likelihood and baseline ability, and targeting scenarios that are consistently difficult for humans while solvable for humans, we aim to avoid premature saturation of the adversarial benchmarks.

Another type of adversarialness we target is the naturally-adversarial datasets that emerges organically in real-world settings—what we refer to as natural adversarialness. Unlike synthetically constructed adversarial examples, these cases reflect genuine challenges users face when interacting with AI: implicit assumptions, underspecified intent, or context-sensitive interpreta-

tions. Traditional benchmarks typically overlook this kind of difficulty, favoring clearly defined tasks with singular answers and low ambiguity. This gap motivates a shift toward incorporating human subjectivity into evaluation, capturing the diversity of human judgments, disagreement patterns, and real usage contexts. In Chapter 4 discusses solving a high-stake task that grounds evaluation in how people naturally interpret, question, and disagree over language. This way, we can better assess whether models are prepared for the complexity of real-world applications and guide the development of systems that are adaptive.

Furthermore, as LLMs are increasingly integrated into high-stakes applications, the lack of robust evaluation measures often forces users to interact with models that are not yet sufficiently reliable (Caruana, 2019; Deng et al., 2025). One of the challenges in addressing the issue lies in effectively evaluating and improving model trustworthiness, especially when their behaviors significantly deviate from human expectations (Ilia and Aziz, 2024). A major concern in this context is users' tendency to overtrust AI systems due to model's miscalibrated confidence. Users frequently interpret high model confidence as an indicator of high accuracy (Krause et al., 2023; Stengel-Eskin and Van Durme, 2023; Liu et al., 2024b; Si et al., 2023) even though LLMs are often overconfident in incorrect predictions—particularly in ways that humans are not. This overconfidence creates a mismatch between perceived and actual model performance, leaving users unprepared (Li et al., 2024a). To tackle this challenge, Chapter 5 of this dissertation proposes human-centered methods to create a benchmark dataset and a metric that compare model confidence to human confidence judgments, aiming to identify and mitigate overconfidence in LLMs and ultimately enhance their trustworthiness in real-world decision-making scenarios.

These trust-related issues are even more acute in multi-agent systems. While these systems may seem to solve complex problems (Xi et al., 2025; Wang et al., 2024a), they remain

vulnerable to reasoning failures (LangChain, 2013; Arawjo et al., 2024) and brittleness in the real-world (Sumers et al., 2023; Jaeger et al., 2013). Such systems often produce outputs that deviate from human expectations, requiring human intervention to detect and correct agent-level errors. However, providing user feedback is difficult in multi-agent systems where reasoning processes are often opaque and divergent from intuitive human logic (Wang et al., 2023b). Furthermore, limited interpretability—especially when reasoning steps are entangled in the system's final output (Cheng et al., 2024)—hinders users from diagnosing failures or offering actionable corrections (Grunde-McLaughlin et al., 2025a). This will eventually undermine the human-AI trust, which is essential for collaboration and task success. Given these limitations, Chapter 6 of this dissertation aims to advance model trustworthiness from a human-centered perspective, supporting interpretable evaluation and aligning with end-user needs.

1.2 Roadmap: Human-Grounded Approaches for Robust AI Evaluation

This dissertation outlines directions for developing robust AI evaluation methodologies with human-grounded approach. In Chapter 2, we elaborate the background and motivations of human-grounded AI evaluation, including the needs for human skills and diversity, human subjectivity, and user values. We also expand on prior works and theoretical background, such as item response theory and calibration-based evaluation methods, to provide a clearer understanding of how our proposed approach addresses existing gaps in the literature. Then, we discuss the pipelines for effectively creating two types of adversarial benchmarks that can evaluate models (Chapter 3 and Chapter 4). Followingly, we discuss how we can build evaluation human-grounded evaluation methods that support AI-user trust (Chapter 5 and Chapter 6). We argue that human

input and feedback are indispensable for bridging the gap between a model evaluation and real-world performance. This dissertation contributes to the development of more robust assessment pipelines and guide the design of AI systems that better serves human expectations and user needs.

1.2.1 Background Roadmap

We begin by outlining LLMs' core applications as well as their limitations in real-world deployment, and highlighting the shortcomings of existing automated evaluation metrics in capturing human-centered performance dimensions (§ 2.1). We then examine the evolution of human-in-the-loop (HITL) evaluation frameworks, with particular emphasis on integrating skills and benchmarking model outputs against human performance standards (§ 2.2). We use item response theory (Lalor et al., 2016, IRT), which provides a psychometric foundation for modeling human and model performance jointly by capturing task difficulty and respondent ability, allowing us to better characterize where models succeed or fail relative to human baselines (§ 2.2.2.1). Next, we discuss the importance of human subjectivity in evaluating real-world adaptability, showing how subjective annotations, disagreement patterns, and nuanced judgments contribute to richer and more realistic evaluations—especially in complex tasks like misleading video headline detection (§ 2.3). We then discuss model calibration literature, which is a concept to assess how well a model's confidence scores reflect the actual correctness of its predictions, which is crucial for interpretability, reliability, and deployment in user-facing applications (§ 2.1.2). We also review recent advances in multi-agent systems and explore how user values embedded within agentic interactions can foster more interpretable and trustworthy NLP evaluation strategies (§ 2.4). Having presented the background, the subsequent four chapters of this dissertation aims to achieve two

overarching goals: the creation of challenging, human-grounded benchmarks and the development of trustworthy, human-aligned AI systems. Together, these chapters demonstrate how integrating human-grounded signals at multiple levels—data, metrics, and interaction framework—can drive the development of more robust and responsible NLP systems.

1.2.2 Pipelines for Challenging Benchmark Creation

A robust evaluation pipeline should thoroughly assess model vulnerabilities to prepare systems for the diverse and unexpected inputs encountered in real-world settings. Although current evaluation methods typically rely on challenging benchmarks or leaderboard performance, many existing datasets lack rigorous quality control and do not accurately reflect the nuances of real human input. For example, a generative QA model may excel on traditional benchmarks but struggle when faced with human-written questions like "*Very thin layers of this substance that look like oil spills are nicknamed for grease*," incorrectly answering "graphene" when the correct answer is *ice*—a clear distinction that is evident to skilled human respondents. Moreover, automated adversarial generation pipelines, when designed without human input, are likely to produce examples that rely on superficial noise or syntactic perturbations. These artificial adversaries may not reveal true model limitations, and instead lead to misleading evaluations.

To address this, in Chapter 3, we explore the adversarial robustness by collecting human-authored examples using HITL framework and quantifying the gap between human and model performance in terms of their respective skills. This way, we can conduct a more realistic assessment of benchmark difficulty by directly comparing model vulnerabilities with human capabilities on prompts that reflect authentic, real-world usage. We intentionally invite expert

writers to directly craft adversarial examples with model responses. We provide an interface that renders immediate model responses so that experts can iteratively refine their prompts to expose model weaknesses while ensuring the examples remain plausible and representative of real-user queries. Importantly, collecting human answers enables measuring the gap between human and model performance, which can then be used to **quantify adversarialness—defined as examples that are clear for humans but difficult for models.** We apply this idea by collecting both human and model responses and using item response theory (Lalor et al., 2016, IRT) to measure the difficulty gap relative to their abilities, while also penalizing ambiguity inferred from their responses. We propose a new metric that captures these human-grounded characteristics and measure its application through a benchmark constructed via expert-authored adversarial generation.

Then, in Chapter 4, the benchmark creation is extended to a practical and high-stakes task: detecting misleading video headlines in the context of misinformation. **This task reflects real-world complexity and subjectivity, making it well-suited for adversarial evaluation.** In this setting, human subjective judgments reveal naturally adversarial cases that expose current models' weaknesses. We develop a hierarchical annotation framework to capture the subjective judgments of human annotators. The resulting benchmark provides insight into model robustness under real-world conditions and highlights performance gaps in handling subjectivity and ambiguity. We argue that grounding adversarial benchmark construction in human judgment ensures a more faithful evaluation of whether models are sufficiently reliable to be deployed in high-stake scenarios.

1.2.3 Evaluation Frameworks for Trustworthy AI

Chapters 5 and 6 shift focus to the development of human-grounded evaluation frameworks that assess and improve the trustworthiness of AI systems in interactive settings. In Chapter 5, to identify points of divergence between human and model expectations, often leading to human's mistrust in models, **we evaluate model calibration by comparing their confidence levels, correctness, and abstention behaviors**. This allows us to better understand how and when models misrepresent their certainty relative to human judgment. Specifically, we investigate *model calibration*, which is a key aspect of user trust. Users often *rely on model confidence scores, expecting that high-confidence outputs are more likely to be correct*. However, prior work has shown that LLMs tend to be overconfident, especially when they are wrong (Li et al., 2024a). To evaluate this systematically, we introduce a human-AI calibration benchmark designed as a competition between models and human experts, using progressively adversarial examples—each crafted to be incrementally more easier and solvable for the model. In addition to collecting answers, we infer human confidence levels using response latency as a proxy. This benchmark includes detailed annotations of correctness, confidence, and abstention behaviors for both humans and models. Specifically, we analyze the rate at which each subject abstains from answering and investigate how this abstention correlates with their overall correctness. Using this data, we propose a new metric that evaluates LLM calibration by directly comparing model predictions to human confidence-adjusted baselines. Our findings reveal a consistent trend: models exhibit significantly higher overconfidence than humans, particularly when incorrect, posing a risk in high-stakes decision-making scenarios.

Moreover, in Chapter 6, to support trust recovery in AI models, we propose a user-grounded

evaluation framework that incorporates user values and standards to uncover model limitations and thereby enables an interpretable feedback mechanism, making model failures understandable to users. In this work, we aim to improve trust and interpretability in multi-agent systems, where reasoning steps are often opaque and distributed across agents. We propose **a user-grounded framework that evaluates individual agent outputs based on user-designed criteria for the given task**. Then, using human binary annotations for each agent failure, we train an external predictor that assesses each agent's failure, incorporating features such as planner structure, per-agent uncertainty scores, and agents' alignment with user criteria. This granular, user-centered evaluation enables identification of agent-level errors that traditional system-level evaluation would overlook. Moreover, it provides intuitive and actionable feedback to users, supporting trust calibration and error mitigation in complex AI systems.

Finally, Chapter 7.1 concludes the dissertation by reiterating the critical role of human-grounded evaluation and development in aligning NLP technologies with real-world user needs. Through the integration of human perspectives in both benchmark design and model assessment, this work contributes to the creation of more robust, trustworthy, and interpretable AI systems, which are better equipped to function reliably in practical, everyday contexts.

Chapter 2: Background

This chapter provides background information on topics relevant to the dissertation. We first review the applications of LLMs and their limitations in real-world deployment, and the shortcomings of existing automated evaluation metrics (§ 2.1). We then discuss how HITL frameworks have evolved to enhance NLP evaluation, with a focus on integrating human skills and benchmarking model performance against human expertise (§ 2.2). Next, we explore the role of human subjectivity in evaluating real-world adaptability, including how subjective annotations and nuanced judgments can inform evaluation in complex tasks such as misleading video headline detection(§ 2.3). Finally, we review recent work on multi-agent systems and examine how user values embedded in agentic system development can support more interpretable and trustworthy system evaluation (§ 2.4).

2.1 Challenges in AI Evaluation and Applications

2.1.1 Capabilities and Limitations of LLMs in Real-World Applications

LLMs have shown remarkable capabilities across a wide range of natural language processing tasks, including retrieval, entailment, reasoning, and comprehension, enabling applications in essay composition, code generation, financial reporting, and journalism (Touvron et al.; Chowdh-

ery et al.; Brown et al., 2020). A key factor behind this success is their ability to perform zero-shot and few-shot learning by conditioning on instructions or a handful of examples (Wang et al., 2023a). Through prompting (Liu et al., 2023b), where an input x is reformulated into a textual prompt x' and completed by the model to produce an output $\hat{x}$, LLMs can adapt to novel tasks without the need for extensive labeled data (Hofstätter et al., 2023). Despite their promise, however, several critical challenges complicate their practical deployment. Hallucination remains a major concern, particularly in high-stakes domains like healthcare and software engineering, where fabricated facts or subtle bugs can lead to serious consequences (Ji et al., 2023b; Liu et al., 2024a). Additionally, when user queries span multiple domains, LLMs often struggle with domain-specific reasoning, resulting in vague or inaccurate outputs (Wang et al., 2024b). This issue is especially problematic in open-domain QA systems that rely on external retrieval sources such as Wikipedia, where model-generated misinformation can further degrade quality (Goldstein et al., 2023). Even in tasks requiring logical reasoning or entailment, LLMs can produce false but convincing outputs, misleading users with overconfident and incorrect inferences (Ji et al., 2023a). These limitations can often be traced back to the lack of high-quality, diverse, and human-like benchmark datasets that truly capture the complexity of real-world language use (McIntosh et al., 2025). In response, HITL methods have gained attention as a promising avenue for creating more robust, diverse, and human-grounded evaluation protocols that better reflect the nuanced demands of real-world deployment scenarios.

2.1.2 Shortcomings of Automated Evaluation Metrics and Static Benchmarks

Before exploring HITL methods, we review traditional automated metrics and benchmarks that are typically used to evaluate current models. Despite the wide use of automated metrics in NLP, they remain fundamentally limited in their ability to capture the full depth of human language understanding (Gehrmann et al., 2021; Resnik and Lin, 2010).

Metrics such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) are commonly used to evaluate text generation tasks by comparing a model's output to one or more reference texts, typically through n-gram overlap. BLEU measures the precision of n-grams (sequences of words) in the generated text relative to the reference, penalizing overly short outputs through a brevity penalty. ROUGE, on the other hand, emphasizes recall as to how much reference content appears in the generated text. It includes several variants such as ROUGE-N (n-gram recall), ROUGE-L (longest common subsequence), and ROUGE-S (skip-bigram), making it particularly useful for tasks like summarization where coverage of key content is important. Similarly, accuracy is used in classification tasks by checking whether the predicted label exactly matches the gold label. However, these metrics are often limited in assessing model performance on open-ended tasks such as abstractive summarization, open-domain question answering, and dialogue systems (Reiter and Belz, 2009). In such tasks, there may be multiple valid outputs that differ lexically but are semantically equivalent or contextually appropriate. Especially in cases where human interpretations vary, these surface-level metrics fail to account for meaning, nuance, or pragmatic appropriateness (Schmidtová et al., 2024). As a result, models that achieve high scores on synthetic or rigidly defined benchmarks may still generate incoherent and misleading responses when evaluated from a real user's perspective (Kiela et al., 2021; Ma et al., 2021a).

This disconnect between benchmark performance and real-world reliability has led to increased attention on model calibration—a promising direction for evaluating trustworthiness in LLMs by measuring how well their predicted confidence aligns with actual correctness. A high correlation between confidence values and correctness is becoming increasingly important, as humans often rely on model confidence to assess the credibility and accuracy of outputs. This relationship is especially critical as AI systems are increasingly used to support human decision-making. Models whose confidence estimates do not correspond well with correctness are referred to as *miscalibrated* models. As AI systems are increasingly used to support human decision-making, their trustworthiness hinges not only on correctness but also on the alignment between confidence and accuracy. Miscalibrated models can be harmful, particularly when they express high confidence in incorrect outputs and users are unable to independently verify the answer (Stengel-Eskin et al., 2024; Khurana et al., 2024). This risk is amplified in complex domains, where users may defer to model suggestions despite uncertainty.

Calibration Evaluation Metrics. Language models tend to be overconfident in their predictions, which can lead to undue trust or erode user confidence in language models (Zhou et al., 2024). Proposed methods to measure model calibration include using raw probabilities (Xiong et al., 2024a), separate confidence predictors (Ulmer et al., 2024), verbalized confidence scores (Tian et al., 2023; Band et al., 2024), or natural language expressions of uncertainty (Zhou et al., 2023). Moreover, Si et al. (2022) improve calibration in question answering by leveraging consistency across multiple training checkpoints, assigning higher confidence to predictions that remain stable throughout training. Chapter 5 extends existing calibration metrics, such as ECE (Naeini et al., 2015) and Brier scores (Brier, 1950), by introducing a metric for calibration measurement.

ECE. ECE measures the weighted average over the absolute difference between accuracy and confidence. To compute this, we first split confidence values into M bins equally. B_m represents the confidence set of the m^{th} bin. N is the total number of instances across the dataset:

$$\text{ECE} = \frac{1}{N} \sum_{m=1}^{M} \left| \sum_{t \in B_m} g_t - \sum_{t \in B_m} c_t \right|$$

We use g_t to denote the correctness (1 if correct, 0 otherwise) and c_t to denote the corresponding model's confidence for instance t.

Brier Score. Brier measures the mean squared difference between the predicted probability and the actual binary outcome, measuring how well the predicted confidence aligns with the true correctness of the answer. A lower Brier score indicates better calibration, as it reflects more accurate and well-calibrated probability estimates:

$$\text{Brier Score} = \frac{1}{N} \sum_{t=1}^{N} (c_t - g_t)^2$$

While these metrics offer calibration measurement under classification settings, they fail to capture real-world variability and user interpretation, particularly in open-ended generative tasks. In particular, they fall short in capturing the variability and subjectivity inherent in real-world, open-ended tasks. In generative scenarios like open-domain question answering, models often produce fluent yet incorrect answers with unjustified confidence, leading to outputs that misalign with human judgment and potentially undermine user trust. As a result, evaluating miscalibration in these open-ended contexts remains a major challenge.

Thus, Chapter 5 introduces a metric, a human-grounded calibration metric. Unlike traditional

approaches that assess calibration, it bases model evaluation to how early humans and models are able to answer a question correctly as clues are revealed. This allows us to assess whether a model's confidence trajectory aligns with *human certainty and decision timing*, not just ground-truth accuracy.

Static Benchmarks. Beyond the limitations of automated metrics, many static benchmarks, especially those created without human adversaries or iterative feedback loops, also fall short in diagnosing model behavior effectively (Ribeiro et al., 2020; Nie et al., 2020). These benchmarks often consist of synthetic or templated examples that fail to reflect the diversity, ambiguity, and nuance found in natural human language (Gardner, 2024; Pavlick and Kwiatkowski, 2019). As a result, they tend to emphasize surface-level difficulty (e.g., lexical variation or syntactic noise) rather than targeting the reasoning gaps or contextual misunderstandings that underlie real-world model failures (Kaushik et al., 2021; Utama et al., 2020). Moreover, because static benchmarks are typically constructed without observing model responses, they may include examples that are either trivially easy or unnaturally difficult, leading to skewed evaluations (Bowman and Dahl, 2021b; Belinkov and Bisk, 2017). In particular, such datasets rarely include human performance baselines, making it difficult to assess whether model behavior meaningfully diverges from what a skilled human would produce (Lalor et al., 2019).

The proposed solutions in this dissertation emerge from the precaution to evaluate models in ways that reflect real-world usage, we must embed human reasoning, variability, and feedback into the evaluation process itself. HITL methods offer a practical mechanism to surface nuanced failure modes by allowing human annotators to craft, refine, or respond to model outputs in real time. This interactivity produces more diagnostically informative test cases that expose brittleness in

model behavior—especially in the face of ambiguity, disagreement, or shifting context. Similarly, calibration-focused evaluation offers a pathway to assess not just what models predict, but how confident they are—a critical factor in determining whether users can rely on model outputs. When anchored in human baselines, calibration metrics help quantify alignment between model confidence and human certainty, closing the gap between benchmark performance and real-world trust. Thus, the studies presented in the subsequent chapters embed the human perspective where traditional metrics fall short, enabling richer, more realistic, and more trustworthy evaluation.

With these motivations, the following section (§ 2.2) shifts toward HITL benchmarking, where human annotators iteratively—by creating and responding to examples—engage with model outputs to design more challenging, realistic, and diagnostically informative test cases (Ma et al., 2021b; Perez et al., 2022).

2.2 Integrating Human Skills to Reveal Model Vulnerabilities

To address the shortcomings of static benchmarks and automated metrics, this dissertation explores HITL evaluation as a dynamic and human-grounded alternative. For example, the question, "How many non-pet characters live in SpongeBob's neighborhood?" was answered with "5" instead of the correct answer "3" by ChatGPT, which infers that model likely overcounts by including characters such as Gary (a pet) or background sea creatures, despite the question clearly specifying "non-pet" characters. This suggests that the model fails to reason about entity types and contextual constraints, relying instead on memorized associations. Such failure modes are rarely detected by static benchmarks but are effectively surfaced through HITL adversarial authoring, where human writers craft questions that are unambiguous for humans yet expose

model's misinterpretation. This section outlines how integrating expert-authored prompts and skilled human responses can lead to more realistic, challenging, and diverse benchmarks, enabling more faithful evaluations of model performance.

2.2.1 Designing Human-in-the-loop Benchmarks

HITL benchmarks directly engages human authors and annotators to author task instances that reflect natural linguistic variation, conversation implications, and domain contexts to challenge models (Bowman et al., 2022). For example, authors craft questions that contain vague phrasing that can lead to multiple interpretations (Min et al., 2020), which is natural in real-life interactions. Moreover, human-curated examples tend to include diverse social and cultural perspectives (Parrish et al., 2021). Thus, grounding benchmark examples in user authenticity and expectations improves the practical validity and realism of model evaluation.

2.2.1.1 Curated Examples by Highly-skilled Humans

As some tasks require professional knowledge, some benchmarks are created through sole expert labor (Wang et al., 2018; Naik et al., 2018). However, it is extremely expensive and time-consuming to recruit domain experts across fields. Also, because a pool of domain experts can take control of ways to craft a dataset and excessively focus on the task at hand; recruiting professionals across diverse domains is time-consuming and expensive (Bowman and Dahl, 2021a). Moreover, expert-curated datasets may reflect narrow perspectives, unintentionally biasing models toward specific knowledge or reasoning patterns (Jia and Liang, 2017; McCoy et al., 2020). In order to mitigate these limitations, it is critical to ensure that experts clearly understand the objectives

of the dataset and follow consistent construction guidelines. In Chapters 3 and 5, where we recruited expert writers for benchmark creation, we ensured that the collected examples were evenly distributed across categories and that the writers reflected a diverse range of expertise and skill levels. This balance helps preserve generalizability and minimizes overfitting to a single expert domain.

2.2.1.2 Human-in-the-Loop Adversarial Prompt Generation

A key limitation of previous adversarial datasets is that they are static and created without observing how models actually respond. This disconnect can lead to examples that are either too simple or overly complex, reducing their diagnostic value. More importantly, such datasets often lack human baseline responses, making it difficult to confirm whether the examples are truly adversarial—i.e., harder for models than for skilled human annotators (Lalor et al., 2016; Rodriguez et al., 2021).

To address these limitations, HITL adversarial refinement involves a more interactive process where human authors dynamically engage with model outputs to iteratively craft more challenging examples (Ma et al., 2021b; Kiela et al., 2021). By engaging with models in real time, annotators can systematically probe specific weaknesses and produce high-quality adversarial examples that better reflect realistic failure scenarios. For QA tasks, for example, HITL methods have been used to validate whether models can answer naturally phrased questions that reflect user behavior (Bartolo et al., 2021a). Similar strategies have been applied to visual QA (Sheng et al., 2021), where annotators test model robustness by identifying questions that mislead the model given an image. Further enhancements involve incentive mechanisms (Wallace et al., 2019;

Eisenschlos et al., 2021a), where human authors score points by generating prompts that fool both models and other humans, thus improving adversarial quality through gamified participation. These work resulted in adversarial questions that are scored based on how well they deceive both models and other human participants.

Recent efforts have also expanded HITL refinement into adversarial competition formats, where human authors both write and validate prompts in collaborative or competitive settings. For example, in Eisenschlos et al. (2021a), humans alternate between creating verifiable claims and verifying others' claims, with rewards based on how well their contributions fool models or fellow annotators. These formats improves the overall quality and realism of adversarial benchmarks by directly testing model vulnerabilities through expert reasoning and diverse user perspectives.

Having revealed the models' weaknesses in Bartolo et al. (2021a); Nie et al. (2018); Gururangan et al. (2018), researchers now understand the necessity of human-AI complementarity, where we aim to combine the strengths of both human and AI systems. One common application for such collaboration is creative writing. Lee et al. (2022a) recruited human writers to interact with GPT-3 for collaborative writing, where humans can get suggestions from GPT-3 and make further edits. He et al. (2022) introduces a work where humans are given model-generated queries to search Wikipedia for answers. Here, a dense retrieval model (e.g., DPR) assists humans to seek the most relevant passage to the query. Ray et al. (2019) proposed an explanation-assisted Guess (ExAG), where a user guesses an image from the model and edits questions. Padmakumar and He (2022) deploy an AI model in the loop for modifying user-selected spans to make the draft more descriptive and figurative. Apart from leveraging the "writing" capabilities of language models, in a question-answering context, Boyd-Graber et al. (2018) recruit experts and novices to play trivia games with AI systems as teammates.

Another important direction that builds on these HITL methodologies is incremental adversarial question answering. In this setting, questions are structured to reveal clues in decreasing order. Its design follows the structure of quizbowl questions, which incrementally reveal information by decreasing difficulty, and participants, human or model, must decide when to "buzz in" with an answer. This structure rewards deeper knowledge and calibrated decision-making (Boyd-Graber et al., 2012). Unlike selective QA, which typically evaluates one-off answers (Ferrucci, 2012), the *incremental format captures the decision process over time, offering a more nuanced evaluation of uncertainty and reasoning* (Rajpurkar et al., 2018).

GRACE introduced in Chapter 5, extends this adversarial example generation framework with a HITL adversarial authoring process explicitly designed to test model calibration. While Rodriguez et al. (2019b) used publicly available questions—now too easy for modern models—Wallace et al. (2019) introduced TrickMe, where humans collaboratively wrote adversarial questions. However, TrickMe emphasized overall accuracy rather than calibration. Our dataset creation process is mainly motivated by Kiela et al. (2021); Ilyas et al. (2019); Engstrom et al. (2020), who argue that adversarial benchmarks must be clear for humans and challenge models, ensuring that questions are diagnostic of genuine model errors that are due to model limitations rather than ambiguous or low-quality questions (Min et al., 2020; Yu et al., 2023).

This broader shift reflects a growing consensus in the NLP community: that benchmark evaluation should go beyond static test sets to more closely reflect robustness, generalization, and real-world reasoning (Melis et al., 2017; Biggio et al., 2013; Belinkov and Bisk, 2017; Jia and Liang, 2017). As Tedeschi et al. (2023) argue, many models labeled as "superhuman" may benefit from poorly annotated or biased test sets, overstating their true capabilities. A promising alternative is to develop HITL-driven benchmarks and metrics that require deeper generalization

and adaptability to the real world (Rychalska et al., 2019; Bowman, 2023; Yuan et al., 2023).

2.2.2 Benchmarking Against Human Expert Performance

Building on these interactive and human-centered generation strategies, HITL evaluation also involves benchmarking model performance directly against human performance on the same tasks. This comparative framework provides a grounded baseline for assessing what is realistically achievable and helps calibrate trust in model outputs. For instance, adversarial competitions allow humans and models to be evaluated on the same ground by comparing how well each performs on carefully constructed challenges (§ 2.2.1.2). For example, if a model seems confident but consistently performs below human level (Liu et al., 2024b; Si et al., 2023; Li et al., 2024a), this should be considered to be at high stakes (Caruana, 2019; Deng et al., 2025; Krause et al., 2023; Stengel-Eskin and Van Durme, 2023; Biggio and Roli, 2018). In contrast, when a model matches or exceeds human performance, it builds trust in its outputs. Thus, benchmarking against human experts is a critical component guiding responsible model deployment. Lalor et al. (2019) introduces an IRT-based ranking method that uses these responses to remedy the issue that current evaluation treats each model independently rather than considering relative differences between models or human versus models. In Chapter 3 and 5, we use human responses as important baselines to compare with model behavior and performance.

2.2.2.1 Using IRT for Capturing Adversarialness Using Human Skills

Prior metrics for evaluating adversarial question generation strategies, such as attack success rate (Uesato et al., 2018), distributional similarity (Dathathri et al., 2019), and proximity measure-

ment (Ross et al., 2021) assess algorithmic adversarialness without human validation. In contrast, we identify adversarial examples that pose realistic challenges aligned with *human* skills, not just pathological cases that break models. This requires evaluating how well the examples align with varying levels of human performance, particularly where models fall short, while ensuring that the examples are unambiguous. To capture this, we adopt item response theory (IRT), which models the interactions between subjects' skills—in the QA setting, the subject answering the question could be either a human or a model—and example difficulty. This framework, widely used in psychometrics and educational testing (Lord et al., 1968), provides insights beyond accuracy: it can diagnose question quality as well as skilled subjects.

2PL-IRT. In question answering (QA) tasks, IRT models the probability that a subject correctly answers a question based on their skill and question difficulties. 2PL-IRT (Eq. 2.1) models the probability of getting a question correct as a function of subject *skill* β_i and question *difficulty* θ_j:

$$p(r_{ij} = 1 \mid \beta_i, \theta_j, \gamma_j) = \sigma(\gamma_j(\underbrace{\beta_i - \theta_j}_{\text{skill gap}})), \tag{2.1}$$

where σ is the sigmoid function (Reckase, 1998). The skill gap, $(\beta_i - \theta_j)$, is the difference between the subject i's skill and question j. When a subject's skill is *equal* to the question's difficulty $(\beta_i = \theta_j)$, they have a 50% probability of answering it correctly. Thus, an agent with skill equal to or greater than the question's difficulty level has at least a 50% chance of answering correctly.

The final latent variable is the question *discriminability* γ_j which models how sensitive this probability is to changes in skill gap.[1] This encodes how strongly the question rewards the skill

[1]Perfect discriminability means that any subjects with a positive skill gap will answer the question correctly (Martínez-Plumed et al., 2019) but negative skill gap will never answer the question correctly.

being higher or lower than the difficulty level. The objective of IRT is to estimate the parameters that maximize the correctness probability $p(r_{ij})$.

Advantages of IRT over question success rate. While question success rate (QSR)—the percentage of subjects answering a question correctly—may seem like a reliable measure of difficulty, it can be misleading. A good yet difficult question and an easy yet poorly written question could yield the same QSR, obscuring the true measure of difficulty.

In contrast, IRT evaluates subject responses. Not only does IRT consider the number of humans who answer a question correctly, but it also accounts for *who* answer *which questions*. If the probability of answering a question correctly increases with subject skill, this relationship will naturally correlate with skill β_i and question discriminability γ_j. The model can confidently assign higher probabilities for these questions, while questions that are answered correctly by luck—rather than skill—will have estimated probabilities closer to 0.5, reflecting their lower discriminability.

Consider three questions: q_{ambig} (ambiguous question: "What is a capital of Georgia?" Answer: [Atlanta or Tbilisi]), q_{hard} (hard but well-formed question: "Who founded Tbilisi?"), and q_{easy} (easy question: "What U.S. state has Atlanta as its capital?"). Comparable QSR values may suggest q_{ambig} and q_{hard} have the same difficulty. However, IRT distinguishes them: q_{ambig} has low discriminability ($\gamma_j \approx 0$), resulting in a low $p(r_{ij})$ close to 0.5 regardless of the subject skill, while q_{hard} and q_{easy} are likely to have high discriminability ($\gamma_j \approx 1$) and reverse difficulty (θ_j) values. IRT thus provides a more nuanced evaluation of question adversarialness, capturing its appropriate challenge levels for humans and models while accounting for its "well-posedness" (§ 3.2.1).[2]

[2]Feasibility, another latent variable in IRT, also reflects poor-quality questions when a large proportion of participants answer incorrectly (Rodriguez et al., 2021). However, our approach explicitly accounts for disagreement among highly skilled human subjects (§ 3.2.1).

After accounting for human skills against model skills to create meaningfully adversarial examples, we explore human subjectivity to identify another type of adversarial example, natural adversarialness that arises organically in real-life scenarios. We specifically focus on the role of human subjectivity.

2.2.3 Recruiting High-Skilled Experts Over Moderate-Skilled Humans

We prioritize high-skilled experts when constructing adversarial benchmarks because they provide a stable and reliable baseline. Expert-authored examples are less likely to contain spurious errors, and their depth of domain knowledge allows them to anticipate model weaknesses in ways that moderate-skilled participants may not (Wang et al., 2018; Naik et al., 2018; Jia and Liang, 2017). This ensures that adversarial examples are unambiguous for humans while remaining challenging for models.

At the same time, a natural question arises: if benchmarks are constructed primarily by experts, can they generalize to moderate-skilled users who are the ones actually interacting with these systems? While experts may introduce more challenging and diagnostically useful cases, there is a risk that such datasets disproportionately reflect expert reasoning strategies, making them less representative of everyday user behavior (Bowman and Dahl, 2021a). To address this, we mitigate overfitting to a single perspective by recruiting experts with diverse backgrounds (Raji and Buolamwini, 2019; Bender and Friedman, 2018), designing tasks that remain accessible to humans across different skill levels, and validating questions through human-in-the-loop answers (§3.4.1 and § 5.2.2). In practice, this approach follows prior work in psychometrics and educational testing, where benchmarks are often developed by experts but validated across a spectrum of participant

skill levels to ensure broad robustness (Embretson and Reise, 2013). Thus, while experts play a central role in question authoring, our frameworks in § 3 and § 5 explicitly accounts for the need to balance expert rigor with usability for moderate-skilled humans. In addition, we evaluate models against experts spanning diverse domains and skill levels to ensure that performance generalizes across varying degrees of human expertise. By doing so, we aim to construct benchmarks that are both diagnostically sharp for model evaluation and realistically representative of the population of end-users.

2.3 Incorporating Human Subjectivity to Measure Real-World Adaptability

This dissertation explores how incorporating human subjectivity—through naturally occurring data, diverse annotator judgments, and disagreement patterns—can help build benchmarks that better reflect real-world expectations. By leveraging the subtlety and variability of human responses, we can assess whether models are equipped to handle the richness of everyday language and decision-making, ultimately guiding the development of more adaptive and human-aligned AI systems.

2.3.1 Collecting Annotations for Naturally-occuring Adversarial Benchmarks

Building benchmarks from naturally occurring data distributions offers an alternative to manually constructed or synthetic datasets. These real-world data sources reduce the human effort involved in benchmark creation and help mitigate bias that may arise when curated benchmarks overlook essential task phenomena (Bowman and Dahl, 2021a). For example, many datasets have been collected from the wild: naturally occurring questions (Kwiatkowski et al., 2019), discourse-

based tasks (Paperno et al., 2016), and community-generated content like StackExchange data (Hazoom et al., 2021). Also, there exist fact checking datasets that are crawled from fact checking websites (Popat et al., 2016; Ferreira and Vlachos, 2016; Hanselowski et al., 2019) or misleading headline datasets harvested from online social media platforms (Slovikovskaya, 2019; Potthast et al., 2018; Ferreira and Vlachos, 2016; Ha et al., 2018; Shang et al., 2019). Furthermore, ample benchmarks utilized crowdsourcing as their primary method to collect data (Rajpurkar et al., 2016). While scalable and efficient, this approach introduces several challenges especially in subjective tasks: annotator biases, inconsistent interpretations of task instructions, and low inter-annotator agreement can all compromise data quality and reliability (Bowman et al., 2015; Bartolo et al., 2021b).

One of the main advantages of using natural data distributions is that they more closely mirror the complexity of real-world scenarios, where target skills are often reflective of domain knowledge and contextual interpretation. While natural data may have challenges in class imbalance or limited data availability, they also encourage models to develop more robust and adaptive behaviors when they are deployed. As a result, evaluations based on natural distributions can offer a more realistic measure of a model's performance in practical applications.

2.3.2 Capturing Subjective Human Judgments and Nuance

Interestingly, the challenges that come with subjective tasks like sentiment analysis or hate speech detection can actually serve as an advantage (Sandri et al., 2023). Since annotators often have different perspectives, their disagreements can instead reflect the nuanced and context-dependent nature of these tasks (Kenyon-Dean et al., 2018; Davani et al., 2022). Some even

postulate that the disagreements in subjective tasks are not just inevitable but essential, as they capture important nuances of the target task that are often ignored while aggregating annotations (Davani et al., 2022). This diversity can lead to evaluating models whether they are more adaptable, better at handling ambiguity, understanding multiple valid interpretations, and aligning with the range of human reasoning. From an evaluation standpoint, it allows us to test whether models are not only accurate, but also adaptable and sensitive to nuance (Giorgi et al., 2024; Röttger et al., 2021; Liu et al., 2023a). For example, in misinformation context, misleading headlines that are incongruent with video content may be technically accurate but still misleading to some viewers, depending on how it frames the content it summarizes. In such cases, annotators with different backgrounds may disagree, but their rationales for why something is misleading can reveal critical linguistic or social cues that models must learn to detect (Uma, 2024). Capturing such disagreement patterns can reveal evaluation blind spots, improve benchmark design, and help identify where models fail to generalize.

To systematically analyze annotator disagreement while accounting for reliability, we can apply MACE, Multi-Annotator Competence Estimation (Hovy et al., 2013), a Bayesian method that estimates the trustworthiness of annotators and helps filter out unreliable or spam-like behavior. MACE models the annotation process by assuming that each annotator either provides a correct label with some probability (based on their competence) or chooses a label according to a personal

"spam" distribution. The generative process is formalized as follows:

$$
\begin{aligned}
&T_i \sim \text{Uniform}, \quad \text{for } i = 1, \ldots, N \\
&S_{ij} \sim \text{Bernoulli}(1 - \theta_j), \quad \text{for } j = 1, \ldots, M \\
&A_{ij} = \begin{cases} T_i & \text{if } S_{ij} = 0 \\ \text{sampled from Multinomial}(\xi_j) & \text{if } S_{ij} = 1, \end{cases}
\end{aligned}
$$

where N is the number of items (e.g., headlines), M is the number of annotators, and T_i is the latent true label for item i. Each annotation A_{ij} is determined either by the true label (if the worker is not spamming) or by a multinomial distribution ξ_j reflecting their spam behavior. The spam indicator S_{ij} follows a Bernoulli distribution parameterized by $1 - \theta_j$, where θ_j is the estimated competence of annotator j. Beta and Dirichlet priors are placed on θ and ξ, respectively. This allows retaining the meaningful variation in subjective annotations while discounting contributions from untrustworthy annotators. This preserves the signal embedded in human disagreement without conflating it with noise, offering a more faithful reflection of real-world subjectivity in evaluation tasks.

2.3.3 Subjective and Challenging Task: Misleading Video Headline Detection

Building on the discussion of subjective tasks and the importance of incorporating human perspectives, we now examine a particularly challenging application: detecting misleading video headlines. This task is inherently subjective, context-dependent, and multimodal, making it an ideal case study for a subjective and challenging task for model evaluation.

2.3.3.1 Clickbaits and Misleading Headlines

There is a line of fact-checking research focused on detecting and correcting misleading headlines. This has been one of the major factors of misinformation, especially on social media platforms (Wei and Wan, 2017). In particular, clickbait is characterized by misleading headlines, depending on the degree of deception the audience experiences (Bourgonje et al., 2017). For example, Potthast et al. (2018) and Slovikovskaya (2019) use crowdsourcing method to label each news headline pair on a four-point scale (not, slightly, considerably clickbaiting, heavily clickbaiting) and four stances (Agrees, Disagrees, Discusses, Unrelated) depending on the level of misleadingness. Ferreira and Vlachos (2016) includes a manual analysis of the first 50 body IDs containing headlines that are an existing sentence in the text articles. Previous works propose that clickbaits are characterized by misleading or ambiguous headlines, depending on the degree of deception the audience experiences (Wei and Wan, 2017).

2.3.3.2 Misleading Headline Detection

While often used interchangeably, clickbait and misleading headlines are not synonymous. Transitioning from clickbait detection to the more nuanced task of identifying misleading headlines, prior research shows that a headline can be exaggerating but not misleading (Chen et al., 2015). For example, some viewers may determine a clickbaity headline to be mildly exaggerating but not deceptive, while others may decide that the headline is excessively exaggerating and thereby misleading. For example, clickbait may omit key information to create curiosity (e.g., "You won't believe what happened next!"), while a misleading headline can falsely frame the content to support a specific narrative (e.g., "New Study Proves Vaccines Are Dangerous" when the

content talks about how vaccines are dangerous when a patient is under the influence of alcohol). This comes from differences in their background knowledge, expectations on what is perceived as *misleading*, and individual preferences to framing. Thus, determining whether a headline is misleading is inherently subjective and challenging for annotators *and* AI models.

2.3.3.3 Multimodal Misleading Headline Detection Benchmarks

To advance this area, recent work has expanded beyond textual headlines to examine misleadingness in multimodal contexts, particularly video content. As the spread of fake news appears in many forms of multimedia (Aïmeur et al., 2023), several works are on constructing datasets to support research on multimodal misleading headline detection (Bu et al., 2023). Some studies have identified video thumbnails as heavily influential in detecting whether the video headline is misleading the audience or not. Ha et al. (2018) introduce an image-based dataset and focuses on misrepresented headlines on Instagram. Also, Shang et al. (2019) present a dataset of Youtube videos with manual annotations generated by misleading seed videos from the Youtube recommendation system. Zannettou et al. (2018) propose manual and automatic misleading-labeling mechanisms. The annotated videos could be biased as manual and automatic annotation may not be in consensus; they can lead to erroneous annotations of misleading headlines.

In Chapter 4, we incorporate human annotations by structurally examining subjectivity and analyzing how individual judgments—such as perceived misleadingness or contextual relevance—vary across annotators through their provided rationales, enabling a more interpretable evaluation of model robustness.

2.4 Embedding User Values and Perceptions for Human-AI Alignment

We now turn to the broader challenge of aligning AI evaluation frameworks with user values and interpretability concerns. In particular, we explore how human-grounded evaluation can embed human expectations throughout the system pipeline to improve reliability and real-world applicability.

2.4.1 Aligning Model Predictions with User Standards

One of human-grounded evaluation methods offers powerful mechanism to integrate user input across all stages of development and evaluation of AI systems (Wang et al., 2021b; Akoury et al., 2020). Rather than limiting user involvement to final scoring, this approach includes leveraging user judgment in task design, prompt formulation, model auditing, and feedback loops (Chang et al., 2024). Such integration recognizes that user values, context awareness, and reasoning capabilities are crucial for identifying a model's blind spots and failure modes. With the feedback loop where users can continuously refine the system, the pipeline can adhere to end user expectations and offer guidelines for making correct revisions of erroneous outputs (Wallace et al., 2019; Cao and Wang, 2021). This iterative loop transforms evaluation from a static, one-time assessment into an adaptive, user-centered process (Wallace et al., 2019; Cao and Wang, 2021).

2.4.2 Direct User Engagement in Multi-agent System Evaluation

Building on the principles of user-grounded evaluation, we apply this principle to the challenges of evaluating multi-agent systems powered by LLMs. LLM-as-agent systems are increasingly being used for complex real-world tasks where individual agents perform subtasks

via decomposition, chaining, or planning, mirroring human problem-solving strategies (Suzgun et al., 2023; Srivastava et al., 2023; Wu et al., 2022a; Lou et al., 2024).

However, these agent-based systems often fall short, often contradicting human cognition and miss critical errors that escape automated detection (LangChain, 2013; Arawjo et al., 2024). For example, an incorrect result by a single agent—misinterpreting a calculation or nonfactual claim generation—may be unflagged by LLM-only evaluations but would likely be identified and corrected by human reviewer (Wu et al., 2022b; Grunde-McLaughlin et al., 2025b; Lee et al., 2022b). This underscores the need for human oversight, particularly in high-stakes applications such as legal or healthcare settings, where the consequences of error are significant. This places a substantial burden on users, who must inspect the reasoning and output of each agent, creating an evaluative challenge: how can users reliably assess complex trajectories of reasoning without formal tools or structured guidance?

Given these limitations, the research focus has increasingly shifted toward understanding how evaluation results should be interpreted and by whom. This shift reflects the growing consensus that AI evaluation must be grounded in user needs and shaped by end-user expectations. For instance, Liao and Xiao (2023) argue that evaluation modules must provide valid assessments of whether and to what extent human needs are met in downstream use cases, ensuring human realism. Similarly, Arabzadeh et al. (2024) emphasizes that identifying and quantifying the criteria for LLM-powered applications is essential to verify whether they satisfy user requirements and ultimately bring utility to end-users.

To operationalize this, several recent works propose integrating user-defined evaluation rubrics directly into the system (Ibrahim et al., 2024). Kim et al. (2024) demonstrate that incorporating user-defined criteria enhances the alignment between LLM evaluator outputs and

human-annotated outputs. Similarly, Shankar et al. (2024) propose aligning LLM evaluators with user-specific goals and criteria, enabling tailored evaluation pipelines that reflect the requirements of downstream tasks. In Chapter 6, these approaches are revisited to reinforce the central message of this dissertation: evaluation should not be limited to correctness or surface-level metrics, but must also reflect human values, expectations, and context. This is especially critical in multi-agent systems, where embedding user input into the evaluation loop is essential for developing models that are robust, trustworthy, and truly aligned with the needs of the people they serve.

The next four chapters of this dissertation build on the foundational concepts to pursue two central objectives: constructing challenging, human-grounded benchmarks and advancing trustworthy, human-aligned AI systems. The first two chapters present dataset generation pipelines and evaluation methodologies designed to surface model vulnerabilities by leveraging naturally adversarial examples, subjective human annotations, and expert-informed judgments. These benchmarks aim to better capture the linguistic nuance and diversity of real-world user interactions, offering a more faithful alternative to traditional evaluation datasets. The latter two chapters turn toward the goal of developing trustworthy AI, introducing approaches that incorporate human feedback, promote interpretability, and embed user values within agentic system design. Together, these chapters demonstrate how integrating human-centered signals throughout the evaluation and development process can lead to more robust, transparent, and user-aligned NLP systems.

Chapter 3: Is your benchmark truly adversarial? AdvScore: Evaluating Human-Grounded Adversarialness[1]

This chapter introduces a HITL approach for constructing adversarial evaluation benchmarks through expert-authored examples and metrics by collecting both human and model responses. This HITL metric measures adversarialness of a benchmark example—defined as examples that are easy for humans but hard for models—using item response theory (IRT) (Lalor et al., 2016) to account for human skills, and use it to ground the metric score in difficulty gap between humans and models, response ambiguity, and example skill distinguishing ability. To create benchmark examples, we present an HITL interface that enables experts to iteratively probe model behavior in real time, refining prompts to expose weaknesses while ensuring practical plausibility.

3.1 Motivation

As language models attain near-perfect performance on existing benchmarks, there is an increasing demand for unexpected and challenging tasks to evaluate them. *Adversarial datasets* contain examples that cause models to generate harmful (Perez et al., 2022), unsafe (Quaye et al., 2024), or incorrect (Goodfellow et al., 2015) responses. An ideal adversarial example should be

[1]Yoo Yeon Sung, Maharshi Gor, Eve Fleisig, Ishani Mondal, and Jordan Lee Boyd-Graber. 2025. Is your benchmark truly adversarial? AdvScore: Evaluating Human-Grounded Adversarialness. In *Proceedings of the Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics*

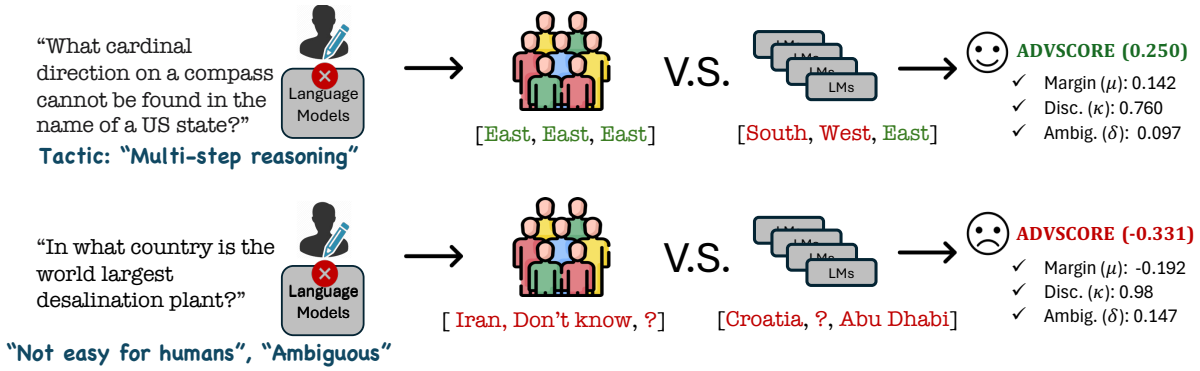

Figure 3.1: ADVSCORE diagnoses when a question is adversarial (top) and difficult for computers to answer for other reasons (bottom). After collecting candidate questions, we ask humans and computers to answer the questions. The top question (from ADVQA) has a higher ADVSCORE because it is specific, adversarial, discriminative, high-quality, and realistic. In contrast, the bottom question is ambiguous (e.g., none of humans or models correctly answered due to its ambiguity), which is confirmed by its low ADVSCORE.

much easier for a human to answer correctly than for a model on realistic tasks (Ilyas et al., 2019; Tsipras et al., 2019; Engstrom et al., 2020; Biggio et al., 2012). However, as models improve, these adversarial datasets can become outdated (Kiela et al., 2021)—what was hard for a model in 2020 can become trivial in five years—requiring periodic updates (Recht et al., 2019; Bowman and Dahl, 2021a). On the other hand, it is difficult to recognize at what point have these adversarial datasets outlived their usefulness systematically, nor is there an established metric to measure which datasets best captures the gap between human and model ability.

To fill this gap, we formulate **ADVSCORE** (§ 3.2). This metric measures two critical aspects: **(i) adversarialness**, which captures the performance gap between models and humans, while penalizing "ill-posed" examples (i.e., ambiguity), and **(ii) discriminability**—how effectively can a dataset rank models by their abilities.

Measuring whether a dataset is truly adversarial requires human answers; thus, ADVSCORE builds on item response theory (Lalor et al., 2016, IRT), a framework widely used in psychometrics and educational testing. It captures the diversity of human and model abilities and identifies poor

examples (§ 2.2.2.1). ADVSCORE is the first metric that evaluates an example's "adversarialness" grounded in human abilities: it can measure whether the dataset's adversarial challenge becomes weaker or stronger as language models improve. We apply ADVSCORE to motivate authors to contribute to a new human-in-the-loop HITL benchmark of adversarial questions, ADVQA. ADVQA's creation pipeline (Figure 3.1) produces *high-quality* and *realistic* questions that are adversarial. Moreover, ADVSCORE helps make ADVQA discriminative, ensuring that the captured adversarialness reflects the varying skills of humans and models.

ADVQA exhibits the least decline in adversarialness over recent years compared to other adversarial benchmarks (§ 3.3). This minimal, but meaningful decline in ADVQA reveals that current models (e.g., GPT4) continue to struggle with tasks requiring *commonsense reasoning* and *multistep reasoning* and on topics such as *Lifestyle* (§ 5.4), which are likely tied to real-world challenges.

We conclude with an analysis of how model have improved improve over the years since researchers began releasing adversarial datasets and how that can inform the development of future adversarial datasets (§ 3.3).

3.2 ADVSCORE

This section introduces ADVSCORE, a metric that evaluates how *adversarial* and *discriminative* a dataset is. We measure these two key criteria: **(i) adversarialness**, how much more challenging a question is for AI models compared to humans while being well-posed; and **(ii) discriminability**, how informative is the question in effectively distinguishing between different skill levels.

3.2.1 Quantifying Adversarialness

A question is adversarial if *skilled* humans consistently answer a question correctly but computers do not. We measure this gap by fitting IRT parameters and then computing the probabilities predicted by the trained 2PL-IRT model (§ 2.2.2.1). During margin computation, we conduct synthetic groups for both human and computer subjects with representative skill levels. Then, we compute the probability of each group correctly answering the question, as estimated by the IRT model, which accounts for question quality. A question is considered adversarial if the human representative has a higher probability of answering correctly than the computer representative.

Skilled Groups. We first define what constitutes a *skilled* group g, and further define its *representative skill* β_*^g, which we use in subsequent equations (3.2,3.4). For a set of randomly sampled subjects S, skilled group $S_{(k)}$ is the subset of subjects with skill at least k standard deviations above the mean— $\beta_i > \mu_\beta^S + k\tau_\beta^S$—where μ_β^S and τ_β^S are the mean and standard deviation of subject skills over the set S, and k indicates the degree of expertise. We define the *representative skill* β_*^g for the chosen group g as the expected skill level of the subjects within that group:

$$\beta_*^g = \mathop{\mathbb{E}}_{\beta_i \sim g}[\beta_i]. \tag{3.1}$$

Margin Computation. For question j in a dataset D, the performance-margin μ_j is the difference between the probabilities of *skilled* humans $H_{(0)}$ and *skilled* models $M_{(0)}$ correctly answering the question, using their respective representative skills $\beta^{H_{(0)}}$ and $\beta^{M_{(0)}}$. We set $k = 0$ and designate *skilled* humans ($H_{(0)}$) and models ($M_{(0)}$) as the skilled subsets of subjects. These subjects have

skills above the average level of their respective subject pools:

$$\mu_j = \underbrace{\sigma_{\text{2pl}}\left(\beta_*^{H_{(0)}}, \theta_j, \gamma_j\right)}_{\text{Skilled human rep. prob.}} - \underbrace{\sigma_{\text{2pl}}\left(\beta_*^{M_{(0)}}, \theta_j, \gamma_j\right)}_{\text{Skilled model rep. prob.}}, \tag{3.2}$$

where $\sigma_{\text{2pl}}(\beta, \theta, \gamma)$ is the logistic function for our 2PL-IRT (Eq. 2.1, § 2.2.2.1), that uses β_*^g as the representative skill for subject group $g \in \{H_{(0)}, M_{(0)}\}$, and θ_j and γ_j are the difficulty and discriminability parameters of the question j.

A positive value for the margin μ_j implies that the question j is *adversarial* (examples in 3.3), while a negative value implies the opposite, and the magnitude indicates the extent of adversarialness.

Accounting for Question Ambiguity. While the margin (μ_j) captures the core of adversarialness, it does not ensure if the questions are genuinely well-posed; ambiguous, or poorly formulated questions could inflate this score without being *truly* adversarial. To address this issue, we introduce a discount term (Eq. 3.3) that relies on the disagreement level among *highly-skilled* (or expert) human subjects ($H_{(1)}$) for each question:

$$\mu'_j = \frac{\mu_j}{1 + \delta_j}, \tag{3.3}$$

where μ'_j is the adjusted adversarialness score, μ_j is the original adversarialness score, and δ_j is a measure of disagreement among highly skilled human subjects $H_{(1)}$ for question j.[2] To keep this measure of disagreement standardized, δ_j is the mean deviation (MD) of the probabilities of $H_{(1)}$

[2]We use this approach for crowdsourced human subjects. For manually identified expert human subjects, we directly use their responses without the need for skill-based filtering.

answering question j correctly:

$$\delta_j = \underset{i \sim H_{(1)}}{\text{MD}} \left[\sigma_{\text{2pl}} \left(\beta_i^{H_{(1)}}, \theta_j, \gamma_j \right) \right]. \tag{3.4}$$

This discount term ensures that questions with high disagreement among expert humans (potentially ambiguous or ill-posed questions) are penalized, even if they show large human-model performance gaps. This approach leverages the value of human judgment for *true* adversarial quality assessment.

3.2.2 Measuring Discriminability

The best questions distinguish between subjects' varying skill levels—they are *informative* and showcase high *discriminability*. We measure this by leveraging Fisher information over our 2PL-IRT's response prediction function, also called Item Information Function (Lord et al., 1968, IIF); it is a function that measures an item's contribution to the measurement precision of $P(\theta)$ across the skill range (θ). With $P(\theta)$ as the 2PL-IRT's response prediction function $\sigma_{\text{2pl}}(\beta, \theta, \gamma)$, we get the item information function ($\text{IIF}_j(\theta)$) that quantifies how much statistical information a question j provides about a subject's skill level θ:

$$\text{IIF}_j(\theta) = \gamma_j^2 \cdot p_j(\theta) \cdot (1 - p_j(\theta)), \text{ where} \tag{3.5}$$

$$p_j(\theta) = \sigma_{\text{2pl}}(\theta, \theta_j, \gamma_j). \tag{3.6}$$

Here, the questions with **high discrimination** (large γ_j^2) and moderate difficulty (resulting in $P(r_{ij}) \approx 0.5$) provide the most information.

Finally, we define the total item information (TIF_j) provided by question j as the area under the $\text{IIF}_j(\theta)$ curve, and scale it by exponential normalization to obtain a standardized, calibrated measure of discriminability κ_j for question j:

$$\text{TIF}_j = \int_{-\infty}^{\infty} \text{IIF}_j(\theta) d\theta, \tag{3.7}$$

$$\kappa_j = 1 - \exp\left(-\text{TIF}_j\right). \tag{3.8}$$

3.2.3 Combining into AdvScore

To recap, an ideal adversarial question should (i) have a high margin of human and model performance gap, while being well-posed (low expert-humans disagreement), and (ii) be discriminative (informative of the subject's skill). Thus, first combine the adversarialness (μ'_j) and discriminability (κ_j) to get a single metric:

$$\text{AdvScore}_\text{j} = \frac{\mu_j}{1 + \delta_j} \cdot (1 + \kappa_j) \tag{3.9}$$

To have human–model probability margin (μ_j) as a key factor in AdvScore, we treat κ_j as a multiplicative bonus to μ_j. This prevents questions with high discriminability (κ_j) from contributing to AdvScore if their μ_j values are low.

A positive AdvScore indicates a truly adversarial dataset, with higher values suggesting more discriminative and adversarial questions. We use AdvScore to evaluate existing datasets (§ 3.3) and to reward authors in our AdvQA dataset creation process (§ 3.4.1). We define the AdvScore of a dataset D as the average AdvScore of its questions. An effective

adversarial dataset should contain numerous questions with high ADVSCORE.

3.3 Adversarial Benchmark Evaluation

We compare adversarial benchmarks across different domains using ADVSCORE. Our evaluation includes ADVQA, a new QA dataset developed through a HITL process to align adversarial data with human capabilities. This section, analyzes ADVSCORE as a metric, while § 3.4 details the creation of ADVQA, and § 5.4 examines what makes ADVQA questions adversarial.

Adversarial datasets with human responses. For ADVQA, we gathered human responses through a live, in-person QA competition involving 8 human teams, as well as through online crowdsourcing with 165 participants. In total, we collected 1,839 human responses from 172 individuals. To compare the adversarialness of these datasets using ADVSCORE, which relies on both human and model response data, we are limited to comparing ADVSCORE with datasets with human annotations. Thus, we select TRICKME (Wallace et al., 2019) and FM2 (Eisenschlos et al., 2021a). While TRICKME challenges models with QA pairs, FM2 uses entailment pairs for fact-checking.[3] Additionally, we included BAMBOOGLE (Press et al., 2022), which consists of general knowledge questions designed to be adversarial, similar to ADVQA. As BAMBOOGLE lacked human responses, we gathered 10,391 responses from 165 crowdworkers.

We also collected model responses for each dataset from ten models, including Dense Passage Retrieval (Karpukhin et al., 2020, DPR), GPT-3-INSTRUCT (Ouyang et al., 2022), GPT-3.5-TURBO (OpenAI, 2023), MISTRAL-V0.1-INSTRUCT (Jiang et al., 2023), GPT-4 (Achiam et al., 2023), LLAMA-2-CHAT models in sizes of 7b and 70b, and LLAMA-3-INSTRUCT models in

[3]We use human responses from Si et al. (2023)

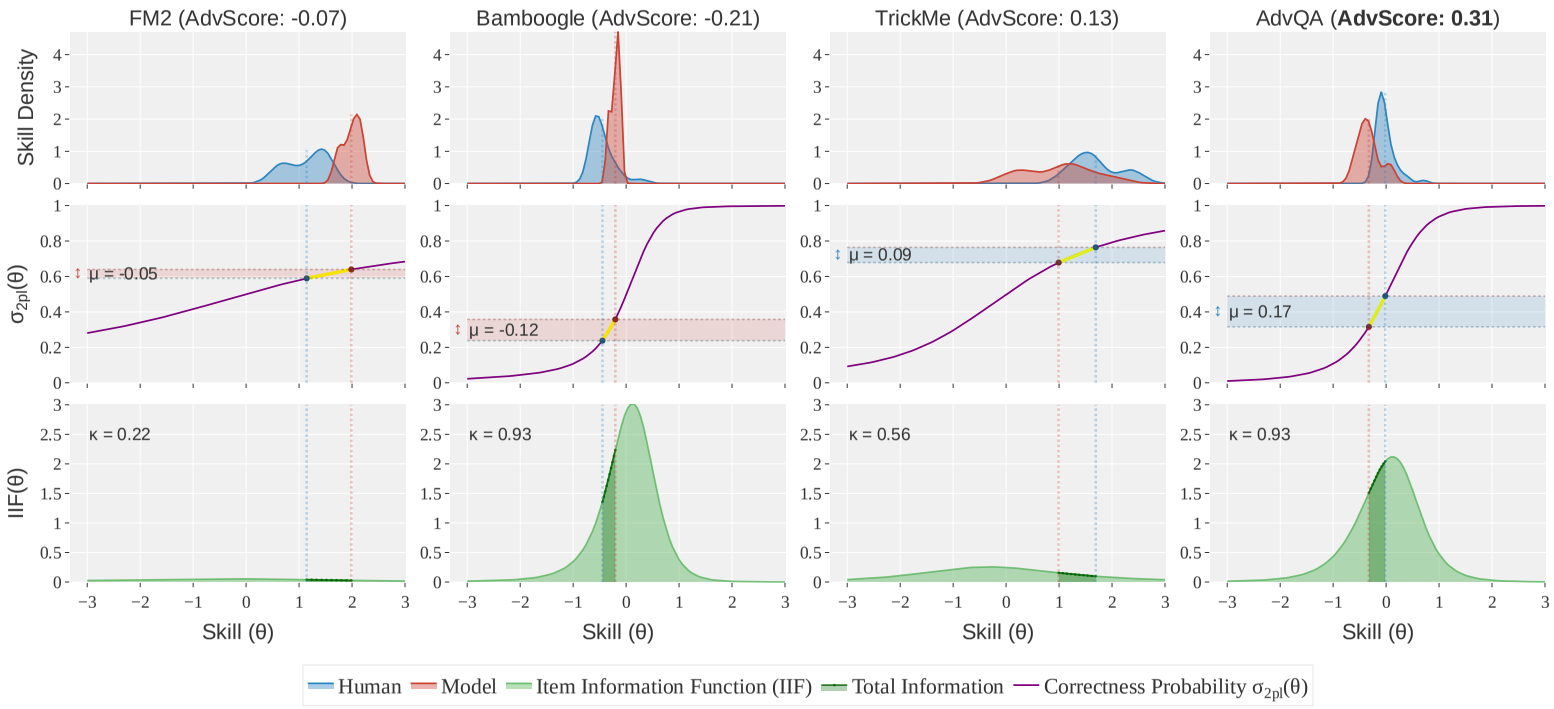

Figure 3.2: **Visualization of key ADVSCORE components across datasets.** For each dataset, we plot: (1) Skill density of skilled humans ($H_{(0)}$) and skilled models ($M_{(0)}$), (2) response correctness probability, $\sigma_{2\text{pl}}(\theta)$ (Eq. 2.1, § 2.2.2.1) averaged over dataset examples, and (3) Item information function (IIF(θ)(Eq. 3.5, § 3.2.2). Vertical dashed lines show representative (average) skill levels for humans and models. The gap between human and model probabilities (shaded region between the horizontal lines) indicates adversarialness (μ_D). IIF peaks show where questions are most informative, with area under curve signaling total informativeness (discriminability, κ_D). **Key insights:** BAMBOOGLE has high informativeness but favors models (negative μ_D). TRICKME separates humans and models but has lower discriminability (positive μ_D). ADVQA is the best of all, effectively discriminating between humans and models while maintaining high informativeness throughout, resulting in the highest ADVSCORE of 0.31.

sizes of 8b and 70b (Touvron et al., 2023). After collecting human and model responses, we apply 2PL-IRT to extract the learned subject and item parameters and compute ADVSCORE.

Comparison of adversarial benchmarks. We compute ADVSCORE$_D$ and its components (μ_D, κ_D, and δ_D) for each dataset, presenting results in Table 3.1. Figure 3.2 walks through the computation of ADVSCORE by illustrating (i) the skill density of *skilled* humans $H_{(0)}$ (blue) and models $M_{(0)}$ (red), (ii) the response correctness probability ($\sigma_{2\text{pl}}$, purple), and (iii) the *item information function*, IIF (green, E.q. 3.5), over skill θ.

Both ADVQA and TRICKME show a clear separation between human and model skill levels (first row), resulting in positive, high margins (μ) of 0.17 and 0.13, correspondingly (yellow in second row). However, ADVQA has a higher overlap of IIF with regions where human skill exceeds model skill (dark green area in third row), compared to TRICKME, which has a flatter and less informative IIF. These lead to lower κ_D (0.56 vs 0.93), suggesting that TRICKME questions are less discriminative (less useful in assessing subject skills).

In contrast, BAMBOOGLE has an informative IIF, but the skill of the model tends to exceed humans, resulting in a negative μ_D (Table 3.1). This suggests that BAMBOOGLE questions are inversely adversarial, containing questions where models outperform humans, and therefore fail to serve as an effective adversarial benchmark. Similarly, FM2 has a negative μ_D and low κ_D, indicating that the dataset is neither adversarial nor discriminative. Our analysis establishes ADVQA questions as most adversarial, as indicated by its highest ADVSCORE$_D$ of 0.31; thus demonstrating that the unique components of ADVSCORE effectively support the evaluation of adversarial benchmarks.

Chronological evaluation of adversarialness. Adversarial datasets inevitably become obsolete

Datasets (D)	μ_D	κ_D	δ_D	**ADVSCORE**$_D$
ADVQA	**0.17**	**0.93**	0.08	**0.31**
FM2	-0.05	0.22	0.01	-0.07
BAMBOOGLE	-0.12	0.93	**0.11**	-0.21
TRICKME	0.09	0.56	0.03	0.13

Table 3.1: ADVQA had the highest ADVSCORE$_D$, along with the highest μ_D and κ_D, indicating that its questions were the most adversarial and best at discriminating subject's skill across the four datasets. While BAMBOOGLE has the same κ_D value, the negative μ_D indicates the reverse adversarialness, suggesting it was distinctively easier for *models* than humans.

as models improve, either by training on these datasets or overcoming previously identified vulnerabilities. Using ADVSCORE, we assess model improvements over the last five years by identifying which datasets have become less adversarial, incorporating new models into the ADVSCORE computation.[4] Figure 3.3 shows the ADVSCORE for each dataset over the years, confirming that ADVQA holds the highest ADVSCORE (2024) with the smallest decline over the last five years. In contrast, TRICKME, which was initially the most highly adversarial (2020), saw a sharp decline over the following four years, indicating that the models improved on the tasks that they previously struggled with. BAMBOOGLE and FM2 are no longer adversarial, showing negative ADVSCORE values since 2022. BAMBOOGLE's reliance on a 2-hop tactic and simple questions (e.g., *"What is the capital of the second largest state in the US by area"*) likely explains its decline since 2021. FM2's drop suggests LLMs have improved at fact-checking or benefitted from similar questions in training. Although pinpointing the exact factors behind model improvement may be challenging, it is crucial to determine whether these models have become more resilient or remain vulnerable as new models emerge. ADVSCORE facilitates this by quantifying how much a dataset has lost its adversarialness, offering a concrete measure of how well the model withstands

[4]Models introduced by year: DPR in 2020, GPT-3-Instruct in 2021, GPT-3.5-TURBO in 2022, Mistral-0.1-instruct, GPT-4, Llama-2-7b-chat, and Llama-2-70b-chat in 2023, and Llama-2-7b-chat, Llama-2-70b-chat, Llama-3-8b-instruct, Llama-3-70b-instruct, and rag-command-r-plus in 2024.

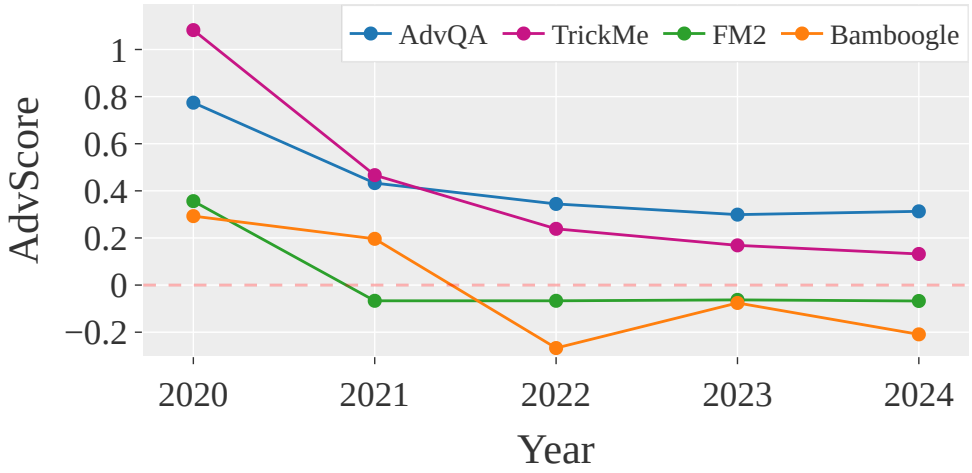

Figure 3.3: We report ADVSCORE for each dataset over the years, confirming that ADVQA holds the highest ADVSCORE with the smallest decline over the last five years, proving its adversarial robustness.

adversarial challenges over time.

Qualitative Examples with ADVSCORE. We examine the human-model margin probability (μ_j) and each subject's answers to the example question for each dataset. In Table 3.3, ADVQA and TRICKME questions show a positive μ_j value, indicating adversarial, correspondent to the human's correct answer to ("Putin") and GPT4's wrong answer ("Russia"). On the other hand, BAMBOOGLE and FM2's negative adversarialness value suggests that the question is easier for models compared to humans, as reflected in the higher correctness from models versus humans.

Comparision of ADVSCORE and QSR. Moreover, we compared the the model and human success rates (QSR) and ADVSCOREs.

While QSR may suggest that humans outperform models, the questions can consistently yield negative ADVSCOREs, due to their low or negative μ (margin) or high δ (ambiguity). This highlights that QSR alone is insufficient to determine question adversarialness, whereas each parameter in ADVSCORE offers a more reliable measure.

For questions like *What was the founding date of the university in which Plutonium was discovered?* and *Who is the father of the father of observational astronomy?*, humans significantly outperform models, but their negative ADVSCORES (-0.365 and -0.340) indicate that these questions remain non-adversarial. This demonstrates that QSR alone is insufficient to identify question adversarialness. ADVSCORE, by incorporating both margin and discriminative power, provides a more nuanced and reliable measure, and reflects the adversarial nature of questions.

3.4 ADVQA Creation Pipeline

We showed that ADVQA is more adversarial and discriminative than other datasets, suggesting its creation process contributed to these qualities. Here, we discuss the ADVQA collection process as a case study to guide future high-quality adversarial datasets.

3.4.1 Collecting Examples through Adversarial Competitions

To obtain human-written question-answer pairs, we hold two adversarial model–human QA competitions. First, in the writing competition, we collect 399 adversarial questions through the interface (§3.4.2), which are then edited and filtered by an expert editor. Second, in the answering competition, we invited eight expert human groups (composed of three to four trivia experts) to run eight human vs. model QA tournaments to obtain 780 human responses. Each tournament initially consisted of 30 questions, which are then filtered based on experts' comments (E.g., *"This question is ill-posed"*). After this filtering process, ADVQA results in 182 questions.[5] After the

[5]Larger than other IRT-analysed test sets (e.g., 139 for RTE, 20 for COMMITMENTBANK, 50 for COPA) (Vania et al., 2021). Also, additional 1,839 human responses collected from 172 individuals (165 crowdsource workers). Dataset value includes both questions and response volume.

AdvQA Dataset							
Question	**Answer**	**Human QSR**	**Model QSR**	μ_j	δ_j	κ_j	**AdvScore**$_j$
Name the color of the sky in Aivazovsky's "The Ninth Wave"	Orange	0.667	0.083	0.583	0.106	0.963	0.188
The title of this book shares a word with the title of a song of which the author, who acted in the 2002 film, 8 Mile, addressed to his daughter and niece	To Kill a Mockingbird	0.333	0.000	0.323	0.102	0.983	0.179
What country shares a language with its more populous northern neighbor but in its written form omits a letter that looks like a Greek beta, writing the sound instead by doubling another letter? That character appears in that language's words for foot, big, outside, and street	Switzerland	0.333	0.000	0.333	0.051	0.626	0.081
A German admiral sailing for Russia named what islands for an English captain and not for the librettist of the HMS Pinafore nor for the announcer of Jeopardy!	Gilbert Islands	0.333	0.100	0.233	0.034	0.504	0.051
Bamboogle Dataset							
Question	**Answer**	**Human QSR**	**Model QSR**	μ_j	δ_j	κ_j	**AdvScore**$_j$
What was the founding date of the university in which Plutonium was discovered?	March 23, 1868	0.452	0.167	0.285	0.127	0.972	-0.365
Who was the father of the father of psychoanalysis?	Jacob Freud	0.528	0.500	0.028	0.149	0.982	-0.354
When did the person who gave the Checkers speech die?	April 22, 1994	0.200	0.167	0.033	0.156	0.985	-0.350
Who is the father of the father of observational astronomy?	Vincenzo Galilei	0.324	0.167	0.157	0.121	0.964	-0.340
What is the third letter of the top-level domain of the military?	l (lower case L)	0.516	0.333	0.183	0.152	0.983	-0.338

Table 3.2: A substantial gap in QSR may suggest human superiority over models, indicating an adversarial question. However, it can still yield negative AdvScores due to low or negative μ or relatively high δ. In both AdvQA and Bamboogle, even when human QSR surpasses model QSR, this is not always reflected in AdvScore, given the distinct criteria of each parameter. For instance, the first question in AdvQA, *Name the color of the sky in Aivazovsky's "The Ninth Wave"* exhibits a significant QSR gap between humans (0.667) and models (0.083), yet its positive AdvScore$_j = 0.188$ remains low, due to high δ (indicating question ambiguity) compared to other examples. The question implies a single color, but the "The Ninth Wave" painting contains multiple hues. It also lacks specificity about which part of the sky is being referenced.

competitions, we incentivize ttop writers by AdvScore and top players by skill.[6]

[6] AdvScore is not computed *during* the dataset construction. It is a post-hoc evaluation metric.

Dataset	Question	Answer	Margin (μ_j)	Human Response	GPT-4
ADVQA	Who is the president of the country represented by the second letter in the acronym BRICS [...]	Vladimir Putin	0.19	Putin	Russia
FM2	Aram Khachaturian had Russian roots.	False	-0.01	"False"	True
TRICKME	In a novel by this author, a detective wraps his arm to survive a dog attack [...]	Durrenmatt	0.12	"Durrenmatt"	Franz Kafka
BAMBOOGLE	Who directed the highest grossing film?	James Cameroon	-0.02	"No idea"	James Cameron

Table 3.3: ADVQA demonstrates the most balanced properties of challenging the model and distinguishing between skills, as indicated by a positive μ_j value, which aligns with humans outperforming the models.

3.4.2 Skilled Writers use Adversarial Interface

We provide an adversarial writing interface as a human-AI collaborative tool for the adversarial writing competition, motivated by You and Lowd (2022)'s finding that human-AI collaboration strengthens adversarial attacks. We supply the writers with real-time model interpretations, inspired by Wallace et al. (2019); they could continuously counteract the model response and make edits.

Eliciting incorrect model predictions. The center of the interface (Figure 3.4) provides the Wikipedia page for the target answer, which they use to write the question. While the author is writing, the retrieval widget and QA models widgets are updated (Eisenschlos et al., 2021a). Motivated by Feng et al. (2018), we embed the input perturbation inside the question writing widget to highlight which words trigger the model predictions. For example, changing "company" to a different token would be most likely to change the prediction except the answer "Apple."

Retrieval systems. Users receive real-time feedback on QA systems' performance on their questions via the interface's fine-tuned retrieval and reader model components (the retrieval system outputs: contexts that elicit QA system predictions). If the target answer appears at the top of the

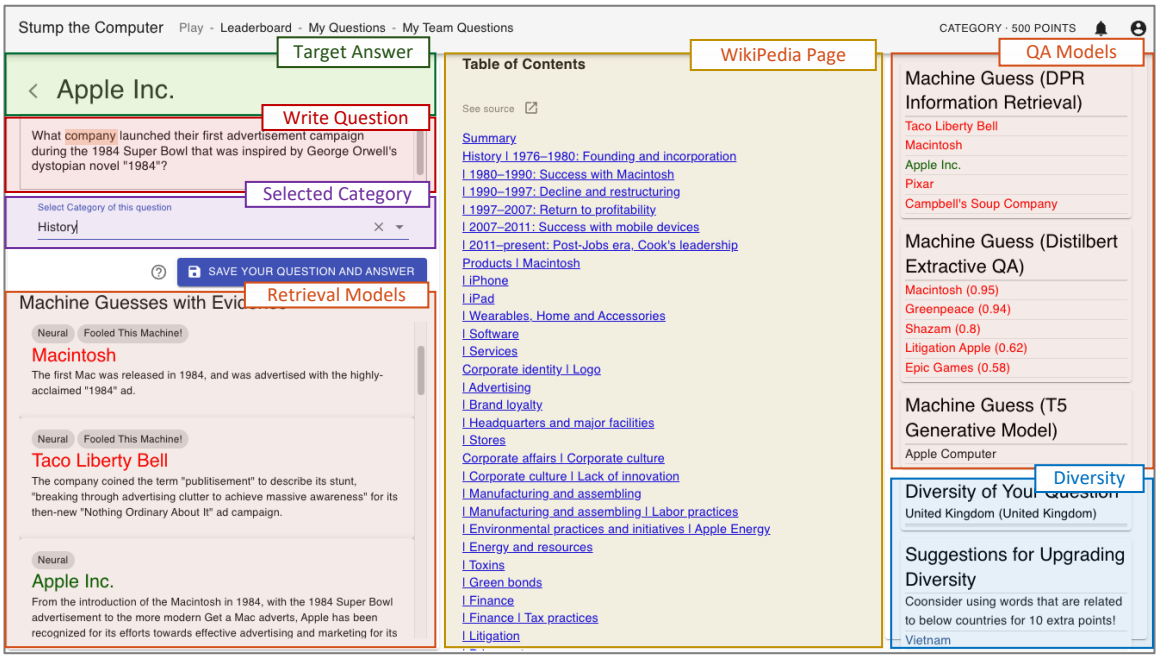

Figure 3.4: As the target answer to the question should be "Apple Inc," the interface is updated with answers from retrieval models with the most relevant sentence and from LMs (e.g., Distilbert, T5). Also, the highlights are updated by the input perturbation technique.

retrieval widget, which means the author failed to fool the retriever and the reader, authors can rephrase questions to avoid retrieving information that makes QA systems answer correctly. We use lightweight sparse and neural retrieval models for writer feedback: a TF-IDF baseline and DPR. To ensure that DPR predictions are diverse and up-to-date, we create a database that indexes each sentence in a set of Wikipedia pages (see § 3.4.2). We then use the RoBerta-based FarmReader, which is fine-tuned on SQuAD (Rajpurkar et al., 2016), to read and sort the retrieved sentences from the two retrieval models by their relevance.

LM-based QA systems. We enrich the model guidance using extractive and generative model answer predictions. For extractive QA, we use DistilBert (fine-tuned on SQuAD), since its promptness and lightness facilitate rapid human-AI interaction. We also use T5[7] (Raffel et al., 2020) to

[7]The writing competition was held in Spring 2023, when DistilBert and T5 were considered comparatively strong.

answer the questions in a closed-book setting.

3.5 Discussion and Analysis on ADVQA

In this section, we show how ADVSCORE can help identify factors that encourage high-quality adversarial datasets. Effective strategies in ADVQA may guide the creation of more adversarial questions, and we analyze how the dataset's realistic aspect can help incorporate human variability during model evaluation.

Ensuring high-quality adversarial questions. The questions should be adversarial for reasons that identify model weaknesses, such as the inability to compose clues or exclude redundant clues (Min et al., 2020, 2022) not because of trivial errors (e.g., grammar mistakes). If the question meets this criteria, we consider it high-quality. We base our criteria on the taxonomy of adversarial categories in Wallace et al. (2019). To understand what yielded ADVQA's *high-quality* adversarial questions, manually annotate the adversarial tactics and topics for ADVQA questions (Table 3.4). With the identified question characteristics, we run a logistic regression model to learn how much each adversarial tactic or topic contributed to ADVSCORE.[8] Since all questions in ADVQA yielded a positive ADVSCORE, the coefficients in Figure 3.5 reflect how much specific features contributed to adversarialness, highlighting areas where models need improvement. For instance, the tactic involving *commonsense knowledge* on the topic of *lifestyle* exposed a model weakness (e.g., *"Take away four from a group including Barnard and Smith, and you get what play?"*), which had a notably high ADVSCORE of 0.27.[9]

[8]Focusing on assessing adversarialness through IRT, we provide only a basic analysis using pre-assigned features. Applying advanced IRT models is encouraged for a richer analysis of adversarial factors (Gor et al., 2024).

[9]The low number of *TV & Film* questions, likely tied to recent news, confirms that ADVQA focuses on probing model capabilities rather than time-sensitive knowledge (Table 3.4).

Adversarial Type	Adversarial Tactics
Composing seen clues	Contains clues that need to be integrated for the question to be answered
Logic and Calculation	Requires mathematical or logical operators
Multi-Step Reasoning	Requires multiple reasoning steps between entities. For eg: "A building dedicated to this man was the site of the "I Have A Dream" speech." A reasoning step is required to infer : "I have a dream" speech to Lincoln Memorial to Abraham Lincoln
Negation	Contains "not" or "non-" and "no" or any negation entities that may confuse the model to answer
Temporal Misalignment	Contains a specific year, month, or timely event that the model is confused about or does not know.
Location Misalignment	Contains a location that the model is confused about or does not know.
Commonsense Knowledge	Requires information that cannot be answered without commonsense
Domain Expert Knowledge	Requires information that cannot be answered without domain expert knowledge
Novel Clues	Contains information that is in the question but is not required to answer. These confuse the models.
Crosslingual	Contains multilingual aspects that confuse the model.

Table 3.4: We list adversarial tactics to determine how each question is using them to stump the models. The annotators are given the description and examples to better understand the reasons why the models may have been stumped. They are expected to tag the examples with the model prediction and question.

Leveraging human feedback for *realisticness*. Realism is crucial for an adversarial dataset as it creates challenges that closely resemble real-world scenarios, effectively testing model robustness against plausible but diverse situations. This approach enhances the reliability of performance evaluation as it reflects high variance in collective human ability. For example, not only should the questions be adversarial, but they should mimic diverse reasoning and problem-solving strategies

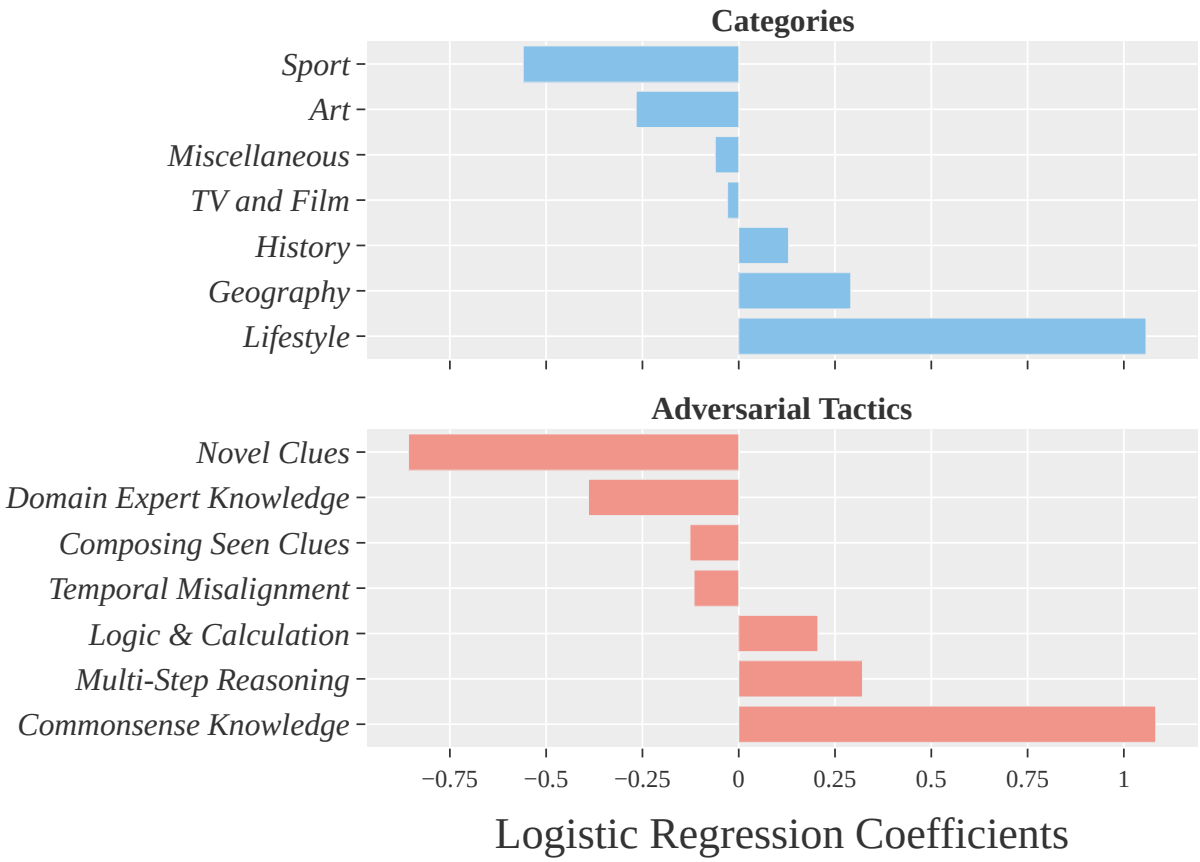

Figure 3.5: The overall distribution of LR coefficients suggests that *lifestyle* and *commonsense knowledge* contribute more to adversarialness than other features. This implies that models still struggle with commonsense knowledge, highlighting an area where they remain vulnerable compared to human understanding.

of different people. Our preliminary results revealed that crowdworkers often produced ambiguous or poorly-formed questions.[10] Although ADVSCORE could identify these issues, many examples were ineffective for assessing model performance. We thus recruit expert trivia writers and guide them in writing adversarial questions. Then, other trivia editors scrutinize the human-authored questions' poor quality (see Table 3.5).

Finally, our human vs. model competition provides an additional quality check, as human subjects flag potential issues while answering questions. If the subject or the editor considers a

[10]E.g., *"Who led the final siege of Constantinople?"* carries ambiguity depending on historical framing (*Mehmet II for the 1453 siege* or *other leaders in prior sieges*).

Adversarial Tactics		Topic Categories	
Features	**Count**	**Topic Category**	**Count**
Commonsense Knowledge	8	Art	7
Composing Seen Clues	57	Geography	17
Crosslingual	2	History	33
Domain Expert Knowledge	10	Lifestyle	11
Location Misalignment	10	Literature	19
Logic & Calculation	14	Miscellaneous	31
Multi-Step Reasoning	50	Music	13
Negation	2	Science	12
Novel Clues	24	Sport	17
Temporal Misalignment	5	TV and Film	22

Table 3.5: Statistics of adversarial tactics and topics in ADVQA

question unnatural or ambiguous, we exclude it from our final dataset.[11]

We emphasize that human responses are especially useful in adversarial evaluation contexts, as they ensure that adversarial examples are genuinely challenging and realistic. Moreover, these responses are provided by each individual's intuition, creativity, and understanding. Thus, capturing variability is crucial to evaluate the benchmarks that are meant to assess evolving models aiming for human alignment. Such aspects are what traditional model-generated adversarial attacks cannot replicate. Ultimately, incorporating human responses adds depth and reliability to adversarial benchmarks, making them essential in evaluating models' true progress toward human-level understanding and their performance.

3.6 Summary

Adversarial datasets offer practical benefits for evaluating models to improve robustness and performance. Grounded in human feedback, ADVSCORE ensures that evaluations of adversarial

[11]When tasking human authors with adversarial writing of questions, Wallace et al. (2019) emphasizes the importance of "who" the authors should be: *talented and eager* question writers with *specific goals*; they should aim to generate questions that stump computers but seem normal enough for humans to answer. To make this work, they recruit members of the quizbowl community, who have deep trivia knowledge and craft question for quizbowl tournaments (Jennings, 2007). However, their challenge was to convey what is "normal" to authors and stimulate examples that can elucidate the weaknesses of QA models.

benchmarks align with human capabilities by post-hoc assessment of adversarial robustness and model improvements. Thus, applying ADVSCORE in real-time benchmark construction can aid in evaluating the robustness of the models, and integrating ADVSCORE into model training can improve their adaptability to real-world applications. In the next chapter, we shift focus to natural adversarialness—examples that emerge organically in real-world contexts—to further explore challenging benchmark construction.

Chapter 4: Not all Fake News is Written: A Dataset and Analysis of Misleading Video Headlines[1]

Following the human-in-the-loop adversarial benchmark introduced in Chapter 3, Chapter 4 shifts focus to a naturally-occurring adversarial benchmark centered on human subjectivity in real-world misinformation. Specifically, we study the task of detecting misleading video headlines—a complex, multimodal challenge where human interpretation plays a critical role. To capture this subjectivity, we develop a hierarchical crowdsourcing framework that models varying perceptions of misleadingness across both headlines and accompanying video content. By grounding the benchmark in human judgment, we expose subtle performance gaps in model behavior that would be missed by traditional, reference-based evaluations. This work demonstrates how subjectivity can be systematically incorporated into benchmark construction to better assess model robustness in real-world and ambiguous scenarios.

4.1 Motivation

Social media platforms are used by half of U.S. adults for everyday news consumption, according to Walker and Matsa (2021). They have even supplanted television as the most common purveyor of news (Wakefield, 2016). However, content created on these online platforms are

[1]Yoo Yeon Sung, Naeemul Hassan, and Jordan Boyd-Graber. 2023. Not all Fake News is Written: A Dataset and Analysis of Misleading Video Headlines. In *Proceedings of Empirical Methods in Natural Language Processing*.

often lower quality than traditional sources and more prone to false stories. Vosoughi et al. (2018) contend that false news spreads six times faster online than offline.

This work focuses on one part of this problem: does a video headline match its content. We call this **misleading video headline** detection. In text, this is referred to as incongruent headline detection (Chesney et al., 2017) and is an important problem because the headline is the first step to a reader accessing content (dos Rieis et al., 2015). While there have been efforts to identify misleading information by analyzing textual content in the headline, recent work has shown that users are more likely to believe fake news when it is accompanied by videos (Wang et al., 2021a).

Hence, it is necessary to investigate content outside the text (e.g., with videos) as it can help make a more informed decision by directly analyzing the relationship between the headline and the video.

To understand this new task, we create a new dataset—Video Misleading Headline (VMH)—that includes $2,247$ news articles labeled as *representative* or *misleading* (Section 4.2). A careful annotation process captures not just whether videos are misleading but *why*. We investigate videos, label rationales, and headline meta information (e.g., venues, news topics, and headline properties) to analyze the features that may contribute towards an instance being identified as misleading (Section 4.3). Section 4.4 shows that existing models fail to identify misleading video headlines, showing that this important but difficult task requires further research in both the text and visual domains.

A *misleading headline* is when the headline distorts the underlying content (Wei and Wan, 2017) and facts in the news body, leading the audience to imply more or less than what was actually presented in the content. For example, in our task, the headline "Obama: I'm proud to be leaving *without* scandal" does not fully engage the video's content because the headline

exaggerates the view of the content; the video plays Obama's speech that he left the administration without a *significant* scandal. This distortion makes detecting misleading video headlines even more arduous because the video content has to be integrated with the headline subtlety while assessing headline veracity.

4.2 Video Misleading Heading Dataset VMH

VMH consists of 2,247 video posts from 2014 to 2016. We focus on this period because it coincided with the 2016 US presidential election, which was rife with disinformation, and is distant enough from current events that we believe annotators can be more confident about determining whether claims are true; as even news organizations are not immune to false news (Starbird et al., 2019).

We harvested Facebook video posts from Rony et al. (2017), where we manually filtered any video that exceeded five minutes or had low-quality video or sound. The videos in VMH are average two minutes long. The resulting video posts (example in Table 4.1) come from fifty-two media venues, including the most circulated print and broadcast media and unreliable media in the US (Edelson et al., 2021; Samory et al., 2020).

We further collect venue-related information such as venue credibility[2] (e.g., High) and venue kind[3] (e.g., Broadcast). Also, we manually assigned news topics (e.g., Politics) inspired by News Areas[4] to each headline. We create audio transcripts (also released in our dataset) using automatic speech recognition software[5] whenever the video is accompanied by intelligible audio.

[2]Mediabiasfactcheck site
[3]State of the News Media
[4]News Topics
[5]https://deepgram.com

VMH Dataset	
Headline	**Clinton Says Trump "Making Up Lies" About New FBI Review**
Video	`https://www.facebook.com/watch/?v=10154955844338812`
Label	**Misleading**
Rationale	**The headline implies more than what is introduced in the video.**
Subrationale	**The headline exaggerates the video content.**
Annotator ID	`A2P8V5SKYLL5I4`
Annotator Profile	`Ages 30-49, Black, Democratic, Men, Post college`
Venue	ABC News
Venue Kind	Broadcast
Venue Credibility	High
News Topic	Politics
Headline Property	Factual Statement
Transcript	...is already making up lies about this he is doing his best to confuse misleading and discourage the American people

Table 4.1: VMH includes video headline, video, annotator's label, and rationales the label is grounded. In the video, the part about "New FBI Review" was not present, and thereby annotation is *misleading* because the headline was implying more than the video content.
Other features in the dataset include the number of tokens per headline (average 7.75 tokens) and annotator profile (e.g., gender).

4.2.1 Annotation

We ask Mechanical Turkers to identify misleading video headlines (Snow et al., 2008). We intentionally assign the annotation task to laypersons to reflect the real-world misleading headline phenomena. The three annotators undergo labeling and rationale annotation (Chandler et al., 2014).

Label Annotation. We structure the label annotation task as a series of questions to help annotators engage with the content of the headline and video (Figure 6.1). Because headlines can take different forms (statements of facts or opinions, questions, etc.), we first ask the user to determine the form of the headline. We refer to these forms as headline property in the sequel. They then engage with

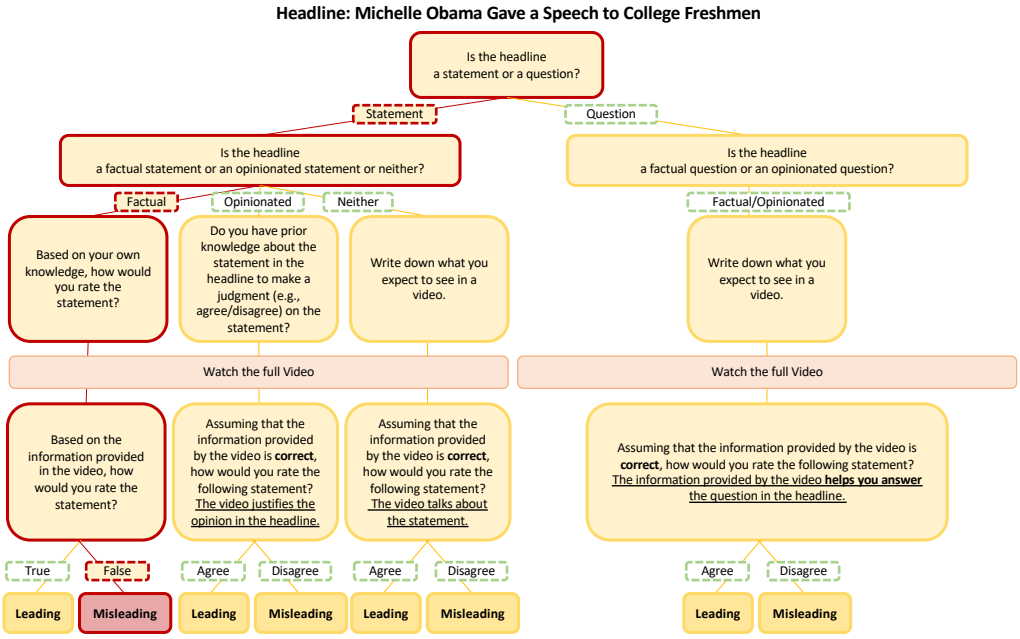

Figure 4.1: In the annotation tree, the annotators first consider if the headline “Michelle Obama Gave a Speech to College Freshmen" is a factual statement. Next, they answer the question, “Based on the information provided in the video, how would you rate the statement?” Because the answer was *False*, the implied label is *misleading*. The headline is indeed *misleading* because whether “College Freshmean” were present in the video is unclear, making it impossible to assess the veracity.

the headline in different ways depending on the headline property they selected (i.e., do they agree with the headline, do they believe the fact is true, etc.). This helps them build a mental model of the content of the hypothetical video before viewing it. We adopted this format after initial pilots indicated that merely asking if a video was misleading is too ambiguous.

After the annotator has built a mental model, we ask the annotators to watch the video and answer whether the information provided in the video is consistent with the annotator’s mental model of the video. If it is, then it suggests the video is *representative*: it answered the question asked by the headline, justified an opinion, or gave evidence of a new event.

In contrast, if the video fails this check, we conclude that the headline is *misleading*. To reflect the subtle difference in participants’ opinions, we provide answer options that represent the

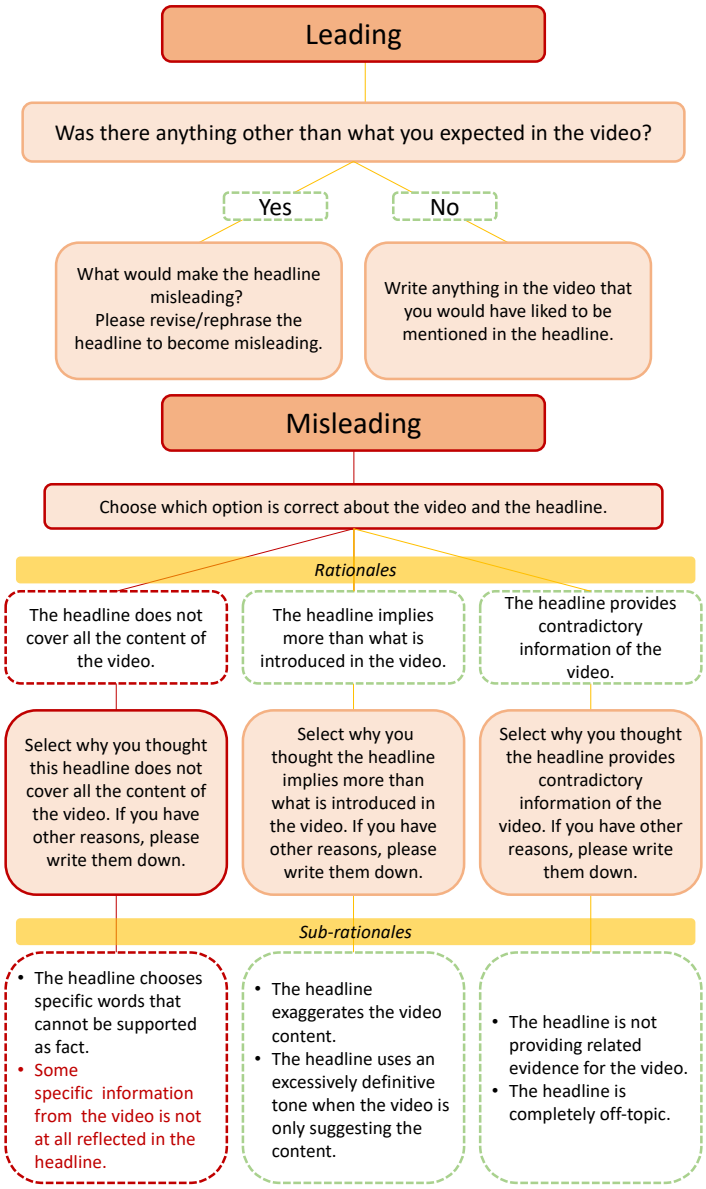

Figure 4.2: After label annotation, the annotators provide grounding for the *misleading* labels. The figure shows how rationales and subrationales are selected in a hierarchical manner.

levels of veracity or agreement with the headline (e.g., True, Mostly True, Mostly False, False, I don't know). For the translation to binary labels, we regard the last three answers as *misleading*.

Rationale Annotation. We then turn to the rationale annotation step. If their label is *misleading*, we ask the annotators to provide justifications for their decision (Figure 4.2). For example, when

an annotator labels a headline as *misleading* and chooses *The headline does not cover all the content of the video* as their rationale for the label, a subrationale is further used to reason the ways in which the headline might not contain the content. We offer pre-populated rationales to force objectivity in the annotator's decision and exploit the rationales more systematically. For subrationales, we allow the annotator to provide free-form text.

Providing such annotations can improve not just data quality(Briakou and Carpuat, 2020)—by forcing the annotator to think about their reasoning—but also model accuracy (Zaidan et al., 2007) for natural language processing tasks. After the annotation is complete, final annotations are determined using a majority vote from the three annotators(Yang et al., 2015). We do not apply majority voting for subrationales that include free-form texts.

4.2.2 Quality Control and Assessment

Quality Control. We control the quality of VMH to select good crowdworkers using their accuracy score on synthetically created accuracy check questions and MACE score (Paun et al., 2018). Accuracy check questions are synthetically created to be always misleading (obviously false). For each annotator, we calculate the ratio between the number of correct answers and the number of accuracy check questions they answered. To determine which users are reliable and to infer the labels annotators disagree on, we use a latent variable model that explicitly estimates an annotator's accuracy. This model, MACE (Martín-Morató et al., 2021) corrects for annotator-level biases (an annotator might overly favor a particular label, could have low overall accuracy, etc.). We use the point estimates—mean—from the posterior distributions of latent variables that stand for the trustworthiness of each worker (details about applying MACE to worker accuracy estimation. We

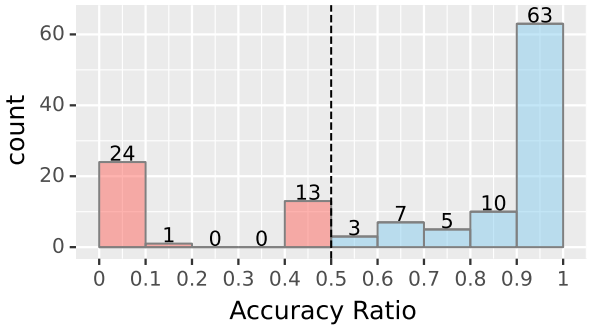

Figure 4.3: Qualified Workers by Accuracy Score Threshold. The thresholds of accuracy ratio are manually assigned to ensure *competent* workers are recruited after each annotation session.

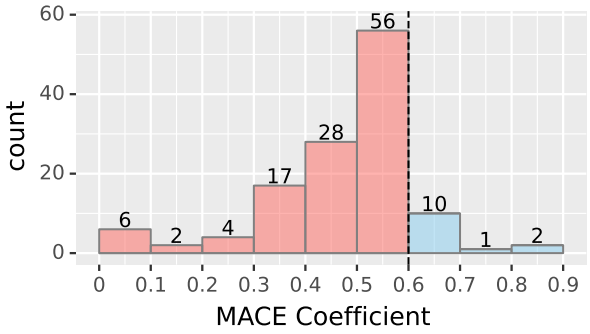

Figure 4.4: Qualified Workers by MACE Score Threshold. The thresholds of MACE Score are manually assigned to ensure *competent* workers are recruited after each annotation session.

run two annotation sessions to estimate and accumulate qualified workers. In the initial session, accuracy and MACE scores are considered to combine working agreement with known and inferred labels (Paun et al., 2018), thereby selectively filtering less competent annotators. Crowdworkers are invited back only if their accuracy (0.5) or MACE score is high enough (0.6). Each threshold is empirically assigned. This yields 88 and 13 qualified workers from each metric (Figure 4.3 and 4.4).

Quality Assessment. We report Krippendorf's α values following Toledo et al. (2019) to quantify annotation quality. Krippendorf's α value of the three annotators who passed the accuracy score threshold are 0.57 for labels and 0.33 for rationales. The Krippendorf's α values of the workers who were found to be competent according to the MACE score are 0.68 and 0.21. While the values

exhibit moderate-to-low agreement, this is expected due to the subjectivity of the annotation task (Daume III and Marcu, 2005).

4.3 Dataset Analysis

Out of 2,247 video headlines, 1,906 headlines are annotated as *representative*, while 341 headlines are annotated as *misleading*, suggesting a high-class imbalance. In this section, we investigate various aspects of VMH to gain a deeper understanding of features that could potentially contribute to a headline being classified as misleading. We further investigate the inherent qualities of VMH by examining annotation patterns in different aspects.

Misleading Features. Figure 4.5 and 4.6 suggest that the venues *TruTV* and *WeAreChange.org*

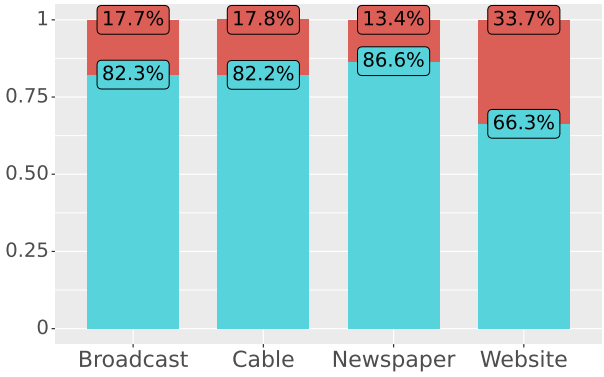

Figure 4.5: Venue Kind Distribution. The venue kind *Website* were the strongest indicators of misleading headlines. The red and blue bars denote bar proportions for *misleading* and *leading* labels respectively.

are strong indicators for misleading headlines. Also, videos from the *Website* venue (as opposed to traditional media) are likely to be misleading (29%). This suggests that the specific venue and the kind of venue may help detect misleading headlines.

Clickbait. Misleading videos and clickbait both have the same goal: to entice more people to

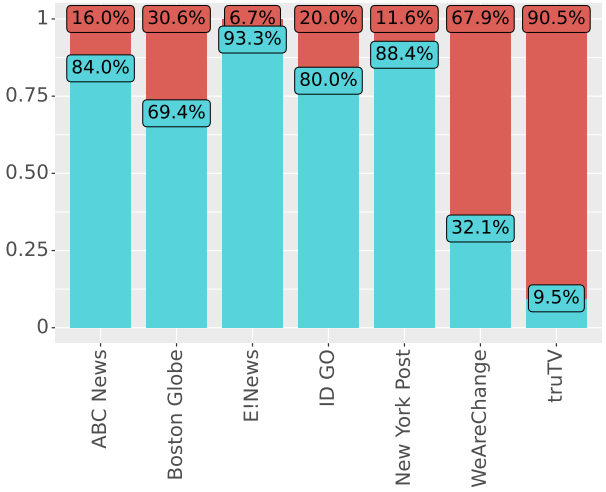

Figure 4.6: Venue Distribution. The venues *TruTV*, *WeAreChange.org* were the strongest indicators of misleading headlines. The red and blue bars denote bar proportions for *misleading* and *leading* labels respectively.

click on the underlying content. A reasonable hypothesis is that they would use similar tricks to lure in users. Thus, we reproduce the features found by (Dhoju et al., 2019) to be associated with clickbait headlines such as the number of demonstrative adjectives, numbers, and WH-words (e.g., what, who, how) for the headlines in VMH. Demonstrative adjectives appear in misleading headlines (Table 4.2), while numbers and superlative word features are less frequent in our dataset. Numbers and modal words appear in similar frequencies. Thus, misleading video headlines are not the same as clickbaits.

Clickbait Patterns	**Presence Ratio**	
	Dhoju et al. (2019)	**VMH (Ours)**
Demonstrative Adj	0.80	0.61
WH-Words	0.70	0.40
Numbers	0.72	0.60
Modal	0.27	0.20
Superlative	0.30	0.06

Table 4.2: Clickbait patterns in misleading headlines in VMH to demonstrate the difference between clickbait detection and misleading video headline task.

Investigation of Bias in Annotation. Because our dataset has many politically relevant videos, we also ask annotators' political leanings to see if it biases their annotations. A χ^2 test does not

suggest that annotations and political leanings are dependent (p-value 0.36); indeed the marginal proportion of misleading videos are comparable (Democratic: 22.9%, Republican: 22.6%, and Independent: 33%).

We also manually check fifty video headlines to see if their ideologies affected a headline's assigned label, finding no substantial consequences. For example, the headline "Charles Blow: Donald Trump is a bigot", presumably "anti-Trump", was annotated *Representative* by an annotator with a "Republican" leaning.

Task Subjectivity. Motivated by Section 4.2.2, we examine the annotations that fail to have consensus among annotator decisions: there were 1436 *representative* and 159 *misleading* instances with the perfect agreement, leaving 30% to annotations that had disagreement. In addition to disagreeing on labels, annotators disagree about why they the headline is misleading (Table 4.3).

Headlines	ID	Ann.	Rationales	Subrationales
Lester Holt Interrupted Trump Repeatedly	81	M	The headline does not cover all the content of the video	The headline is not providing related evidence for the video
	111	M	Neither of above: The headline provides contradictory information of the video	The headline chooses specific words that cannot be supported as fact
	97	R	-	-
Emily Blunt Weighs In On John Kransinskis Obsession With The D...	42	M	The headline does not cover all the content of the video	The headline chooses specific words that cannot be supported as fact
	45	M	The headline does not cover all the content of the video	Some specific information from the video is not at all reflected in the headline
	97	R	-	-
Did This Man Murder A Beautiful Country Music Producer	77	M	Neither of above: The headline provides contradictory information of the video	The headline is not providing related evidence for the video
	12	M	The headline implies more than what what is introduced in the video	The headline uses an excessively definitive tone when the video is only suggesting the content
	10	M	Neither of above: The headline provides contradictory information of the video	(Free Form Input) No mention of her being a country music producer

Table 4.3: Examples of Samples with Subjectivity. The second headline shows that each annotator's rationales are different even when the annotations are the same. The third headline shows an example where annotated subrationales all vary in their content (e.g., free-form text). ID is Annotator's ID and Ann. is the annotation result from each annotator (M: Misleading, R: Representative)

4.4 Experiments

The misleading headline detection task is challenging because of the inherent subjectivity of the task. It also necessitates multimodal approaches that can consider both the headline and the video to make inferences about the nature of the relationship (*representative* or *misleading*) between the two. Hence, in this section, we benchmark both text-only and multimodal approaches typically used for detecting video-text similarity and video-text entailment tasks.

Experiment Settings. We compare the performance of models when trained with various combinations of input features in our dataset. The features that we consider are headlines (H) and their associated video clips (V), transcripts (T), rationales, and sub-rationales (R).

For textual feature, we concatenate features as: `[SEP]` – {Headline `[SEP]` Transcript `[SEP]` rationale[6] `[SEP]` sub-rationale}. We also extract embeddings corresponding to two multimodal models. We use VideoCLIP (Xu et al., 2021b) and VLM models (Xu et al., 2021a) that adopt zero-shot transfer learning to video-text understanding tasks. VideoCLIP trains a transformer model using a contrastive objective on paired examples of video-text clips that maximize association between temporarily overlapping text and video segments (Xu et al., 2021b). In contrast, VLM is a task-agnostic multimodal learning model that uses novel masking schemes to improve the learning of multimodal fusion between the text and the video. We finetune a classification layer that takes input features extracted from video and text-based encoders as described above to predict the label associated with a given video-headline pair.

Data and Evaluation Metrics. We divide VMH into three sets: 70% for the training set, 15% for

[6]While gold rationales might not be available during inference, our objective to study them as features are to highlight and understand if and how rationales can help improve detection accuracy in this task. We leave automatic prediction of the rationales to future work.

Model	Input	Evaluation Metrics				
		F1-Score	Precision	Recall	AUPRC	Accuracy
BERT	**H**	**0.16 (0.07)**	**0.29 (0.14)**	0.11 (0.05)	**0.17 (0.02)**	**0.82 (0.01)**
	H+T	0.16 (0.08)	0.26 (0.11)	**0.12 (0.06)**	0.15 (0.01)	**0.82 (0.01)**
VideoCLIP	**H**	0.16 (0.06)	0.22 (0.05)	0.13 (0.06)	0.17 (0.01)	0.80 (0.01)
	V	0.17 (0.03)	0.25 (0.06)	0.14 (0.04)	0.16 (0.00)	0.79 (0.02)
	V+H	0.26 (0.09)	0.32 (0.13)	0.24 (0.09)	0.20 (0.04)	0.79 (0.05)
	V+H+T	0.21 (0.04)	0.29 (0.06)	0.17 (0.03)	0.17 (0.01)	0.80 (0.01)
	V+H+T+R	**0.53 (0.06)**	**0.65 (0.08)**	**0.44 (0.06)**	**0.41 (0.05)**	**0.88 (0.01)**
VLM	**H**	0.18 (0.05)	0.20 (0.06)	0.19 (0.09)	0.16 (0.01)	0.76 (0.04)
	V	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.15 (0.00)	0.83 (0.00)
	V+H	0.22 (0.06)	0.23 (0.05)	0.22 (0.06)	0.18 (0.02)	0.77 (0.02)
	V+H+T	0.23 (0.04)	0.23 (0.04)	0.56 (0.01)	0.17 (0.01)	0.76 (0.01)
	V+H+T+R	**0.56 (0.03)**	**0.63 (0.02)**	**0.52 (0.05)**	**0.40 (0.03)**	**0.88 (0.00)**

Table 4.4: Benchmark Evaluation Results. Rows for each model shows performance with different input features: headlines (**H**), videos (**V**), transcripts (**T**), and rationales (**R**). The reported metrics are the average F1-score, average Precision score, average Recall score, average AUPRC score, and average accuracy score of 5 replicates of stratified random splits of the train, valid, and test sets. The brackets indicate standard deviation for each metric.

the valid set, and 15% for the test set. We evaluate using the following metrics: F1, precision, recall, AUPRC score, and accuracy. We report the precision and recall scores of the positive class, *misleading*. Each metric is estimated by averaging five replicates of stratified random splits.

4.5 Results

Experiment Results. Table 4.4 reports the main results: the multimodal models that use all the features, {Video Frame + Headline + Transcript + Rationale (V+H+T+R)} result in the best performance across the board, outperforming text-only based model. Adding rationales that provide information about the headline and video relationship improves metrics across the board. F1-scores drop when transcripts are augmented to {Video + Headline} the multimodal models. This could be attributed to the quality of the transcripts automatically extracted from the videos.

In the next section, we perform an analysis to validate the utility of the multimodal features

in our dataset in a partial-input setting. Furthermore, we explore how the subjectivity in the task can affect the model performance.

Partial Input Analysis. Validating a dataset with a partial-input baseline is now important in multimodal domains (Thomason et al., 2019). Artifacts in the dataset can lead the models to *cheat* using shortcut features that can result in poor generalizability (Feng et al., 2019). Thus, in our case, we also experiment with unimodal settings (partial input) — {Video} and {Headline} — to ensure that VMH does not contain such artifacts. The results show that using only video or text-based features result in poor F1-scores ($0.16 - 0.18$) relative to utilizing multimodal features (F1-score: > 0.22).

Model Subjectivity Analysis. To understand the subjectivity of the task (Section 4.3), we also report F1-scores on the subset of the dataset, *subjective* samples (30%), that had low consensus in the annotation process. Training on this subset, even the best model that utilizes all the features: {Video + Headline + Transcript + Rationale} only gets an F1-score of 0.12 and 0.10 with the VideoCLIP and VLM models respectively compared to the F1-scores (i.e., 0.53, 0.56) using the entire training set. The degraded performance suggests that the difficult instances for humans to reach a consensus on might not include any reliable features for the model, indicating that high subjectivity is indeed a factor leading to poor detection.

Video-Text Entailment Analysis. We investigate how the misleading headline detection task differs from other video-text entailment tasks by comparing entailment properties and annotations.

We use transcripts as video representation and headlines to predict each sample's entailment relation. We adopt the RoBERTa NLI model[7] to infer the relation between the transcript and the

[7] fine-tuned on SNLI, MNLI, FEVER-NLI, and ANLI

Headlines	Transcripts	Entail	Score	Answer
The sounds of emotions	... We use the principles of music to work with rhythm and melody to regain the functional use of language. Phrase is if we... ...Nice job. Let's all. Well You wanna skip this up? Okay. Do you wanna skip it or singing it? You wanna try to sing it? Let's jump to the chorus. Okay? So darling then. Music is what emotions sound like ...	✓	0.71	M
There is a double standard	... Is there a double standard when it comes to transparency between Trump and Clinton? Well, of course, there's a double standard...He's doing over a hundred foreign deals and he wants to be both the commander chief and the representative in the world for the United States... I mean, the difference between telling somebody you had pneumonia on Sunday instead of Friday is not even in the same league really. ...	✗	0.20	R
Honor a Vet I Warfighters	... Having worked with veterans throughout my career, I know firsthand the importance of honoring our troops. This veterans day our series the war fighters and history are partnering with Team Rub con to create honor event. ...Honor the vets and more fighters in your life, and share a photo and a story today. Learn more history dot com honor that. ...	✓	0.53	R

Table 4.5: Example of Comparison between Entailment Result and Annotations. The first headline shows high entailment score with the transcript while annotated as *misleading* with the rationale of "The headline does not cover all the content of the video". The second and third headline are predicted with low entailment score or "not entail" while being annotated as "representative" by majority annotators.

headline. We average the entailment score between chunked sentences from transcripts and the headlines to compromise the different lengths. To calculate if there exists any correlation between entailment predictions and the labels, we conduct a t-test (Gerald, 2018). The p-value is 0.01, which indicates that the difference between the two is statistically significant.

Table 4.5 shows how entailment decisions contradict the annotator's judgments. For example, the first headline shows a high entailment score with the transcript while annotated as misleading with the rationale of "The headline does not cover all the video content". The second and third

headlines are predicted with low entailment scores or "not entail" while being annotated as *representative* by majority annotators.

4.6 Summary

This work presents VMH, a dataset of misleading headlines from social media videos. Our annotation scheme reduces the task's subjectivity, and we verify the reliability of the annotations. We believe incorporating the crowd workers' distinct opinions (e.g., headline types and rationales) on misleading headlines allows crude reflection of the current social media misinformation phenomenon. Through their lenses, we anticipate a better understanding of how people perceive misinformation in misleading video headlines and for future work, use it to generalize the detection models that are soon to be deployed.

To obtain even more realistic examples for this task, we encourage applying a dynamic adversarial generation pipeline. Motivated by Eisenschlos et al. (2021b); Wallace et al. (2019), misleading headlines could be authored by humans guided to break the existing video headline detection models. For example, while they are writing a "misleading" headline, if the model falsely predicts the headline as "representative", it would become an adversarial, *realistic* example (Ma et al., 2021a). These examples can prevent the model from learning superficial patterns (Kiela et al., 2021) and further be developed to become a *robust* tool for journalists to prevent them from making "honest" mistakes when writing video headlines (Dhiman, 2023).

In the next chapter, we shift our focus toward evaluating how trustworthy a language model is by introducing a human-grounded benchmark designed to evaluate model calibration and robustness through incrementally revealed, human-authored adversarial questions that challenge.

Chapter 5: GRACE: A Granular Benchmark for Evaluating Model Calibration Against Human Calibration[1]

In Chapter 5, we extend the HITL adversarial generation method introduced in Chapter 3 to develop a benchmark for evaluating and improving the trustworthiness of NLP systems, with a particular focus on calibration. Since users often treat model confidence as a proxy for reliability, we investigate how well model confidence aligns with correctness compared to humans. We collect progressively adversarial examples authored by humans, alongside expert responses and inferred confidence—estimated via response timing—in a competitive human-AI setting. This enables us to build a benchmark capturing human and model correctness, confidence, and abstention behavior. Using these signals, we propose a benchmark and two metrics to systematically compare LLM calibration to human calibration, revealing that models are typically more overconfident, especially when incorrect.

5.1 Motivation

Because language models are often miscalibrated, they are often confidently wrong (Kaur et al., 2020). This mismatch between accuracy and confidence causes users to trust models more

[1]Yoo Yeon Sung*, Eve Fleisig*, Yu Hope, Ishan Upadhyay, and Jordan Boyd-Graber. 2025. GRACE: A Granular Benchmark for Evaluating Model Calibration against Human Calibration. In *Proceedings of the Association for Computational Linguistics.*

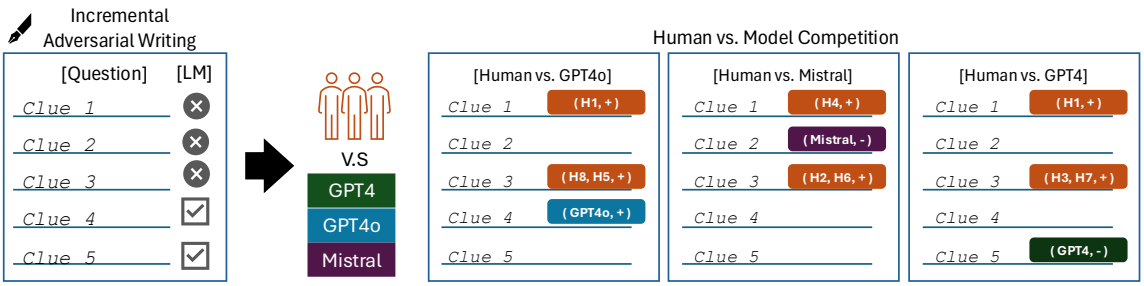

Figure 5.1: To create the GRACE dataset, expert question writers develop questions with multiple clues of decreasing difficulty via an interface that shows where weaker models struggle to answer the questions. These questions are used in human vs. model competitions where teams compete to be the first to interrupt the sequence of clues with a correct answer. We record when the human and model teams buzz in each question with their correctness (+) or incorrectness (-) (*buzzpoints* 🔔). The dataset contains all buzzpoints throughout the competition. Then, CALSCORE measures each model's human-grounded calibration performance (§ 5.3).

than they should (Caruana, 2019; Deng et al., 2025), even over their own correct judgment (Krause et al., 2023; Stengel-Eskin and Van Durme, 2023; Liu et al., 2024b; Si et al., 2023). These issues are particularly severe when models are miscalibrated in ways that humans are not: users expect models to be at least as calibrated as humans, and when models are worse, users are often not prepared to address these errors (Li et al., 2024a). Thus, models should be *at least* as calibrated as humans, making it especially crucial to identify when models commit calibration errors that humans do not. However, existing work on model calibration lacks comparison with human calibration.

We thus introduce GRACE, a **Gra**nular, Human-grounded Benchmark for Model **C**alibration **E**valuation. Each instance allows **fine-grained calibration measurement** using an incremental question-answering (QA) framework. Expert writers design GRACE questions, each consisting of at least five sentences of clues that gradually become easier. To prevent models from being confused by ambiguities or false presuppositions (Min et al., 2020, 2022), we require that clues challenge models but remain clear for humans. This format measures model calibration with

human performance as a reference point: models should give correct answers earlier and more confidently than humans, while minimizing confidently incorrect guesses (§ 5.2).

GRACE incorporates human responses from our live QA competitions. Unlike prior calibration evaluation methods that only allow model–model calibration comparisons, our dataset thus allows direct *human–model* calibration comparison. **GRACE is the first benchmark dataset designed to evaluate model calibration grounded in human performance**. This unique dataset is the foundation for a new metric (CALSCORE, § 5.3). In contrast to other calibration evaluation methods that only calculate aggregate calibration over the entire dataset, GRACE also facilitates per-instance evaluation, which helps in identifying specific contexts where models are much worse than humans at avoiding confidently incorrect answers.

Language models are more overconfident than humans in incorrect answers and relatively underconfident in correct answers. In contrast, humans tend to be highly confident—over 50%—when correct (§ 5.4.1). Models struggle with abstract descriptions—they are both overconfident and inaccurate—but excel in retrieving facts given unambiguous clues (§ 5.4.3). We conclude with a discussion of how GRACE and CALSCORE can aid in the creation of models that are more accurate and better calibrated.

In sum, we (1) introduce GRACE, a benchmark that compares LLM and human calibration to identify LLM calibration failures, and a novel calibration metric using our benchmark; (2) conduct extensive human response collection, which grounds GRACE in accurate human confidence calibration assessment; (3) ensure that GRACE contains expert-authored and repeatedly validated questions that are harder and longer than previous work; and (4) analyze human vs. LLM calibration, finding that, relative to humans, LLMs are underconfident in correct answers and overconfident in wrong answers.

5.2 GRACE: Dataset Development

To create our dataset, expert writers and editors first construct incremental, adversarial, and rigorously quality-checked examples (§ 5.2.1). Then, we collect model guesses and confidences on these questions (§ 5.2.2), and compare them against human performance in a live competition (§ 5.2.2.2).

5.2.1 Question Writing Process

Collecting QA pairs from expert writers. Similar to Chapter 3, we recruit experienced question writers and editors to ensure that questions are high-quality. However, one distinction is that we target specific questions that can help in calibration measurement. We thus adopt an incremental structure where the clues become progressively easier with adversarial clues; clues become easier over time for both humans and models, while still remaining more difficult for models than for humans. We hire six writers and ten editors to author the questions, following this format.[2] The questions contain 575–650 words[3] and cover content across a range of subjects.[4] All questions are reviewed by the writer, category editor, and head editor to check that clues are unambiguous and factually correct.

Interface setup. To create incremental and adversarial questions, writers and editors use a human-AI collaborative writing interface (Figure 5.2). Because these examples are meant to

[2]Writers and editors were recruited via a public quizbowl forum and were located in the US and UK. All editors underwent IRB training and had written for at least three previous tournaments. Writers were paid $5 per question and editors paid $1 per question edited. A head editor with 5 years of experience writing and editing quizbowl questions, including as head editor of two previous tournaments, supervised the writers and provided an additional quality check.

[3]We refer to these as *questions* although they are not grammatical questions, but rather sequences of sentences with clues uniquely identifying an answer (examples in Figures 5.2 and 5.3).

[4]20% literature, 20% history, 20% science, 15% arts, 15% social sciences, 5% geography and current events, and 5% myth, pop culture, and other.

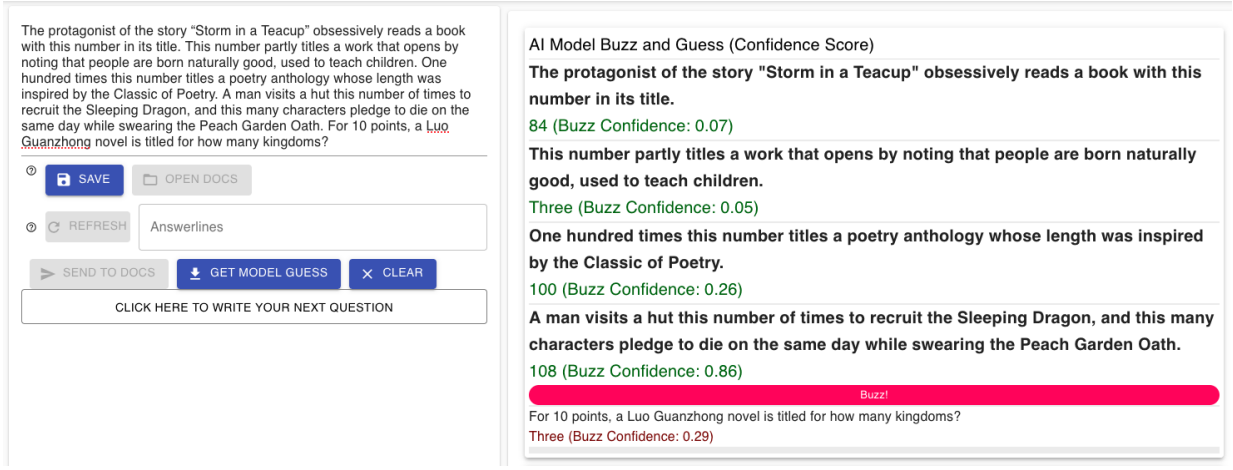

Figure 5.2: Example question on Chinese literature (with the answer of three) being written in the interface. Writers compose questions in the left box. On the right, they see the model's guess and confidence after every sentence and the point at which the model would buzz in and attempt to answer. Writers learn which sentences make it harder for models to answer correctly and refine their questions to be sufficiently hard for models but still answerable by humans. This incremental, adversarial format permits granular calibration measurement.

be incremental, we break the input into sentences $\{s_1, s_2, \ldots s_k\}$. GPT-3.5 provides a guess $\{a_1, a_2, \ldots a_k\}$ for each sentence in addition to its confidence $\{c_1, c_2 \ldots c_k\}$. The interface also highlights when the model confidence would be high enough to buzz.[5]

To ensure that questions remain incremental for models (§ 2.2.1.2), we instruct writers to write questions so that model's guess is correct *no earlier than* the penultimate sentence.[6] As experienced question writers, they use their domain knowledge to ensure difficulty also decreases for humans, so that most humans can answer correctly by the end. Writers dynamically interact with the models to refine their questions (§ 2.2.1.2) (You and Lowd, 2022; Eisenschlos et al., 2021a). For example, the second line in Figure 5.2 was originally “This number of characters

[5]For the writing interface, the logistic regression model was trained with two kinds of features: GPT-3.5's logit based confidence, and pre-designed features derived from the question, answer, and metadata. Features include text-based metrics (e.g., TF-IDF scores, overlaps between Llama predictions and TF-IDF guesses), probabilistic outputs (Llama log and prompt probabilities), and contextual indicators (sentence index, length, and presence of phrases like “10 points”). This model was trained on Rodriguez et al. (2019a), also pyramidal questions, using Llama-13b predictions (Touvron et al., 2023). A primary distinction from Wallace et al. (2019) is that their interface only showed the final correct guess.

[6]The model's confidence for *correct* answers should remain low for all but the last two sentences; clues that trigger high-confidence, *incorrect* model guesses are encouraged.

appear in the name of a Chinese classic," which the model answered correctly. Instead, the writer revises the first line of the text, which fools the models while allowing humans to answer correctly. The final dataset consists of 243 QA pairs, with a total of 1,236 sentences of clues. Each sentence uniquely points to the answer, making it usable as a standalone QA pair.[7]

5.2.2 Collecting Human–model Buzzpoints

The questions described above are designed to be read aloud and interrupted. In the competition, teams compete by buzzing to interrupt and answer, with this timing referred to as buzzpoints 🔔 (Figure 5.3). However, modern LLMs do not operate this way: they generate an output given an input. Thus, we first extract guesses from models and humans *offline* to assess teams on the same questions (§5.2.2.1). We then compute the model buzzpoints for each clue. Finally, using these precomputed model buzzpoints, we host live human–computer competitions to collect real-time human buzzpoints (§5.2.2.2).

5.2.2.1 Offline human and model buzzpoints

Model guesses and confidence. We first break each question into clues. We then retrieve a model's guess given the first n clues with a prompt using a TF-IDF retriever to select similar question-answer pairs from QA datasets. Before deciding when to buzz, models need to generate answers as clues are revealed. We call this process "guessing" and the model is used as a *guesser*.

To retrieve a model's top guess after n clues have been revealed, we prompt the model with the first n clues from the question. To retrieve the best guess, we employ a "retrieval and guess"

[7]While we have been using the term *clue* informally, for the purpose of analysis, we now use the term *clue* as a substring of the full question, averaging 13 words or 35 characters, incrementally extending from the beginning. Clues are split at whitespace boundaries and may contain multiple pieces of information about the answer.

Q: In a reference to an object notably missing from one of these works, Diemut Strebe used genetic samples from a man's great-great-grandnephew to clone a certain feature. In one of these works, Utagawa Togokuni's Geishas in a Landscape **GPT-4o: "Rodin statues"** hangs on a yellow wall behind a man in a fur-brimmed hat. The backside of The Potato Peeler includes one of these works featuring a man in a straw hat. One work shows a man in a light-blue green suit against a light-green-blue swirling background, and another dedicated to Gauguin shows subject with cropped hair and a red beard. For 10 points, a Dutch artist **H1: "Van Gogh self-portraits"** painted what portraits of himself with a bandaged ear?

ANSWER: self-portraits of Vincent Van Gogh

Figure 5.3: While GPT-4o buzzes too early with an **incorrect answer**, losing 5 points, the human team (H1) buzzes later with a **correct answer**, earning 10 points. Both teams must balance accuracy and speed; here, `GPT-4o` shows poorer calibration than `H1`.

prompt to enhance QA performance.[8] Then, we train a TF-IDF model as the retriever with past quizbowl questions following Rodriguez et al. (2019a). The main goal is to reduce hallucinations and guide the model to learn the granular clue and guess format.

To determine if model guesses were correct, our post-processing uses both transformer-based answer equivalence (PEDANTS, Li et al., 2024b), followed by manual verification by dataset editors. We store the resulting guesses and two forms of confidence from LLMs, token logits[9] and verbalized confidence. Logit-based confidence are the average of the exponentials of the token logit probabilities, while verbalized confidence prompts models to directly express confidence in the output tokens. Log probability of the generation is a common method to estimate the model confidence (Nguyen and O'Connor, 2015). To get the confidence score in our setup, we retrieve the logit for each generated token, and take the average of the exponentials of these logit values (Huang et al., 2023a). On the other hand, recent study shows verbalized probabilities can

[8]Prompt: *Given the following information, provide the title of the Wikipedia page that best answers the last question fragment. If unsure, provide your best guess. The answer should be concise. Question: {retrieved examples}. The answer is: {retrieved examples}. Question: {each clue}. The answer is:*

[9]GRACE answers are short, typically 3-4 words long, making token logits a reliable measure of confidence.

be better calibrated than log probabilities (Tian et al., 2023; Xiong et al.), which motivate us to include the verbalized confidence in our experiments. We follow the above retrieval-based prompt and add the instructions from Tian et al. (2023) to return the confidence to attain verbalized probabilities.[10]

To ensure accurate extraction of probability scores from model outputs, we initially define the desired format based on the prompt. We then proceed to identify and print any cases where confidence scores are not successfully extracted. By observing these cases, we can discern patterns and refine our post-processing rules. This iterative approach allows us to capture as many corner cases as possible, enhancing the robustness of our data extraction process.

Precomputed Model buzzpoints. Our metric, CALSCORE (§ 5.3.1), uses the raw confidence values from continuous probabilities. On the other hand, in our competitions (§ 5.2.2.2), models can only buzz at a single position. Thus, we turn the continuous confidence into a binary buzz by thresholding to indicate when the model buzzes. For each model, we set a threshold based on human gameplay data on preexisting non-adversarial questions (He et al., 2016).

We used question data from the 2023 Expo quizbowl competition, where expert players competed against ChatGPT as a testbed to assign buzzer threshold. The dataset includes questions covering various topics, with recorded buzz and guess correctness.

To align model buzzing behavior with human tendencies, we estimate the likelihood that a player has correctly answered a question by a given word position. This estimate helps determine

[10]Prompt: *Given the following information, provide the title of the Wikipedia page that would best answer the last question fragment. If you are not sure, just give your best guess. If you don't know, answer None. The answer should be as short as possible.* ***While you give the guess, please also provide the probability that it is correct (0.0 to 1.0). Give ONLY the guess and probability, no other words or explanation. For example: The answer is: <most likely guess, as short as possible; not a complete sentence, just the guess!> Probability: <the probability between 0.0 and 1.0 that your guess is correct, without any extra commentary whatsoever; just the probability!>*** *Question: {retrieved examples} The answer is: {retrieved examples} Question: {each clue}*

an appropriate threshold for buzzing, enabling fair comparison between human and model performance. Following the approach of Rodriguez et al. (2019a), we define $\pi(t)$ as the probability that a player has answered correctly by position t in the question, computed as:

$$\pi(t) = 1 - \frac{N_t}{N}, \tag{5.1}$$

where N is the total number of player-question records, and N_t is the number of instances where a player has answered correctly by position t. To make this probability easier to use in practice and smoother for modeling purposes, we adopt a polynomial approximation:

$$\pi(t) = 0.0775t - 1.278t^2 + 0.588t^3. \tag{5.2}$$

This polynomial provides a data-driven estimate of how human accuracy evolves as the question progresses, supporting principled decisions about when a player—or model—should buzz. Using this estimated human buzzing behavior, we adopt calibrated thresholds to determine when models should buzz. Specifically, we set the confidence thresholds to -0.03 for GPT models, and -0.05 for Mistral models, allowing for a data-driven approach to determining optimal buzz thresholds.

This threshold is chosen to maximize the probability of the model buzzing correctly before the average trivia player as estimated by the *expected wins* metric from Rodriguez et al. (2019a). When the logit score exceeds the threshold, the model buzzes in, marking its buzzpoint:

Offline human guesses. To compute humans' raw accuracy, independent of confidence (§ 5.4.1), we survey fifteen players on 35–40 held-out GRACE questions. Like the models, players view clues, submit their guess after each clue, and indicate whether they would buzz at that

Algorithm 1 Find model buzzpoints for a question

Input: N, the number of clues in the question; and t, the buzz threshold.
Let $n = 0$
while $n < N$ **do**
Prompt the model to answer the question, given the first n clues.
Compute the model's confidence c in its top guess by summing the log probabilities of the tokens comprising the guess.
if $c > t$ **then**
Buzz in
break
end if
n += 1
end while

point. However, this data collection format is time-consuming and tedious (one player called it "remarkably hard"), potentially reducing player engagement and response quality. Instead, we collect human calibration data through a fast-paced trivia tournament.

5.2.2.2 Human buzzpoints, *live* competition

GRACE records human and computer guess correctness on interruptible questions designed to challenge model calibration. A human moderator reads each question to both teams (a model and a team of humans). Teams compete by buzzing to interrupt and answer. Model buzzpoints are computed in advance (§ 5.2.2.1). When the model's confidence exceeds the threshold, the reading stops with a buzz sound, and the model's guess is announced.

Human buzzpoints. In contrast, human buzzpoints are recorded in real time when the moderator is interrupted. Players press a physical buzzer when confident in an answer, and the moderator verifies if the answer is correct. We log the timing of human teams' buzzes and answer correctness.

If a team answers incorrectly, the moderator continues reading until the other team buzzes in. Because earlier clues are harder, more skilled teams tend to buzz earlier, while less skilled teams wait until near the end. Thus, teams must be knowledgeable *and* well-calibrated to buzz optimally.

Human players in live competitions. Our three competitions consist of a total of 93 matches involving 17 human trivia teams and three LLMs (GPT-4o, GPT-4, and Mistral-7b-Instruct). Of these, 55 are human vs. model matches, while 38 are human vs. human matches. For the matches, the 243 QA pairs are divided into 12 sets of 20, stratified by category, with three questions for tiebreakers.

Hosting real-time competitions with human players provides several benefits: (1) direct comparison of confidence calibration between humans and models on the same questions, (2) recruiting experienced players skilled in calibrating their answers,[11] and (3) validating that questions are human-answerable and unambiguous, as an additional quality check for the dataset.

5.3 Human-Grounded Calibration Evaluation

To compare model and human calibration, we analyze response correctness and buzz decisions. Then, we introduce a baseline metric, CALSCORE.

This metric differs from the ADVSCORE (Chapter 3) or traditional calibration metrics (§ 2.1.2) in two important ways. First, while ADVSCORE identifies questions where humans succeed and models fail, it operates at a binary, dataset-level resolution and does not consider model confidence. In contrast, CALSCORE evaluates how well a model's confidence aligns with its actual correctness on each question, enabling fine-grained, per-instance calibration analysis. Second, CALSCORE explicitly incorporates human performance by penalizing cases where the

[11] After the competitions, we survey players about their experience levels and individual strengths. Respondents had an average of 5.5 years of previous experience playing quizbowl. 22% of players had studied or were currently studying in the physical sciences or engineering; 31% studied computer science or math; 17% studied the humanities; 13% studied a combination of fields; and 17% were undecided. Since quizbowl players typically specialize in certain categories and learn more about those areas, we also asked them for their areas of specialization. 39.13% of respondents listed the sciences as an area of specialization; 21.74% listed history; 39.13% listed the social sciences; 52.17% listed literature; 39.13% listed fine arts; and 21.74% listed geography or current events.

model is ***confidently incorrect*** *on questions that humans answer incorrectly or with* ***low confidence***. This allows us to highlight not just where models are wrong, but where they are overconfident in ways that diverge from human judgment. Overall, CALSCORE provides a more targeted evaluation of model calibration, complementing adversarial evaluations by focusing on confidence reliability rather than just correctness.

5.3.1 Human-grounded metric: CALSCORE

CALSCORE evaluates model calibration error while incorporating human buzzpoints. This adjustment reflects the structure of the competition—models must be confidently correct before humans know the answer. The adjustment also places higher weight on instances where model errors are more likely to mislead users—if a model is confidently incorrect when humans are still uncertain, humans are less likely to recognize and override the error (§ 5.4.2).

Using the live competition data, we track the proportion of humans answering correctly up to a specific clue so that the metric applies higher penalties and rewards for earlier (harder) clues. To measure the expected probability of a team buzzing correctly on a given question, we consider teams' buzzes at each clue t of question q. We define h_t as the cumulative probability of a human team correctly buzzing up to clue t, calculated as the number of correct buzzes by human teams up to t divided by total buzzes by human teams up to t. For model responses, g_t indicates the correctness of a model's guess at clue t (1 if correct, -1 if not), and c_t indicates the model's confidence in its guess.

5.3.2 MCE: Unadjusted model calibration error

As introduced in the § 2.1.2, model calibration reflects how well a model's confidence aligns with its actual correctness. Traditional metrics like ECE focus on aggregate trends, but in our setting—where each question consists of a sequence of clues and both timing and confidence matter; we require a more fine-grained, per-question formulation.

To address this, we define an unadjusted model calibration error (MCE) that evaluates how well-calibrated a model is on a single question. The normalized expectation $r(\mathbb{E}_t\left[g_t c_t\right])$, calculated over clues in a single question, measures calibration as the expectation that the model answers correctly, weighted by confidence. $r(x)$ renormalizes our metric to a [0, 1] range:

$$\textsc{CalScore}(x) = 1 - r\left(\mathbb{E}\left[(1-h_t)g_t c_t\right]\right).$$

$r(x)$ is a normalized sigmoid function designed to map an expected value from a [-1, 1] range to a [0,1] range.

$$r(x) = \frac{\sigma(x) - \sigma(-1)}{\sigma(1) - \sigma(-1)},$$

where

$$\sigma(x) = \frac{1}{1+e^{-x}}.$$

Conversely, $1 - r(\mathbb{E}_t\left[g_t c_t\right])$ evaluates the model's calibration *error* on that question. High-confidence incorrect answers and low-confidence correct answers result in higher error, indicating poor calibration on question q.

5.3.3 CALSCORE using GRACE

Formulating CALSCORE. To better reflect real-world, competitive QA settings, we extend MCE by incorporating human buzzing behavior into the CALSCORE. Specifically, we modify the expectation to weight model calibration more heavily during the portion of the question before most humans would typically answer. This adjustment simulates a realistic competition dynamic: in quizbowl, buzzing early with confidence is rewarded—if correct—and penalized more severely—if incorrect.

To do this, we use *empirical human buzz distributions* to estimate when the humans would buzz on a given question. We then apply higher weights to model decisions made before that human buzzing threshold and lower weights after it. This means the model is rewarded more for being *confidently correct before humans typically answer, and penalized more for being confidently incorrect based on the same clue*. Conversely, when models are cautious or incorrect after the human buzz point, the penalty is reduced, since humans also struggle at that stage. This facilitates the identification of specific cases where models are less calibrated than humans. Also, it helps identify questions where the model is either outperforming or underperforming relative to human decision timing and confidence, providing a more targeted lens on practical calibration under competitive conditions.

We thus weight the calibration at clue t by $(1 - h_t)$, the proportion of humans who have not yet answered correctly by clue t. A high score of $r(\mathbb{E}_t\left[(1 - h_t)g_t c_t\right])$ indicates that the model is well-calibrated relative to human buzz performance.[12] Conversely, we estimate the expected

[12]A model is perfectly calibrated when it buzzes with full confidence ($c_t = 1$), is always correct ($g_t = 1$), and answers before humans buzz correctly ($h_t = 0$), resulting in $r(\mathbb{E}_t\left[(1 - h_t)g_t c_t\right]) = 1$.

Clue (t)	CALSCORE$_c$	Text	Model Guess	Correct (y)	Confidence
0	-0.27	He was born Eric Arthur Blair…	Dickens	-1	0.3
1	-0.08	This British writer is known for his dystopian themes…	Lorca	-1	0.1
2	-0.35	He coined the term "doublethink" and envisioned a regime where "Big Brother" watches everyone.	Marx	-1	0.7
3	0.09	His most famous works include *Animal Farm* and *1984*.	Orwell	1	0.9

Table 5.1: Clue-by-clue model predictions with confidence scores and correctness. CALSCORE$_c$ is the per-clue score, and the final CALSCORE is computed by averaging and normalizing these values: -0.27, -0.08, -0.35, $0.09 \Rightarrow$ raw score -0.61, normalized to 0.18.

probability for cases where the model does *not* improve over humans, either due to incorrect answers or low confidence:

$$\text{CALSCORE}_q = 1 - r(\mathbb{E}_t\left[(1 - h_t)g_t c_t\right]). \tag{5.3}$$

This adjustment evaluates the model's calibration error relative to human calibration performance on the same question. We then define CALSCORE$_D$, the human-adjusted model calibration error for a benchmark D, as the average of CALSCORE$_q$ across all questions.

Illustrating CALSCORE. In Table 5.1, CALSCORE first builds on unadjusted model calibration (MCE), which does not incorporate human data (Section 4.2). MCE penalizes confidently wrong answers by multiplying correctness (1 or -1) with model confidence (0–1); this captures models that are confidently wrong, when correctness is negative and confidence is high. CALSCORE then incorporates human data by the proportion of humans yet to answer correctly, reflecting calibration relative to human performance.

The overall CALSCORE is computed by averaging the individual scores for each clue (CALSCORE$_c$). Given the per-clue values of -0.27, -0.08, -0.35, and 0.09, the resulting score is

-0.61. The final CALScore is 0.18, obtained by applying the normalization (0.82), which is then subtracted from 1.

5.3.4 CALSCORE^2 using GRACE

Defining CALSCORE^2. To evaluate model calibration in the context of buzzing decisions, we introduce CALSCORE^2, a secondary metric designed to capture when a model chooses to buzz relative to both its own confidence and human response behavior.

We frame this as a stick-breaking process: at each time step t, the model decides whether to buzz (i.e., commit to an answer) or to wait for more information. Buzzing at step t removes some probability mass from future decisions; the model cannot buzz again, and the remaining probability is *broken off* from the total mass. This reflects the constraint that models, like humans in quizbowl, can buzz only once.

We define the model's buzz confidence at step t as:

$$b_t = c_t \prod_{i=0}^{t-1} (1 - c_i).$$

where c_t is the model's confidence at time t, and the product term ensures that the model has not already buzzed at any earlier step. This construction guarantees that: $\sum_{t=0}^{T} b_t = 1$, which induces a valid probability distribution over buzz positions. The analogy is to starting with a unit-length stick: at each timestep, a portion b_t is broken off and used, reducing the remaining length available for future buzzing decisions. This formulation ensures that buzzing is a one-shot process, mirroring the real-world constraint: the model can make a decision only once, and must allocate its probability mass carefully across time.

To compare model and human calibration in this setting, we define **CALSCORE**^{2}, using human buzz probabilities h_t (proportion answering correctly by step t) as a benchmark. Let $K_t = h_t \sum_{e=0}^{t} b_e g_e$, where the system buzzes correctly before or at step t. The full metric is:

$$\text{CALSCORE}^2 = 1 - \left(\sum_{t=0}^{T} K_t + \left(1 - \sum_{t=0}^{T} h_t\right) \sum_{t=0}^{T} b_t g_t \right), \tag{5.4}$$

which rewards early correct buzzes relative to humans and penalizes overconfidence when humans abstain.

Illustrating CALSCORE^2. Table 5.2 presents a step-by-step breakdown of a question where both system and human buzzes are tracked across multiple time steps. At each clue t, the system

Clue (t)	**Question**	**System Guess**	**Conf.**	h_t	b_t	SH_q
0	This author talked about his time fighting fascists in his autobiographical book *Homage to Catalonia*.	Lorca	0.1	0.1	0.1	0
1	He discussed his poverty in his essay "How the Poor Die" and *Down and Out in London and Paris*.	Orwell	0.7	0.1	0.63	0.06
2	The character of Old Major represented Lenin in his allegory *Animal Farm*.	Orwell	0.8	0.2	0.22	0.17
3	For ten points, name this author of *1984*.	Orwell	~~0.9~~ 1.0	0.3	0.05	0.27

Table 5.2: Clue-by-clue question details with model guesses, confidence scores, human and model buzz probabilities, and human-adjusted model scores (SH_t). CALSCORE^2 computes the probability of a system buzzing before the humans correctly answer the question. The resulting score is 0.77.

produces a guess with associated confidence and a probability of buzzing (b_t). Simultaneously, we track the cumulative proportion of human participants who have correctly answered by that time (h_t).

In clue 0, the system guesses incorrectly with low confidence, and only one out of three

humans has buzzed correctly ($h_0 = 0.1$, approximating $1/3$). The human-adjusted score SH_t is zero, as the system's guess is incorrect. In clue 1, the system correctly guesses "Orwell" with higher confidence (0.7), while the cumulative human correct rate increases to $h_1 = 0.1$. Since the system buzzed before most humans had answered correctly, it receives partial credit: $SH_1 = 0.06$.

By clue 2, more humans have answered correctly ($h_2 = 0.2$), and the system again guesses correctly, further increasing the cumulative adjusted score. Finally, in clue 3, both the system confidence (0.9) and buzz probability ($b_t = 1.0$) are high, but a larger proportion of humans ($h_3 = 0.3$) have also answered correctly by this point. As a result, the final score $SH_3 = 0.27$ reflects a lower human-adjusted credit, even though the guess was correct.

The reward for CALSCORE^2 for the question is reported at the bottom of the table as 0.53, with the value in parentheses (0.77) denoting the unadjusted baseline. This illustrates how incorporating human buzz data provides a more realistic measure of model competitiveness under time pressure. In CALSCORE^2, we subtract the reward term from 1 to quantify the model's calibration error.

5.4 Model Calibration Evaluation

GRACE helps to evaluate differences between human and model calibration (§5.4.1). We also validate the dataset's difficulty granularity and discuss calibration errors using our proposed metric (§5.4.2) and qualitative analysis (§5.4.3).

5.4.1 Comparing Human and Model Calibration

Buzz performance. To compare human and model calibration, we first examine when and

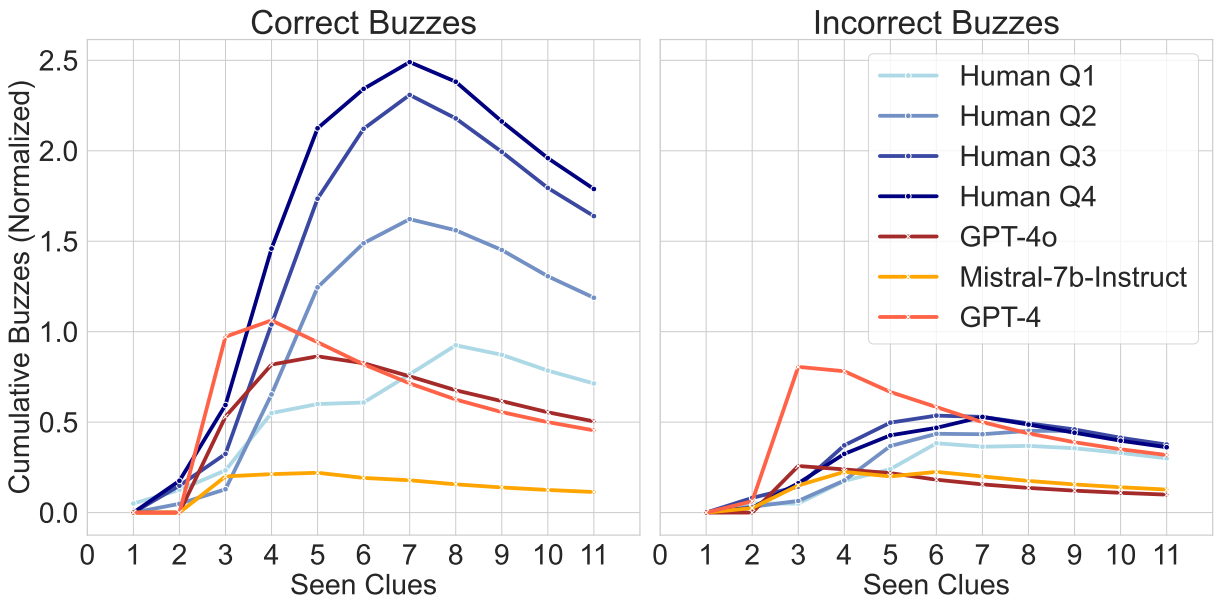

Figure 5.4: Each team's cumulative buzzes (normalized by the number of matches each team participated in). The top quartile of human teams (Q4) achieves the highest cumulative correct buzz rate, peaking over twice as high as the best model. Top human teams are thus more accurate and better-calibrated than models, even as the difficulty changes when more clues are revealed.

whether each team buzzes on the question, as well as the correctness of their answers. Figure 5.4 gives each team's cumulative buzzes over the number of matches each team participated in. The 17 human teams are divided into quartiles, from Q1 (bottom) to Q4 (top), according to their total correct buzzes. Human teams, especially the top quartile, achieve the highest cumulative correct buzz rate (peaking in the middle of the questions), demonstrating their ability to confidently infer correct answers with fewer clues and indicating better accuracy and calibration than models. In contrast, GPT-4 exhibits a moderate cumulative correct buzz rate, which is only briefly higher than the top human teams and lower than 50% of human teams for most of the question. Meanwhile, Mistral-7b-Instruct lags significantly behind all other teams, indicating poor calibration. In addition, GPT-4 and GPT-4o exhibit substantially higher incorrect buzz rates than human teams (right plot). All models, especially GPT-4, are overconfident early in the questions when little information is available: they are **especially miscalibrated relative to humans when the question**

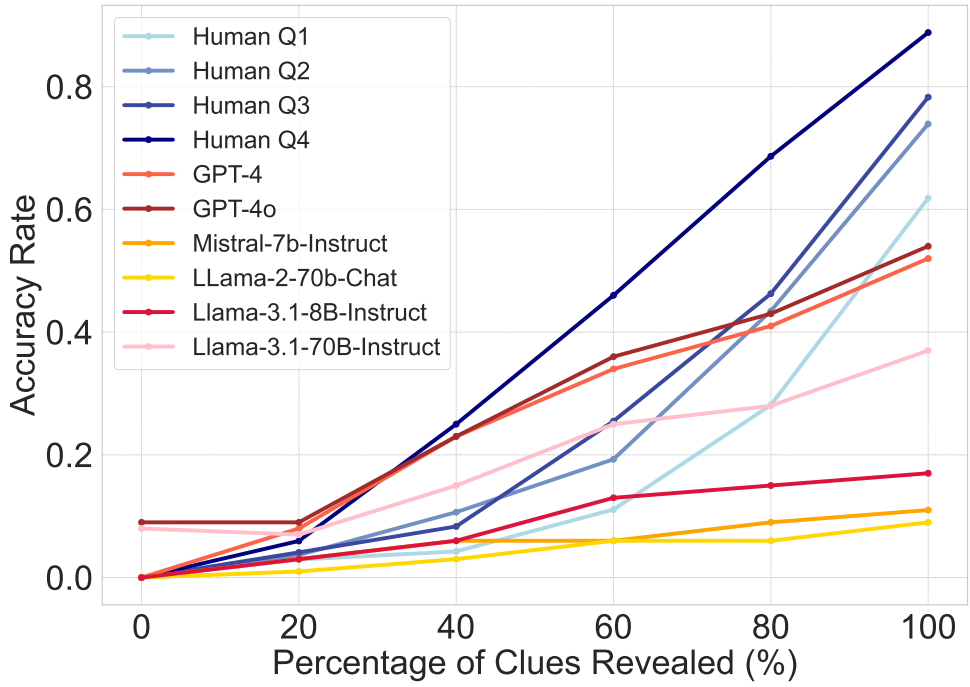

Figure 5.5: Comparison of human and model average accuracy rates as more clues are revealed (whether the team's guess is correct after seeing the first n clues). As more clues are revealed, accuracy improves for both models and humans. Models often answer incorrectly until most clues are provided, and human accuracy increases more rapidly, validating that each instance becomes easier for both humans and models and that most humans can answer correctly by the end.

is still hard. Overall, the models tend to buzz incorrectly more often than humans and correctly less often, indicating **overconfidence in wrong answers and underconfidence in correct ones**.

Difficulty granularity of each question. To evaluate model calibration over a range of difficulty levels for models, we asked the writers to write questions that are easier to answer as more clues are revealed (§ 5.2). To validate this design, we examine model and human correctness as the percent of clues revealed increases. For models, we consider the correctness of a model's guess for the first n clues. The questions in GRACE are appropriately challenging and become easier for models and humans as more clues are revealed (Figure 5.5). Human team accuracy (blue) increases steadily, indicating that question difficulty indeed decreases as clues are revealed for human players. Moreover, even top models like GPT-4 and GPT-4o have under 50% accuracy

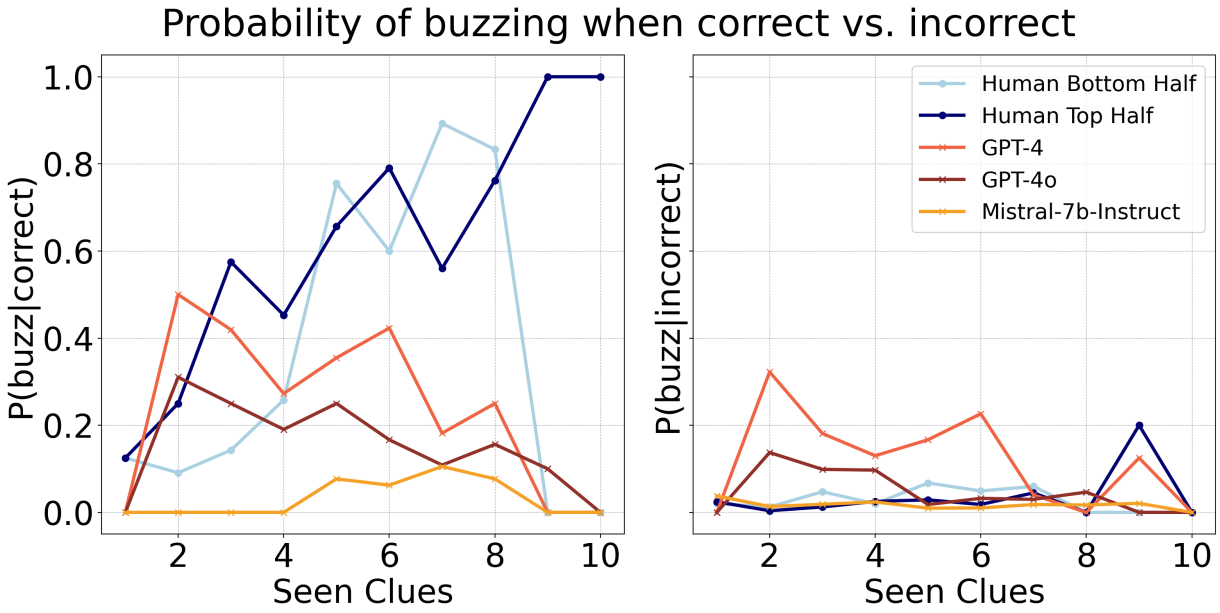

Figure 5.6: Humans are far more likely than models to buzz in when they are correct (left), and typically less likely to buzz in when they are incorrect (right), indicating that models remain miscalibrated relative to humans even when explicitly controlling for accuracy. (Due to the smaller sample size of human buzzpoints in the survey data, we use halves instead of quartiles here.)

until at least 90% of clues are provided, highlighting significant room for improvement on this benchmark. To measure human accuracy per corresponding clue, we used offline human responses (§5.2.2.1 with human quartiles calculated.

Notably, the bottom three quartiles of humans are less accurate than top models for most of the question (Figure 5.5), yet still typically outperform models on maximizing correct buzzes relative to incorrect buzzes (Figure 5.4). This trend suggests that **models' relatively high rate of incorrect buzzes and low rate of correct buzzes is due to miscalibration, not inaccuracy.** We investigate this distinction further in the next section.

Conditional likelihood of correct answers.. While the tournament allows us to observe $P(g = 1 \mid b)$, the likelihood that a team's guess is correct ($g = 1$) when they buzz (b), we also aimed to compare how often teams are confident enough to buzz when correct, $P(b \mid g = 1)$, and when incorrect, $P(b \mid g = 0)$. Using the offline human responses (§5.2.2.1), we estimate

$P(b \mid g = 1)$ for each human team. For each player, we calculate $P_{player}(b \mid g = 1)$: the number of instances when a player buzzed in correctly with n clues revealed, divided by the number of instances when a player's guess was correct with n clues revealed. We then estimate $P(b \mid g = 1)$ for the top and bottom half of respondents as the average of $P_{player}(b \mid g = 1)$ across all surveyed players within each half. Finally, we compare this estimation with $P(g = 1 \mid b)$ for the tested models. We follow the same process to estimate $P(b \mid g = 0)$.[13] The results indicate that even **the strongest models are less confident than humans on correct answers and more confident on incorrect ones** (Figure 5.6). For most questions, all humans are more than 50% likely to buzz when correct, while models remain below 45%, indicating lower confidence in correct answers. Among the models, GPT-4 was most likely to buzz incorrectly, reflecting its confidence in wrong answers.

As clues are revealed, humans become more likely to buzz when they know the correct answer, while models become less likely to buzz. This suggests that seeing more clues strengthens human confidence, but not model confidence.[14]

5.4.2 CALSCORE Analysis

We evaluate the calibration error of six LLMs on GRACE using existing metrics (ECE and Brier scores; Chapter 2.1.2) and CALSCORE (§ 5.3). For each clue in a question, we collect logit-based and verbalized confidences to compute metric scores.

CALSCORE and CALSCORE^2 correlate with ECE and Brier score results (Table 5.3); however, across both confidence elicitation methods, all models display greater error under two

[13] For all estimates, we consider only the guess correctness and buzz statistics up to the point when a player first buzzes, as later guesses do not count in real competitions.

[14] A small fraction of human participants (n=1) has a sharp spike in incorrect buzzes near the end of a question.

Verbalized-based Confidence (Sorted by CALSCORE*)*

Model	Brier Score		ECE		MCE		CALSCORE		CALSCORE^2	
GPT-4	0.274	2	0.259	2	0.584	1	0.588	1	0.878	1
GPT-4o	0.266	1	0.224	1	0.601	2	0.604	2	0.900	2
Llama-3.1-70B-Instruct	0.373	3	0.392	3	0.685	3	0.719	3	0.955	3
Llama-2-70b-Chat	0.490	4	0.570	4	0.739	4	0.803	4	0.961	4
Llama-3.1-8B-Instruct	0.623	5	0.693	5	0.774	5	0.843	5	0.978	5
Mistral-7b-Instruct	0.716	6	0.784	6	0.790	6	0.881	6	0.973	6

Logit-based Confidence (Sorted by CALSCORE*)*

Model	Brier Score		ECE		MCE		CALSCORE		CALSCORE^2	
GPT-4o	0.341	3	0.353	2	0.654	2	0.654	1	0.955	2
Llama-3.1-70B-Instruct	0.323	2	0.339	1	0.651	1	0.679	2	0.939	1
GPT-4	0.380	4	0.388	3	0.672	3	0.684	3	0.962	3
Llama-3.1-8B-Instruct	0.302	1	0.397	4	0.675	4	0.718	4	0.965	4
Mistral-7b-Instruct	0.553	5	0.677	5	0.766	5	0.846	5	0.980	5
Llama-2-70b-Chat	0.774	6	0.829	6	0.825	6	0.921	6	0.992	6

Table 5.3: Models are sorted by CALSCORE. Compared to MCE, CALSCORE offers a more human-aligned assessment of calibration quality.

human-adjusted metrics compared to MCE (CALSCORE without human adjustment). The two metrics capture errors that existing methods overlook: cases where models underperform relative to humans by being confidently wrong or underconfident when correct. Thus, **CALSCORE and CALSCORE^2 capture that models are especially ill-calibrated compared to humans**, and factoring in human performance reveals more room for improvement on LLM calibration. The gap between MCE and the CALSCOREs widens for worse-performing models, suggesting that weaker models are even more miscalibrated relative to stronger models when factoring in human performance. Additionally, CALSCOREs report higher errors than ECE and Brier scores across both confidence elicitation methods for most models, underscoring calibration deficiencies that previous metrics underestimate.[15]

[15]All four metrics use a [0,1] scale; lower is better.

Thus, **CALSCORE and CALSCORE^2 reveal that models are often miscalibrated relative to humans**, particularly in high-stakes or early-decision settings. Unlike traditional calibration metrics, these human-aware metrics expose errors that arise not only from misalignment between confidence and correctness, but also from misalignment with human behavior, which is considerably critical in real-world AI applications where AI systems interact with or make decisions on behalf of humans.

This distinction becomes especially relevant as LLMs approach human-level performance. A model may achieve superhuman accuracy but still exhibit poor calibration if it is confidently incorrect on examples that humans are uncertain about. In such cases, conventional metrics like ECE or Brier score may report low error, while CALSCORE and CALSCORE^2 reveal substantial overconfidence. Conversely, a well-calibrated superhuman model would buzz earlier and more confidently than humans—but only when justified by its correctness—and our metrics would reward that behavior.

Finally, across both logit-based and verbalized probability-based confidence elicitation methods, CALSCORE and CALSCORE^2 consistently report higher error than ECE and Brier scores for most models.[16] This underscores how human-grounded calibration evaluation can surface deficiencies that standard methods miss.

5.4.3 Qualitative Analysis and Model Errors

Miscalibrated instances from CALSCORE. All six models exhibit similar patterns for the questions on which they were most- and least-calibrated under CALSCORE (Figure 5.7). Models did best on questions that mention concrete proper nouns closely associated with the answer,

[16] All four metrics use a [0,1] scale; lower is better.

Q: One thinker's argument that this claim is a "hyperbolic point which ought to be silent" was subject to a response titled "My Body, This Paper, This Fire." Jacques Derrida first coined the word "différance" in a book responding to Michel Foucault's *Madness and Civilization* and partially titled for this statement. In *The Search for Natural Light*, a premise involving doubt was added to this statement. This claim, which Pierre Gassendi criticized for being circular, was presented as an example of a "clear and distinct" idea. For 10 points, name this first principle coined in René Descartes' *Discourse on the Method*.

ANSWER: "I think, therefore I am" [or "cogito, ergo sum"]

Figure 5.7: Sample question on which models are poorly calibrated.

even on obscure topics: for example, a question on Ireland that gives the titles of Irish songs, a question on telomeres that mentions the protein TRF2, and a question on Brooklyn that mentions the neighborhood of Midwood. Models tend to be least-calibrated on questions with multiple plausible answers (e.g., one on fish as a Buddhist symbol, since other animals also have symbolic meanings in Buddhism). Models also struggle on questions that use descriptions instead of titles (e.g. a question that describes music by Maurice Ravel, and one that describes Jewish birth ceremonies).

Qualitative feedback. We survey the human players for feedback on model abilities. Differences in model calibration are visible to the players: several find GPT-4 "too aggressive," while Mistral seems much weaker, often buzzing late in the question. One player notes that the models "obviously knew a lot, but were quite bad at gauging how well they knew something to [buzz]." Others note that models tend to buzz on "more concrete clues" and struggle with multi-step reasoning. For example, a question on Alice Walker mentions her trip to Eatonville to write about local author Zora Neale Hurston. Players note that GPT-4 incorrectly guesses "Zora Neale Hurston," while human players correctly say "Alice Walker."

Players also note that when models were incorrect, they give more "unreasonable" an-

swers than humans do. For example, models incorrectly answer a question on the treatise Philosophical Investigations with "Fermat's Little Theorem" and "*The Lion, the Witch and the Wardrobe*." Guessing an equation and a children's book with high confidence for a work of philosophy suggests serious miscalibration, since either option should be completely outside the realm of possibility; no human players gave answers so distant from the correct one. Other model errors not observed among human players include buzzing before any substantive clues are revealed, answering with a song title for a question asking for a surname, and hallucinating inexistent schools of philosophy. These types of confidently incorrect predictions, especially early in the question when humans have low confidence, lead to large penalties in CALSCORE. For example, a model buzzing at 10% into the question with high confidence but giving an incorrect answer might contribute a score of 0.9 (i.e., 90% error) for that instance in CALSCORE, since the model is confidently wrong while humans have not yet buzzed. This highlights how CALSCORE penalizes overconfident, early mistakes more heavily than traditional metrics.

Model and human strengths between question topics differ greatly. We examine models' and humans' ratios of correct to incorrect buzzes per category. Human players are best at literature, but this is the weakest or second-weakest category for all models. All models did relatively well on science. GPT-4o is much stronger at social science, arts, and science than other categories, and slightly outperforms humans for every category; GPT-4 was worse than the humans for all categories.

All human participants in our competitions were experienced players, but we find that calibration performance varies greatly even among these experts: stronger humans substantially outperform top models, but not all humans do. A general takeaway for future model-human comparisons on tasks involving calibration is that variance in human skill can greatly affect the

outcome of a comparison.

A side benefit of conducting live human-model competitions was a significant degree of community involvement from trivia enthusiasts who were not researchers. In-person data collection, though more involved than crowdsourced data, offers other benefits: we found that participants were attentive and enthusiastic; moreover, in-person data collection (especially "gamified" approaches) raises awareness of and interest in AI.

5.5 Summary

For users to trust LLMs, they need assurance that these models will not confidently produce wrong answers. To address this, GRACE offers a benchmark for fine-grained calibration evaluation, grounded in human calibration. Our analyses on GRACE reveal that models are often miscalibrated relative to well-informed humans. Specifically, model calibration errors came from difficulty with abstract descriptions, far-fetched incorrect guesses, and confidently incorrect answers given few clues. Our new metric, CALSCORE, combined with GRACE, evaluates the performance of six LLMs, revealing significant room for improvement.

GRACE provides a blueprint for developing human-centered improvements to model calibration, particularly in cooperative settings where humans and models interact to make joint decisions. Unlike the competitive setting where calibration errors can lead to early, overconfident buzzing and incorrect answers, cooperative settings open the door to mitigation strategies that can reduce the negative impact of miscalibration. This human-AI cooperation allows for dynamic trust calibration, where AIs can learn to defer, abstain, or adjust their expression of uncertainty based on the skill or preferences of the human user.

GRACE contributes actionable directions in this space: it encourages improving verbalized confidences that support human decision-making, personalizing abstention strategies based on human skill level, and evaluating these interactions through human-model teaming experiments. These interventions offer a path toward making AI systems more responsibly aligned with human expectations and needs.

Chapter 6: VeriLA: A Human-Centered Evaluation Framework for Interpretable Verification of LLM Agent Failures[1]

Finally, in Chapter 6, we move beyond comparing human and model performance to focus on aligning AI behavior with human-defined values. The previous chapter showed that raw model confidence is a poor proxy for accuracy, underscoring the limits of confidence-based evaluation. In this chapter, we turn to uncertainty estimates as a potentially richer signal: by using them as features for training a verifier, we investigate whether evaluation can move closer to human-like judgment, capturing both correctness and the nuanced ways humans gauge reliability.

This approach fits within a human-grounded evaluation framework that emphasizes what users expect from AI agents, underscoring the need for evaluation signals that go beyond raw correctness. Instead of evaluating agents solely by correctness or skill, we assess their behavior in a multi-agent system through the lens of user expectations and values, providing a more holistic foundation for improving human–AI interaction. This shift allows us to evaluate not just whether an AI is correct, but whether it behaves in ways that are useful and trustworthy in context.

The proposed framework begins by applying user-designed criteria to the system planner in order to identify agent failures relative to human standards. Building on this, we collect binary

[1]Yoo Yeon Sung, Hannah Kim, and Dan Zhang. 2025. VeriLA: A Human-Centered Evaluation Framework for Interpretable Verification of LLM Agent Failures. In *Human-centered Evaluation and Auditing of Language Models workshop (HEAL) of ACM Conference on Human Factors in Computing Systems*.

human annotations of each agent's decision and train an external model to predict these failures. By incorporating agent-level features—user-specified evaluation criteria, agent confidence scores, and structural information from the planner—the model offers interpretable predictions aligned with user-defined notions of correctness. This granular, user-centric approach not only detects agent failures as perceived by humans but also yields actionable feedback grounded in the specific criteria that users care about, supporting more transparent and value-aligned AI systems.

6.1 Motivation

As large language models (LLMs) continue to excel across various fields, they are increasingly used to address complex reasoning tasks through LLM-as-agent systems (Xi et al., 2025; Wang et al., 2024a). A key application is a LLM-based compound AI system, where a planning agent breaks a complex task into simpler subtasks, and delegates these subtasks to multiple specialized LLM agents (Wu et al., 2022a; Zhang et al.; Zaharia et al., 2024). Each assigned agent must accurately execute its subtask, as the final agent's output heavily relies on the previous agents' outputs and is considered as the final solution of the overall task (Cheng et al., 2024). Failures in any agent can propagate and cause the overall task failure (Sumers et al., 2023; Jaeger et al., 2013). While these compound systems demonstrate strong problem-solving capabilities, they face significant limitations in that they may produce outputs that contradict human expectations, often requiring human intervention for failure feedback and revision (LangChain, 2013; Arawjo et al., 2024). However, providing feedback is challenging because agent outputs come from reasoning that deviates from humans or lack clarity on the cause of failure, hindering users from providing guidance on remedying execution failures. Moreover, manually reviewing each step is labor

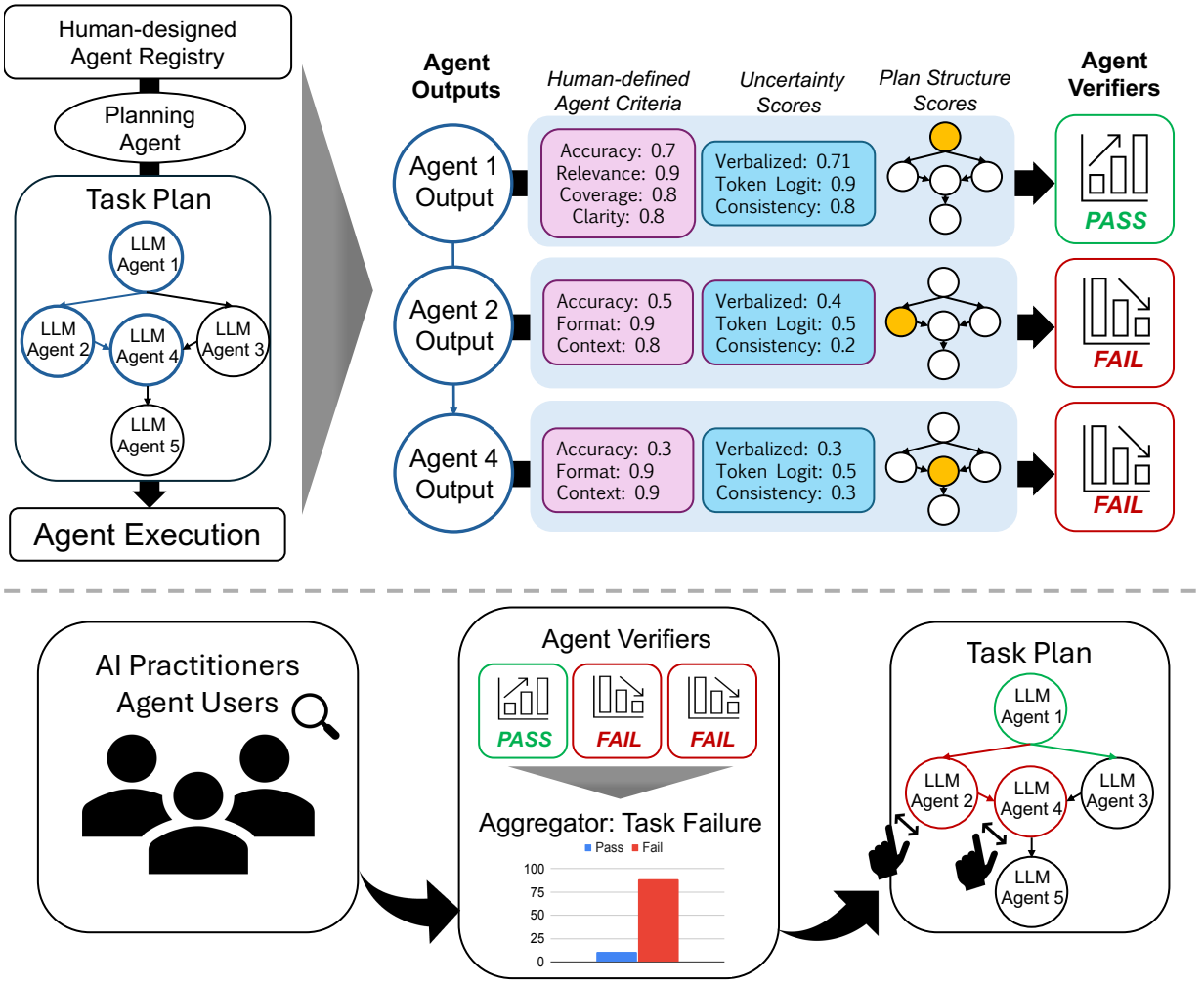

Figure 6.1: Overview of VeriLA. Our framework operates in three main stages (1) planning where a planning agent decomposes a task into subtasks using a human-designed agent registry and generates a plan graph; (2) agent execution where specialized LLM agents perform the subtasks; and (3) execution verification, which verifies each LLM agent's outputs based on human-defined agent criteria, agent uncertainty, and dependency information from the plan structure. We then assess task failure with aggregation metrics that combine verifier scores. Our framework guides users to detect task failures efficiently, identify faulty agents, and analyze the root causes of their failure.

intensive and not scalable. It forces users to audit each agent's execution outputs, increasing the risk of errors. This underscores the need for more systematic and efficient methods for auditing and supervising compound AI systems with LLM agents.

To address this challenge, our framework consists of three interconnected components: (1) **planning**, where a planning agent decomposes the target task into simpler subtasks based on a predefined agent registry, generating a graph-based plan. The agent registry provides each agent with clear role assignments and execution guidelines that adhere to human expectations; it defines the available agent types, specifying for each agent its role and capabilities, and output format; (2) **agent execution**, in which LLM agents sequentially perform their designated subtasks, with success determined by their adherence to the predefined roles; (3) **execution verification, where a verifier module automatically assesses each agent output to ensure its success in fulfilling its assigned subtask.** Our verifier incorporates human-centered judgments on agent outputs along with its relationship with other agents: an agent's subtask type, scores based on human-defined agent criteria, agent uncertainty, and agent's dependency within the plan structure.

In addition to verifying individual agent outputs, VeriLA evaluates whether the **overall multi-agent plan achieves the correct final answer**. It introduces a task-level metric that aggregates verifier results to identify when task failure stems from specific agent errors.

For example, consider a math problem (Figure 6.2): "Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?" A planner decomposes this into: (1) Identify operands from the question (2) Subtract number of egges from breakfast from number of eggs laid per day (3) Subtract number of eggs used for baking from remaining eggs after breakfast and (4)

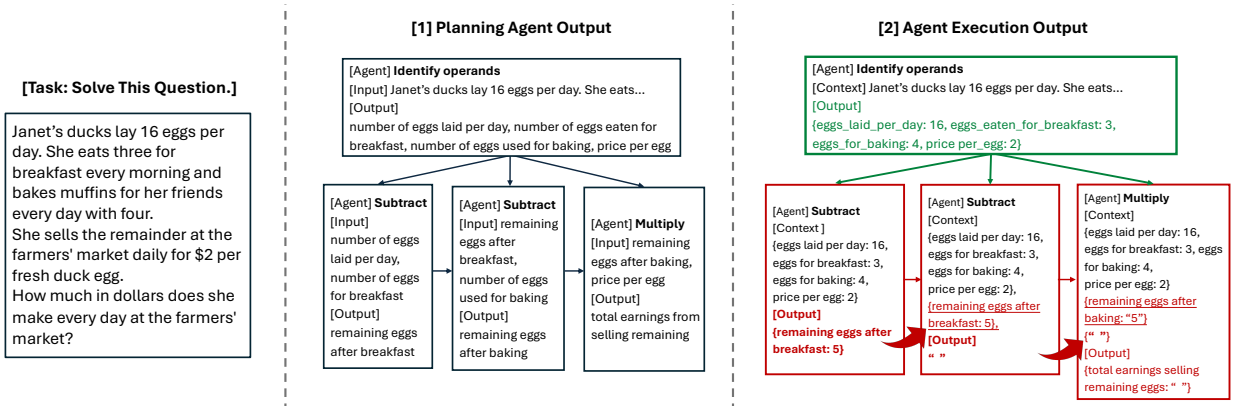

Figure 6.2: Example of agent's failure propagating to overall task failure. For example, based on the generated plan from the planning agent, each agent should accurately execute their subtasks. The first "subtract" agent failed to calculate the remaining eggs, causing subsequent "subtract" and "multiply" agents to lack the necessary context for a successful execution (three red boxes). An agent-specific verifier can help users trace the error propagation, identify the root cause of the error, and understand how it led to the task failure.

Multiply remaining eggs after baking and price per egg.

If the first agent incorrectly outputs 5 instead of 13, the final answer will be wrong. VeriLA identifies this failure and attributes it to the first agent. This approach supports **goal-based verification and error attribution**, moving beyond accuracy to structured, human-aligned evaluation.

We introduce a metric that aggregates verifier results across agents to identify failed tasks due to agent failures. By leveraging the plan's graph structure, it enables targeted analysis of verifiers and failure criteria, streamlining failure detection. To evaluate our framework, we conducted a case study on a complex reasoning task—solving mathematical reasoning problems—demonstrating its effectiveness and practical applicability to AI practitioners.

In summary, VeriLA enables detailed auditing of agentic systems to ensure transparency in agent failures, reducing manual review efforts, and strengthening trust between users and compound AI systems. Beyond merely automating the validation of each agent's success, we hope for a collaborative problem-solving between human and LLM agents by tailoring agent workflow to human needs.

6.2 VeriLA: Framework for Verifying LLM Agents in Compound AI Systems

We propose an evaluation framework that aims to assist users in auditing and interacting with compound AI systems. During users' inspection of overall task failures, they can detect each agent's execution failures, and quickly and clearly understand the reasons behind the failure. This enables them to provide actionable suggestions for planning or execution revisions.

6.2.1 Planning

In a compound AI system, a planning agent's role is to decompose a task into a sequence of subtasks, assign them to specialized agents—which are typically predefined in the system's agent registry—and generate a plan to solve the task.[2]

To curate an agent registry that aligns with the human reasoning process, we first ask AI practitioners to curate a system's agent registry tailored to the target application. They register each agent with a specific role, along with its expected input and output. For example, in a math reasoning task, an "Add" agent is responsible for summing given operands, where the input is a list of numbers, and the output is a single sum (Table 6.1).[3] Then, they identify the most common agents and craft their roles, inputs, and outputs. Using this agent registry, the planning agent decomposes a given task into subtasks and delegates them to appropriate agents (Upper left in Figure 6.1). This later helps users in diagnosing each agent's failure; it informs them whether the agent was executed appropriately based on its role and has a proper output format.

Then, our planning agent generates a plan with a directed acyclic graph (DAG) format,

[2]The planning agent is not registered in agent registry and thus does not participate in the plan.

[3]To guide practitioners on the necessary agents and their required functionalities, we use Chain-of-Thought (CoT) prompting to generate a pool of agent candidates. We provide the AI practioners to examine 50 samples resulting from one-shot promoting, and determine which agents are adequate for math reasoning application.

where each node represents an agent, and directed edges show input-output dependencies between the nodes (example in Appendix 8.1). This graph-based planning ensures that the task complexity is decomposed into an interrelated sequence of simplified subtasks. Then, each subtask is assigned to an appropriate agent with specific inputs and outputs, mirroring how humans often approach complex tasks (Ghallab et al., 2004). This decomposition reflects a natural progression from earlier chapters: while Chapters 3 and 5 focused on single-turn QA tasks and calibration errors in isolation, here we shift to evaluating multi-step reasoning tasks that demand structured coordination among multiple agents. By assigning each subtask to a specialized agent with defined inputs and outputs, our framework mirrors how humans approach complex problems—delegating responsibility, verifying intermediate steps, and aligning behavior with shared expectations (Ghallab et al., 2004).

Planning agents also generate instruction prompts for each agent, reflecting its role and input–output format. These prompts should be closely tied to the original task, reducing the risk of agents hallucinating or forgetting the trajectory needed to solve the task.

6.2.2 Agent Execution

Agents execute assigned specific subtasks and instruction prompts from the generated plan. Additionally, they receive relevant context information as input which includes relevant outputs produced from the preceding agents. Because agents can fail during execution, users must intervene to correct errors and prevent an agent's failure from propagating to overall task failure. For example, in Figure 6.2, as the first agent incorrectly computed, a human annotator can step in after the first step, flag the mistake, and correct the value to 13. This correction is then used as the input for the next agent, allowing the overall task to succeed despite the initial failure. Thus,

human annotation in this setting is not just for labeling final answers but debugs the reasoning process, ensuring alignment between agent execution and user expectations.

However, manually auditing whether an agent has fulfilled its subtask is both cumbersome and cognitively demanding for users. Thus, we introduce a agent-specific verifier in the next section.

6.2.3 Execution Verification by Human-Aligned Agent Verifier

Our agent verifier autonomously evaluates agent executions and flags potentially incorrect results, enabling users to focus only on the problematic agents within the plan. Although some self-verifying agents exist (Sun et al., 2023; Madaan et al., 2024), we question their reliability (Stechly et al., 2024), lack of contextual understanding (Prasad et al., 2024), and insufficient human alignment (Goyal et al., 2024).

To address these issues, we build a separate evaluation module that can verify each agent's execution failures. This module functions as a binary classifier, trained on multiple features such as human-defined agent criteria, agent uncertainty, and plan structures with subtask types. We use GPT-4o for all experiments, employing a temperature setting of 0.7 for consistency-related metrics to obtain as diverse answers as possible, and a temperature of 0.1 for the remaining experiments (Xiong et al., 2024b). To balance verifier's autonomy with human-designed criteria, we integrate external LLM judge scores that are assessed with human-defined criteria:

- **Per-criteria scores by LLM judges** These features are introduced to measure the successfulness from human perspective (details in Table 8.2). We prompt an LLM to score the execution results based on these predefined criteria per subtask. For example, the execution

result of an “add” excutor, "9 apples", might be evaluated with the following binary scores: {"accuracy of numerical value": 1.0, "sufficiency of context information": 0.0, "adherence to format": 1.0}. By assessing the correctness of the LLM executor’s outputs and using these evaluations as features in training a verification agent, we ensure that the agent’s detection is grounded in reliable, human-aligned guidelines.[4]

These criteria are carefully tailored for each agent to ensure that expectations align with human standards. This way, our verifier ensures that each agent aligns with human expectations and is evaluated according to users’ specific needs (Liao and Xiao, 2023). For instance, the “Subtract” agent’s execution is evaluated not only on the accuracy of summation but also on its adherence to the expected format (e.g., number) and sufficiency of context information for the subtask (Figure 6.2). Grounding agents’ evaluation in predefined criteria permits objective judgment and clarifies failure reasons.

Additionally, we integrate three LLM uncertainty estimation techniques: verbalized confidence, logit-based confidence, and confidence based on self-consistency. This is under the assumption that lower certainty often correlates with a higher likelihood of errors or deviations. This builds directly on the uncertainty modeling approaches introduced in Chapter 5, where we showed that a model’s confidence, whether expressed explicitly through language or derived from logits, can be informative but often miscalibrated. Despite these limitations, confidence still provides useful *relative* signals about reliability, especially when aggregated across agents or supplemented with additional cues like self-consistency. For example, an agent that expresses low verbalized confidence or produces highly inconsistent outputs across generations can be flagged

[4]Given that each subtask has a varying number of criteria, we create a one-hot vector to indicate the specific subtask to which each sample refers. This vector is then concatenated with a matrix containing the union of criteria columns across all subtasks, populated with the corresponding binary values from the execution result.

as uncertain and potentially inaccurate. Rather than treating confidence as an absolute indicator of correctness, we use it as a heuristic for identifying points of potential failure that may warrant human attention or intervention.

- **Verbalized confidence** (Xiong et al., 2024b) reflects how confident the executor module is about its output, often derived from explicit confidence scores or qualitative indications of certainty (e.g., "0.7").

- **Logit-based confidence** (Huang et al., 2023b) reflects the average of the exponentials of the token log probabilities (LP):

$$LP_{avg} = \frac{1}{N}\sum_{i=1}^{N} p(s_j \mid x_i) = \frac{1}{N}\sum_{i=1}^{N} \exp(\log p(s_j \mid x_i)), \tag{6.1}$$

  where N is the number of tokens and $\log p(s_j \mid x) = \sum_{i=1}^{j} \log p(s_i \mid s_{<i})$, where s_i is the i-th output token and $s_{<i}$ denotes the set of previous tokens. We denote $\log p(s_j \mid x_i)$ as token log probability.

- **Softmax-based confidence** reflects the average of softmax values across the generated tokens, providing a measure of the overall uncertainty of the model based on the top-k token log probabilities:

$$Softmax_{avg} = \frac{1}{N}\sum_{n=1}^{N} \frac{\exp(\mathbf{z}_n)}{\sum_{i=1}^{k} \exp(\mathbf{z}_{n,i})}, \tag{6.2}$$

  where $\mathbf{z}_n$ is the token log probability and N is the number of tokens.

- **Entropy-based confidence** (Huang et al., 2023b) reflects the average of entropy values across the generated tokens, providing a measure of the overall uncertainty of the model

based on the top-k token log probabilities:

$$Entropy_{avg} = \frac{1}{N}\sum_{n=1}^{N}\left(-\sum_{i=1}^{k}\frac{\exp(\mathbf{z}_{n,i})}{\sum_{j=1}^{k}\exp(\mathbf{z}_{n,j})}\log\left(\frac{\exp(\mathbf{z}_{n,i})}{\sum_{j=1}^{k}\exp(\mathbf{z}_{n,j})}\right)\right), \tag{6.3}$$

where $\mathbf{z}_n$ is the token log probability and $\mathbf{z}_{n,i}$ is the i-th top log probability.

- Features from the a separate LLM evaluator capture the uncertainty in its assessment of the initial LLM execution.
  - **Verbalized confidence from external LLM evaluator** is attained directly from the external evaluator's generated response.
  - **Logit-based confidence from LLM evaluator** is attained by the exponential of the logit value of the LLM evaluatorś verification assessment (e.g., logit value of the `TRUE`).
- Features using self-consistency (Wei et al., 2022) technique; we run the same prompt five times and aggregate the confidence values in the following ways:
  - **Self-consistency (Type A): frequency** (Yona et al., 2024) measures the confidence of the executor module by the degree of agreement among the candidate outputs and integrates the inherent uncertainty in the model's output (Xiong et al., 2024b):

$$\text{Confidence}_{freq} = \frac{1}{M}\sum_{i=1}^{M}\mathbb{1}_{\{\text{agreement}(\hat{Y}_i,\tilde{Y})>\theta\}}, \tag{6.4}$$

where $\mathbb{1}_{\{\text{condition}\}}$ is the indicator function that returns 1 if the candidate answer $\hat{Y}_i$ is consistent with the initial execution result $\tilde{Y}$ based on the agreement threshold θ, and

0 otherwise. Here, M denotes the number of candidate answers, and θ is the threshold for agreement. We use the answer equivalence package PEDANT (Li et al., 2024c) with their recommended threshold of 0.5 to assess the agreement.

– **Self-consistency (Type B): verbalized confidence** (Xiong et al., 2024b) measures the average verbalized confidence among the subset of candidate answers identified as correct.

$$\text{Confidence}_{verb} = \frac{\sum_{i=1}^{M} C_i^{verb} \cdot \mathbb{1}_{\{\text{correctness}_i=1\}}}{\sum_{i=1}^{M} \mathbb{1}_{\{\text{correctness}_i=1\}}}, \tag{6.5}$$

where M is the total number of candidate answers and C_i denotes the verbalized confidence of candidate answers.

– **Self-consistency (Type C): logit-based** measures the average logit-based confidence among the subset of candidate answers identified as correct.

$$\text{Confidence}_{log} = \frac{\sum_{i=1}^{M} C_i^{log} \cdot \mathbb{1}_{\{\text{correctness}_i=1\}}}{\sum_{i=1}^{M} \mathbb{1}_{\{\text{correctness}_i=1\}}}, \tag{6.6}$$

where M is the total number of candidate answers and C_i denotes their average log probabilities.

Finally, we include each agent's subtask type and its position within the plan's DAG structure, by using subtask categorical encoding and features like the number of preceding nodes to capture dependency relationships between agents:

- **Subtask type** is represented as one-hot encoding for all subtasks in our taxonomy.
- **Features on Plan Structures**

- **Number of preceding subtasks** is the number of previous subtasks that this subtask is depending on (i.e., in-degree). We incorporate this feature to reflect the dependency information between subtasks within the plan.
- **Source distance** measures the shortest chain length to reach the current subtask as a proxy for node importance.

These structural features enhance our verifier's ability to assess execution reliability within the overall task.

Ground Truth to Train Agent Verifier. We use human-annotated labels as ground truth to train our agent verifier. These labels indicate whether a given agent output satisfies its assigned subtask, based on human expectations. By learning patterns in agent behavior and associated execution features from this labeled data, the verifier predicts whether a given agent is likely to succeed or fail in future runs.

These annotations are collected as part of a structured evaluation process in our multi-agent setting. For each completed task, human annotators assess the correctness of individual agent outputs by referencing: the original user-defined task goal (e.g., a GSM8K math problem), the graph-based plan and the agent's specific subgoal (from the planner), and the agent registry, which defines the expected capabilities, role, and output format of each agent type. They are provided with the **same criteria used by LLM judges**, such as correctness, completeness, and format consistency to provide respective scores for each criterion.

Annotators are provided with full execution context (including prior agent outputs if needed) and explicit instructions for how to determine whether an agent's output satisfies its subgoal.

6.2.4 Aggregation Metrics for Overall Task Failure Prediction

To support human-agent interaction, we guide users in identifying task failures when decomposed and executed by agents. Our verifier predicts potential failures and provides confidence scores for each agent's execution, which we use to assess overall task success. We propose several aggregation metrics to aggregate individual verifier scores. The metric scores represent the likelihood of overall task success, with higher scores indicating a greater chance of success.

We begin with simple methods such as selecting the lowest score among all subtasks (*min* aggregator), highlighting the weakest agent execution; and computing the arithmetic average of all scores (*mean* aggregator), providing a balanced view of subtask performances.

We consider the relative importance of agents within the plan structure, as overall task performance depends on how effectively an agent's output transfers to others. To capture this, we propose two structural metrics for weighting agent scores. The distance-based metric emphasizes agents positioned closer to the source or sink nodes in the plan graph, under the assumption that these agents play a more critical role in the execution flow. This metric weights each agent's score inversely proportional to its distance from either the source or sink node (*source distance* or *sink distance* aggregator), ensuring that agents with greater structural influence have a higher impact on the aggregated score. Given the overall task T consisting of subtasks fulfilled by agents $\{S_1, S_2, \ldots, S_m\}$, we denote the i-th agent S_i's verifier score as $\hat{y}_i$, which is predicted based on features extracted from its execution outputs.

$$\text{AggScore}_{\text{dist}}\left(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}m\right) = \frac{\sum_{i=1}^{m} \frac{\hat{y}_i}{d_i}}{\sum_{i=1}^{m} \frac{1}{d_i}}, \tag{6.7}$$

Agent	Role	Input	Output	Output Format
Identify Operands	Identify operands with text description of each operands	Math question	List of operand names with their values	{<name>: Number, ...}
Add	Add numbers or dates	List of operands	One summed value	Number or Date
Subtract	Subtract numbers or dates	List of operands	One subtracted value	Number or Date
Multiply	Multiply numbers	List of operands	One multiplied value	Number
Divide	Divide numbers	List of operands	One divided value	Number
Filter	Filter a list based on a condition	List, condition	Filtered list	List
Sort	Sort a list by an attribute	List, attribute	Sorted list	List
Convert Format	Convert input from one format to another format	Text, format	Formatted text	Text
Date Lookup	Identify year, month, and day from a natural language description	Text	Date	Date

Table 6.1: Human-designed agent registry for mathematical reasoning tasks. To guide practitioners on the necessary agents and their required functionalities, we use Chain-of-Thought (CoT) prompting to generate a pool of agent candidates. Then, they identify the most common agents and craft their roles, inputs, and outputs. Using this agent registry, the planning agent decomposes a given task into subtasks and delegates them to appropriate agents.

where d_i denotes the shortest path distance from agent S_i.

Similarly, the degree-based metric assigns higher weights to agents with greater connectivity, reflecting their influence within the overall plan structure. It weights each agent's score based on the indegree or outdegree (*indegree* or *outdegree* aggregator) of its node:

$$\text{AggScore}_{\text{deg}}\left(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}m\right) = \frac{\sum_{i=1}^{m} \text{deg}_i \cdot \hat{y}i}{\sum_{i=1}^{m} \text{deg}_i}, \tag{6.8}$$

where $\hat{y}_i$ represents the prediction score for subtask S_i, and deg_i denotes the degree (either indegree or outdegree) of subtask S_i in the plan DAG.

We later assess the accuracy of the aggregated verification results by comparing the task's gold label, indicating actual plan success, with the final agent label predicted by the verifier. This comparison evaluates how effectively the verifier is aggregated to predict task failure (§ 6.3.3).

6.3 Case Study: Mathematical Reasoning

We demonstrate the effectiveness of VeriLA on mathematical reasoning tasks. Math reasoning problems can be naturally decomposed into step-by-step plans, allowing us to focus on evaluating agent execution failures while ensuring that the results are easily verifiable by humans.

6.3.1 Experiment Setting

6.3.1.1 Datasets

We evaluate our pipeline on four math reasoning datasets: GSM8K (Cobbe et al., 2021) and Date Understanding (C3), Multi-Step Arithmetic (C11), Object Counting (C13) from BIG-Bench Hard (BBH) (Suzgun et al., 2023). The GSM8K dataset consists of grade-school-level math word problems written in natural language, whereas the Multi-Step Arithmetic dataset is presented in equation format and involves more complex problems that require multiple steps. Date Understanding focuses on reasoning about and performing operations with dates, while Object Counting involves enumerating objects of interests.

6.3.1.2 Planning and Agent Execution

For each task, our planning agent generates a DAG plan to solve it using a human-designed agent registry (detailed in Table 6.1). We manually filter out instances where the generated plans are invalid,[5] keeping only those with valid plans. The average number of subtasks per plan is four for GSM8K, 3.5 for Date Understanding, 2.5 for Object Counting, and 5.6 for Multistep

[5]We consider both structural validity (e.g., missing dependencies between subtasks) and semantic correctness (e.g., incorrect agent assignment or faulty instructions).

Arithmetic. Each subtask in a plan is then executed by an assigned agent. Both the planning agent and agents in agent registry use GPT-4o with a temperature of 0.1. All prompts for planning, execution, and verification are provided in Appendix 8.2.

6.3.1.3 Agent Execution Verfication

Gold (Execution Failure) Label Annotation. To train the verifier, human annotators label agent's execution failures based on agent criteria, expected inputs and outputs, and input information (§ 6.2.3). For GSM8K, we collected 1,975 subtasks from 497 tasks, each subtask labeled by three annotators from crowdsourcing. We used MTurk platform to recruit crowdworkers for labeling the GSM8K dataset. They label whether an agent's execution has failed, based on its assigned criteria, expected inputs and outputs, and available context information (§ 6.2.3). The annotation task involves assessing whether the agent's response correctly fulfills its subgoal, given what it was instructed to do.

For example, consider a math problem decomposed into subtasks, where one agent is assigned the subgoal: "Multiply 7 by 8." The input provided is the number 7 and the instruction to compute 78. If the agent outputs "56," the annotator marks this as correct. If the agent instead responds with "14" or "I'm not sure," the annotator marks this as a failure (Example in Figure 8.1). For each instance, three labels were collected from workers who passed the qualification test. Due to low inter-rater reliability (Fleiss' kappa: 0.41), only unanimously labeled samples (973 subtasks) were retained for training.[6] For the BBH datasets, the three authors handled annotations

[6]Unlike subjective judgments about deception or intent (Chapter 4), agent verification in our setting is grounded in well-defined task structures, such as whether a numerical computation or string-matching operation was executed correctly. Disagreements here typically signal confusion or inattentiveness, rather than divergent interpretations of a fuzzy concept.

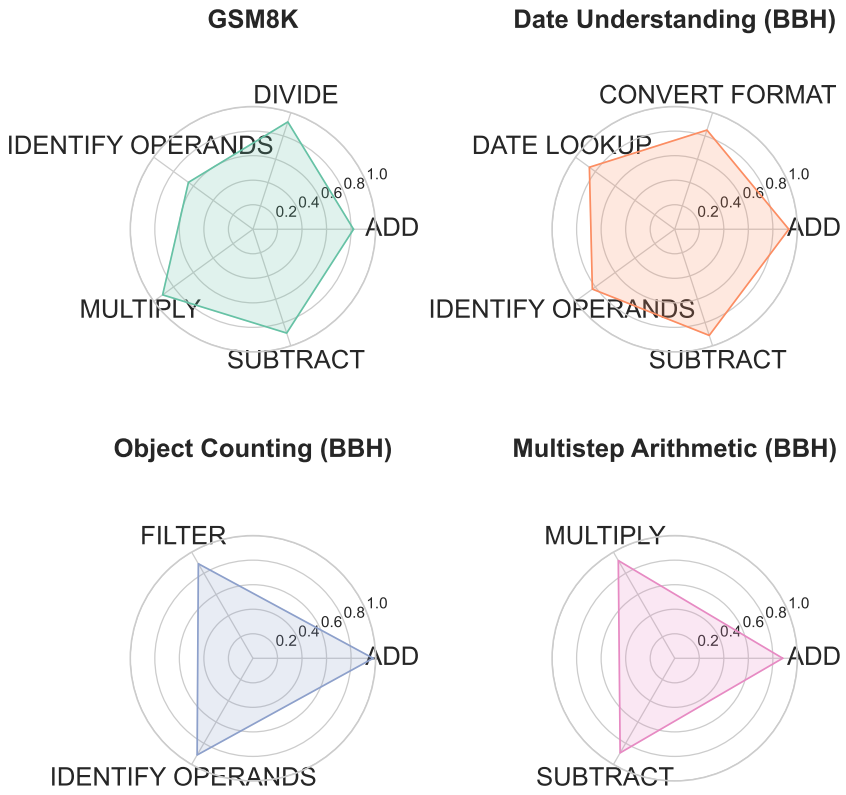

Figure 6.3: Verifier accuracy across datasets. The test accuracy remains consistently high across subtasks, without bias toward any specific one. Similar subtasks, like "Add" and "Subtract," which share the same criteria, also show comparable accuracies across all datasets.

to ensure quality while reducing costs.

Feature Collection. We extracted 26 execution features as described in § 6.2.3. For self-consistency, we used the same model as the corresponding execution agent with a temperature of 0.7 to generate diverse outputs, following Xiong et al. (2024b). For external LLM judges, we employed GPT-4o with a temperature of 0.1.

6.3.2 Verifier Results for Agent Failures

Using features from all agent outputs in the plan, we train a simple machine learning model, Random Forest model, which achieve a high average accuracy (0.88%) across four datasets.,[7] suggesting the verifier's effectiveness in identifying failed agent executions.

Next, we investigate whether its performance varies across different agents. As shown in Figure 6.3, the test accuracy remains consistently high across various subtasks except "identify operands" in GSM8K where agents often struggle with accurate formatting. Moreover, similar subtasks—such as "Add" and "Subtract", which share the same subjective criteria—exhibit comparable test accuracies across all datasets, suggesting their generalizability to other tasks.

To further analyze verifier behavior, we conduct an ablation study with different features to predict agent execution failures. Our verifier achieves the highest performance with all features included while excluding any single feature leads to a drop in performance (Figure 6.4). This suggests that the features provide complementary information, each playing a distinct role in model accuracy. Notably, the agent criteria features has the largest performance drop when removed, suggesting that incorporating human-defined agent criteria from external LLM judges improves the verifiers' ability to align their predictions with human priorities. On the other hand, the consistency-related features had minimal impact on performance, implying they may be less critical.

[7]We compared several ML models and select the one with the highest accuracy. We train agent verifiers separately for each dataset using the following machine learning models: Logistic Regression (Cox, 1958), SGD Classifier (Bottou, 2010), Decision Tree (Breiman et al., 1986), Random Forest (Parmar et al., 2019), AdaBoost (Schapire, 2013), XGBoost (Freund and Schapire, 1997), Gaussian Naive Bayes (GaussianNB) (John and Langley, 1995), and Multi-Layer Perceptron (MLP) (Rumelhart et al., 1986). Among all tested models, the random forest classifier with 100 tree estimators achieved the highest average accuracy among other models in four datasets (0.88).

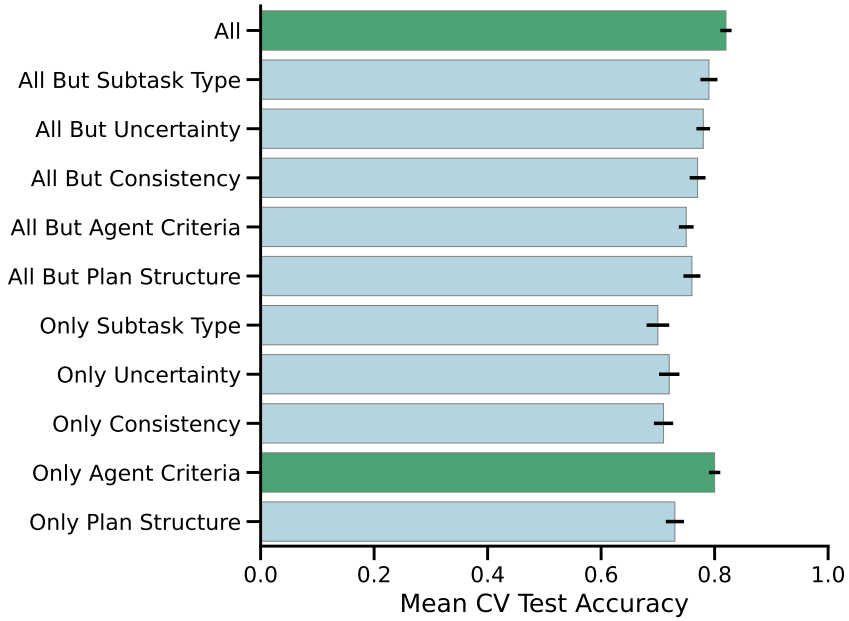

Figure 6.4: Ablation study on different feature configurations evaluating verifiers' test accuracy. Human-defined agent criteria feature enhances its performance, showing the highest accuracy when all features are used.

6.3.3 Aggregator Results for Overall Task Failures

To predict whether the overall task is failing, we present a few aggregation metrics (aggregator), which combine subtask verification scores, that function as a *task-level verifier*.[8] By leveraging the aggregated score from the task verifier, users can prioritize and investigate tasks that are most likely to generate false LLM outputs. They can then analyze the reasons behind a task's failure by examining the flagged subtasks identified by the agent verifier.

Figure 6.5 shows task-level verification performance from different aggregation methods, with lower score indicating a greater chance of failure. x-axis represents percentiles of ranked aggregation scores, and y-axis shows the cumulative ratio of detected failures within each percentile relative to all failed tasks. The curves closer to the top-left corner indicate better performance.

[8] A task is considered successfully solved if the last step's execution output is linguistically equivalent (Li et al., 2024c) to the gold answer from the original dataset.

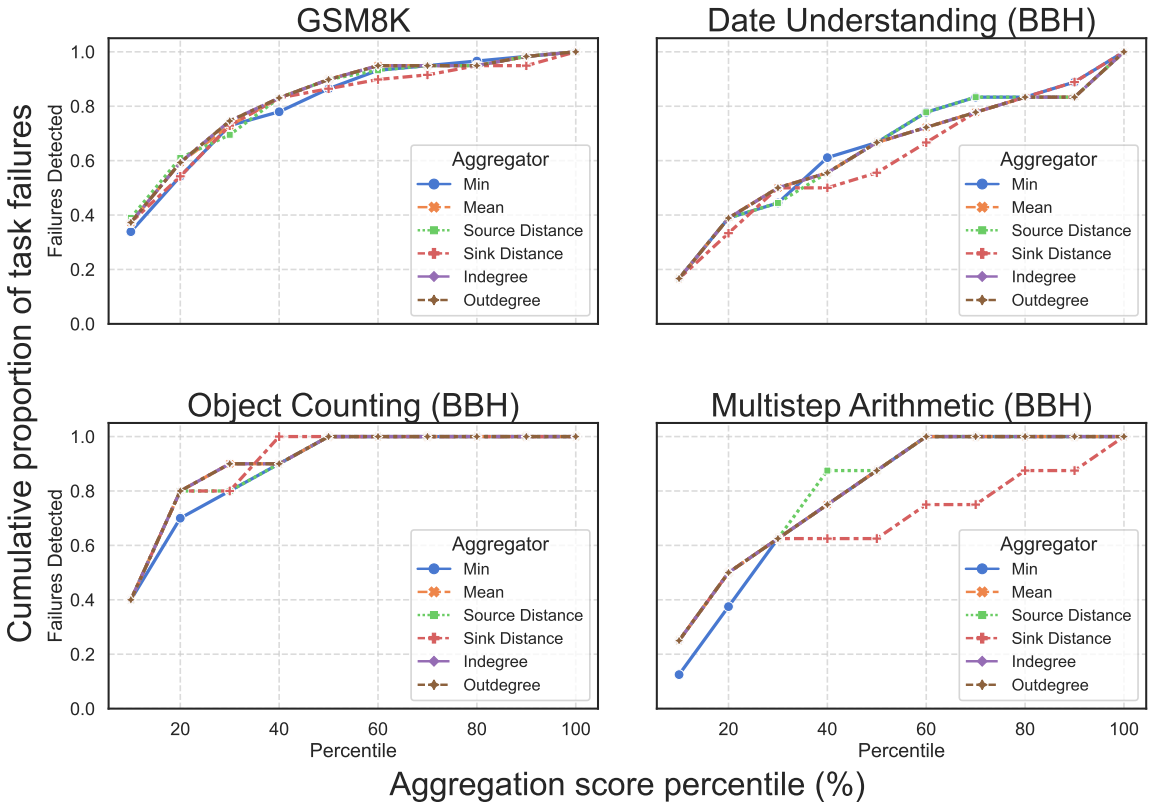

Figure 6.5: Aggregation performance measured by failure rate across aggregation score percentiles. They all show an upward trend, suggesting that they can help users prioritize tasks more likely to fail, when the labor budget is limited, allowing auditing of high-risk tasks first. Overall, *mean* and *outdegree* showed stable performance across datasets and can be used as default aggregation metrics for new datasets.

Sink distance aggregator identified all failures fastest in object counting but was slowest in date understanding. *Source distance* aggregator generally outperformed *sink distance*, except in GSM8K. This suggests that, for GSM8K, proximity to the starting nodes is a stronger indicator than proximity to the final node; GSM8K's typical first subtask—identifying operands—is a common source of error. Although no single aggregator consistently outperforms others across all datasets, they all show an upward trend, suggesting that they can help users prioritize tasks more likely to fail. This can be especially useful when labor is limited, allowing auditing of high-risk tasks first. Overall, *mean* and *outdegree* showed stable performance across datasets and can be used as default aggregation metrics for new datasets.

6.4 Summary

In this work, we introduce VeriLA, a human-centered evaluation framework that verifies agent execution failures; this encourages reliability and interpretability in humans using compound AI systems. By applying a verifier that assesses each agent's outputs through a combination of human-defined criteria, agent output uncertainty, and agent dependencies within the plan. Thus, VeriLA also captures error propagation within the plan, facilitating better human-agent interaction. We also present a verifier-driven task failure metric that help users detect tasks prior to their auditing. Thus, VeriLA enhances accountability on agent performance and labor efficiency by enabling granular human inspection of failing agents and the underlying reasons for their failures.

VeriLA thus brings together themes from earlier chapters—gap between model and human expectations and calibration—and operationalizes them in a multi-agent context. In the next and final chapter, we reflect on the broader implications of the findings that were discussed until now, discussing challenges in human-AI evaluation and outlining key directions for building more trustworthy, user-aligned AI systems.

Chapter 7: Conclusion and Future Directions

7.1 Conclusion

This dissertation proposed a human-grounded framework for evaluating and improving NLP systems, addressing the gap between benchmark performance and real-world user needs. As LLMs continue to be integrated into high-stakes settings, current evaluation methods, which focus on accuracy or benchmark performance alone, are insufficient. This dissertation tackled this challenge across four main directions: creating human-grounded adversarial benchmarks, capturing subjectivity in natural language tasks, aligning model calibration with human judgment, and detecting errors in multi-agent systems using user-defined standards.

In Chapter 3, we introduced an adversarial dataset generation framework for constructing adversarial examples that reveal human–model performance gaps. Using IRT and expert-authored QA data, we proposed a metric that measured adversarialness based on the gap between human answerability and model failure. This method allows benchmark designers to systematically construct examples that remain challenging and relevant over time, especially as models improve. By targeting cases that humans find easy but models find hard, we avoid benchmark saturation and create principled adversarial evaluation pipelines grounded in human ability.

In Chapter 4, we extended adversarial benchmarking into real-world contexts by examining misleading video headlines. This task surfaces a distinct kind of adversarialness, stemming

from subjective and interpretive disagreement. We proposed a human-in-the-loop annotation pipeline to model these disagreements, treating them as informative rather than noisy. This chapter demonstrated that in high-stakes tasks like misinformation detection, it is critical to model human perspectives and ambiguity to surface realistic model limitations.

Chapter 5 turned to the issue of model trustworthiness and user overreliance. We introduced GRACE, a benchmark and metric designed to evaluate how well LLM confidence aligns with human judgments, by comparing models and human experts in a competitive QA setting with abstention, latency, and correctness signals. We showed that models are often overconfident when wrong, in ways humans typically are not. This overconfidence poses real risks in user-facing scenarios. We proposed new metrics, such as CALSCORE and CALSCORE^2, that penalize model overconfidence relative to human abstention behavior, and highlighted opportunities for improving trust through confidence modulation and better verbalized uncertainty.

Finally, while GRACE addresses trust issues in single-LLM settings by aligning confidence with human expectations, multi-agent systems introduce a new layer of complexity: reasoning is distributed, failure is harder to localize, and outputs are often opaque to users. Chapter 6 addressed a growing concern: the complexity and opacity of multi-agent AI systems. While such systems can decompose and solve complex tasks, they often fail in ways that are difficult for users to diagnose. We introduced VeriLA, a verification framework that uses human-labeled agent-level correctness, planner structure, and confidence cues to identify and predict execution failures. This approach supports fine-grained, interpretable evaluation of multi-agent chains, enabling users to debug failures and intervene, thus improving overall system accountability.

7.2 Future Directions: Toward Human-Grounded Real-world AI

We hope that this dissertation lays the groundwork for future research in developing evaluation frameworks and AI tools that are deeply aligned with user expectations; not only in what they achieve, but in how they behave, explain themselves, and fail. The following research directions reflect a multi-year agenda for building robust, trustworthy, and user-aligned language technologies.

While GRACE and ADVSCORE focused on developing evaluation metrics and benchmarks to assess models, it primarily emphasized quantitative skill levels without fully accounting for the perspectives of diverse stakeholders, particularly experts from interdisciplinary domains. Moving forward, I aim to develop richer and more nuanced representations of model skill by comparing domain experts' mental models and expectations to the actual behavior of language models. This comparison will enable more robust assessments of model reliability, especially in high-stakes contexts where expert reasoning is the benchmark for trustworthiness.

To support this vision, I will extend human-grounded benchmarking approaches to domain-specific settings such as healthcare, education, and law, mainly focusing on questions: **does the model reason in ways that experts find intuitive, justifiable, and reliable?** I plan to design evaluation pipelines that (1) recruit annotators with matching expertise, (2) calibrate tasks to real-world difficulty levels, and (3) introduce new metrics that reveal when models deviate from expert reasoning, even when surface-level accuracy appears high. This approach will surface hidden risks, capture deployment-relevant failure modes, and help develop language models that better support professionals in specialized domains.

VeriLA and GRACE revealed that LLMs often exhibit overconfident, brittle, or opaque

reasoning patterns that diverge from human logic, making them difficult to trust or supervise, especially in high-stakes settings. Thus, I plan to develop systems that actively support interpretability and accountability in model behavior. Specifically, I will design modular and transparent agentic systems that trace their reasoning steps, validate outputs against user-defined criteria, and expose failure points in ways users can inspect and debug. Building on VMH, which emphasized the importance of modeling human expectations and adversarial examples grounded in real-world ambiguity, partially from multimodal content, I also plan to extend this vision to the design of multimodal agentic interfaces. Going beyond text-only outputs, I will explore how visuals, structured reasoning diagrams, and interactive step-by-step justifications can help users better interpret and guide model behavior to enhance human-AI collaboration. These agentic tools (e.g., decision-supporting, domain-specific assistants) will dynamically adapt and calibrate their output style based on user goals, preferences, and cognitive needs.

Together, these directions aim to enable more safe, responsible, and trustworthy use of LLMs in practical settings, where the ability to explain, verify, and interact meaningfully with models is as important as raw performance. I hope to design systems that are responsive to human reasoning, ultimately supporting better decision-making and more equitable deployment across domains.

Chapter 8: Appendix

8.1 An Example Plan from Planning Agent

Example Plan from Planning Agent


```
{
  "id_": 0,
  "question": "Janet's ducks lay 16 eggs per day. She eats 3 and uses 4 for baking. She sells
       the rest at $2 each. How much does she earn daily?",
  "answer": "Janet sells 16 - 3 - 4 = <<16-3-4=9>>9 eggs a day.\nShe makes 9 * 2 = $
      <<9*2=18>>18.\n#### 18",
  "system_prompt": "You are a helpful assistant in solving math questions.",
  "user_prompt": {
    "1": "Identify all quantities and the price per egg.",
    "2": "Subtract eaten eggs from total.",
    "3": "Subtract baking eggs from remaining.",
    "4": "Multiply eggs left by price."
  },
  "plan": [
    { "id": 1, "name": "identify", "input": "question", "output": "quantities, price" },
    { "id": 2, "name": "subtract", "input": "total, eaten", "output": "post-breakfast" },
    { "id": 3, "name": "subtract", "input": "post-breakfast, baking", "output": "for sale" },
    { "id": 4, "name": "multiply", "input": "for sale, price", "output": "earnings" }
  ],
  "edges": [
    [1, 2], [2, 3], [1, 3], [3, 4], [1, 4]
  ]
}
```

8.2 List of Prompts

We provide the prompts used for planning, agent execution, and criteria evaluation by external evaluators.

Prompt used for planning

```
You are a planner responsible for creating high-level plans to solve any tasks using a set of agents.
Your goal is to break down a given task into a sequence of subtasks that, when executed correctly by the appropriate agents, will lead to the correct solution.

For each step in the plan:
1. Describe the subtask the agent must perform.
2. Provide a brief, self-contained description of the expected inputs and outputs. Do not include any specific values or examples.
3. Provide a user prompt for each task that includes the expected input and output information.

Represent your plan as a graph where each node corresponds to a step, and each edge represents a dependency between two steps.
If a node requires the output from a previous node as an input, ensure it is included in the edge list.
The output should be structured in the following JSON format:
{
"nodes": <list of JSON nodes { "id": <node id as integer>, "name": <assigned agent name>, "task": <task instruction>,
"input": <list of inputs>, "output": <list of outputs>}>,
"edges": <list of tuples [node_id, node_id]>
"user_prompts": <list of strings per node>
}
Available agents: {agent taxonomy}

Examples
{plan demonstration examples}
--------------------------------------------------------------------------------
{task query}
```

Prompt used for agent execution and verbalized confidence

```
Use the following contextual information to answer: {context info}.

If contextual information is “None”, answer it without external information.

JUST PERFORM WHAT YOU ARE ASKED TO DO, DO NOT ANSWER THE QUESTION, JUST BECAUSE THE QUESTION EXISTS IN THE

PROMPT.

Your answer should always be in JSON object format. {answer: <answer>, confidence: <confidence>}.
```

```
{subtask instruction prompt from the plan} + Also, provide how confident you are in your answer.

If not, use your own memory to execute the prompt as best as you can.

If you do not know the answer, your confidence should be 0.0.

The answer format should be like {answer: <text>, confidence: <float value between [0-1]>}.
```

Prompt used for using LLM evaluator with human-defined criteria

```
You’re a helpful assistant that evaluates an agent {agent}’s answer in different criteria.

Your answer should always be in JSON format.

{‘criteria’:‘criteria score’}.”
```

```
Please evaluate the following agent’s answer to a user prompt with the following context information.

If the context information is ‘None’, ignore and use your own knowledge to answer.

Here are some examples to help you score the agent’s answer: {agent examples}

The user prompt: {user}.

The context information: {context info}.

The agent: {agent}.

The agent’s input format: {agent input}.

The agent’s output format: {agent output}.

The agent’s answer: {answer}.

You should find the agent’s essential criteria to evaluate the answer from {agent criteria list}

Then, score each criterion in a float value between [0-1] in the ’criteria score’ placeholder.

Your answer should look like:

{’criteria score’: {’<criteria 1>’: <float value between [0-1]>, ’<criteria 2>’: <float value between [0-1]>,

’<criteria 3>’: <float value between [0-1]>, ’<criteria 4>’: <float value between [0-1]>, ’<criteria 5>’: <float

value between [0-1]>} }

Your answer should always be in JSON object format: {’criteria’:’criteria score’}.
```

8.3 Agent Registry Curation using Chain-of-Thought Prompting

We provide the AI practioners to examine 50 samples resulting from one-shot promoting, and determine which agents are adequate for math reasoning application. As an example, we provide result for the task in Figure 6.2:

CoT planning result example

```
{
  "question": "Janet's ducks lay 16 eggs per day. She eats 3, bakes with 4, and sells the
      rest at $2 each. How much does she make daily?",
  "answer": "She sells 16 - 3 - 4 = <<16-3-4=9>>9 eggs.\n9 * 2 = $<<9*2=18>>18\n#### 18",
  "result": {
    "role": "assistant",
    "plans": {
      "step1": "Add eggs used for breakfast and baking.",
      "step2": "Subtract from total eggs to get eggs sold.",
      "step3": "Multiply eggs sold by price."
    },
    "answers": {
      "step1": "3 + 4 = 7 eggs used.",
      "step2": "16 - 7 = 9 eggs to sell.",
      "step3": "9 * 2 = $18 per day."
    }
  }
}
```

8.3.1 Candidate Agents for Other Tasks

We additionally present potential agent registries for the open-domain question answering task and fact-checking task (Table 8.1). We plan to expand VeriLA in these domains.

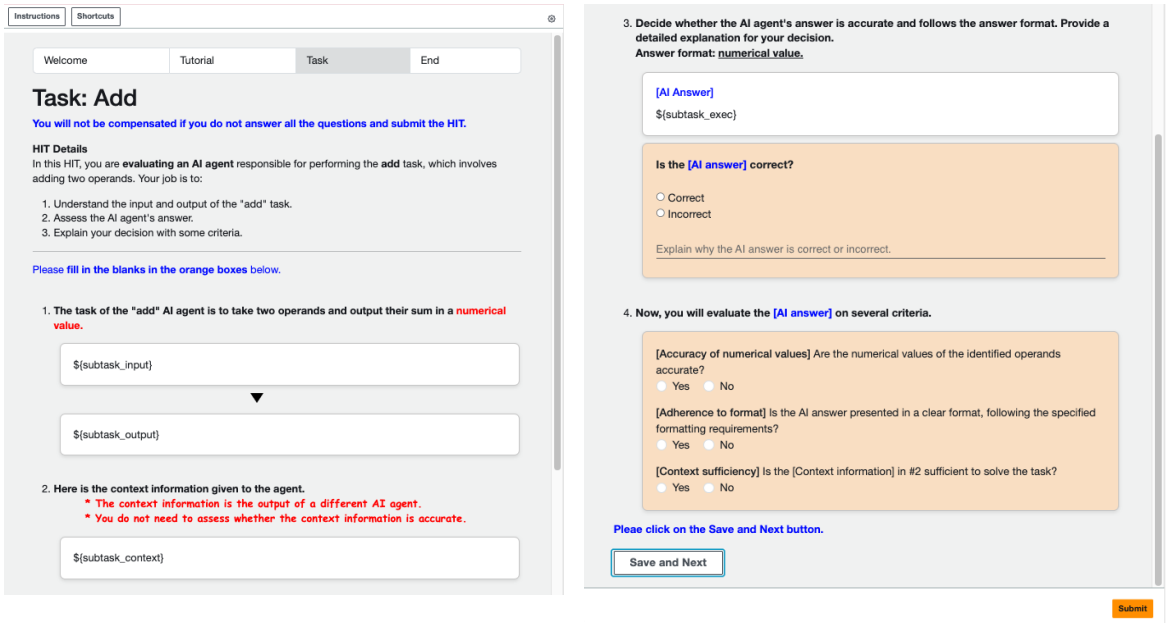

Figure 8.1: Example annotation for evaluation of LLM execution result of "add" subtask. We prohibit the users from moving on to the next page if they did not get the answer correct for questions in the tutorial.

8.4 Human-Defined Criteria for Other Agents

Table 8.2 lists evaluation criteria for each agent in our agent registry.

8.5 Crowdsourced Annotation Procedure

We used MTurk platform to recruit crowdworkers for labeling the GSM8K dataset. For each instance, three labels were collected from workers who passed the qualification test, with an average pay rate of $14 per hour. For a subtask assigned to an agent, workers were asked to annotate whether the agent's answer was successful or not, given its role, input, and evaluation criteria. A sample interface illustrating the annotation process is shown in Figure 8.1.

Table 8.1: Candidate agents for open-domain question answering or fact-checking tasks.

Agent	Role	Input	Output	Output Format
Identify Query	Identify query for Wikipedia retrieval	Text	Text query	Text
Retrieve From Wikipedia	Retrieve from Wikipedia database	Text query	Most relevant paragraph	Text
Retrieve From DB	Retrieve paragraph from selected DB	Text query	Most relevant paragraph	Text
Brainstorm	Brainstorm ideas given input format	Text query	Brainstormed ideas	List
Rationalize	Generate explanation for given input	Text query	Explanation	Text
Classify	Assign category label to the input	Text query	Category-labeled JSON	JSON
Partition	Partition the problem into sub-problems	Text query	Partitioned sub-tasks	List
Merge	Combine multiple inputs into one	List of text	Combined text	Text
Compare	Compare two or more items	List of text queries	Comparison result	Text
Suggest	Generate improved ideas from input and instruction	Text query	Improved ideas	List
Transform	Transform text using suggested idea	Text query	Transformed text	Text
Extract	Extract smaller unit of text from input	Text	Extracted content	Text
Rate	Give rating to input	List of text	Sorted list by rating	List
Rank	Rank text based on criteria	List, criteria	Ranked list	List
Computation	Solve numerical computation task	Text query	Computed result	Text
Format Text	Generate formatted text from input and instruction	Text, instruction	Formatted text	Text
Prune	Remove redundant information	Text query	Pruned text	Text
Add	Add missing details to input	Text query	Completed text	Text
Correct	Correct errors in input text	Text query	Corrected text	Text

Table 8.2: Human-designed agent criteria. Each agent's criteria are assigned by users based on their own experience performing the task using the agent registry. Thus, these criteria are grounded in human needs and are integrated into LLM evaluators, with their outputs used as part of our agent verifier features.

Subtask and Description	**Essential Criteria**
Identify Operands — Identify operands with text description of each operand	**Accuracy**: Are numerical values accurate? **Relevance**: Are all operands relevant? **Coverage**: Are all necessary operands identified? **Clarity**: Are operand descriptions clear? **Format Adherence**: Is the output correctly formatted?
Add — Add numbers or dates	**Accuracy**: Is the sum correct? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to solve the task?
Subtract — Subtract numbers or dates	**Accuracy**: Is the result correct? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to solve the task?
Multiply — Multiply numbers	**Accuracy**: Is the result correct? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to solve the task?
Divide — Divide numbers	**Accuracy**: Is the result correct? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to solve the task?
Filter — Filter a list based on a condition	**Relevance**: Are irrelevant items excluded? **Completeness**: Are all valid items included? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to solve the task?
Sort — Sort a list by an attribute	**Correctness**: Is the order accurate? **Completeness**: Are all items included? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to solve the task?
Convert Format — Convert input from one format to another	**Accuracy**: Was the conversion correct? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to perform the conversion?
Date Lookup — Identify year, month, and day from a natural language description	**Accuracy**: Is the date correctly identified? **Format Adherence**: Is the output correctly formatted? **Context Sufficiency**: Is the context enough to extract the date?

Bibliography

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Esma Aïmeur, Sabrine Amri, and Gilles Brassard. 2023. Fake news, disinformation and misinformation in social media: a review. *Social Network Analysis and Mining*, 13(1):30.

Oluwafemi Akanfe, Paras Bhatt, and Diane A Lawong. 2025. Technology advancements shaping the financial inclusion landscape: Present interventions, emergence of artificial intelligence and future directions. *Information Systems Frontiers*, pages 1–24.

Nader Akoury, Shufan Wang, Josh Whiting, Stephen Hood, Nanyun Peng, and Mohit Iyyer. 2020. Storium: A dataset and evaluation platform for machine-in-the-loop story generation. *arXiv preprint arXiv:2010.01717*.

Negar Arabzadeh, Siqing Huo, Nikhil Mehta, Qingyun Wu, Chi Wang, Ahmed Hassan Awadallah, Charles L. A. Clarke, and Julia Kiseleva. 2024. Assessing and verifying task utility in LLM-powered applications. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21868–21888, Miami, Florida, USA. Association for Computational Linguistics.

Ian Arawjo, Chelse Swoopes, Priyan Vaithilingam, Martin Wattenberg, and Elena L. Glassman. 2024. Chainforge: A visual toolkit for prompt engineering and llm hypothesis testing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '24, New York, NY, USA. Association for Computing Machinery.

Neil Band, Xuechen Li, Tengyu Ma, and Tatsunori Hashimoto. 2024. Linguistic calibration of long-form generations. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021a. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021b. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yonatan Belinkov and Yonatan Bisk. 2017. Synthetic and natural noise both break neural machine translation. *arXiv preprint arXiv:1711.02173*.

Emily M Bender and Batya Friedman. 2018. Data statements for natural language processing:

Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. 2013. Evasion attacks against machine learning at test time. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13*, pages 387–402. Springer.

Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1467–1474.

Battista Biggio and Fabio Roli. 2018. Wild patterns: Ten years after the rise of adversarial machine learning. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 2154–2156.

Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of the 19th International Conference on Computational Statistics (COMPSTAT 2010)*, pages 177–186. Physica-Verlag HD.

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. 2017. From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP workshop: natural language processing meets journalism*, pages 84–89.

Samuel R Bowman. 2023. Eight things to know about large language models. *arXiv e-prints*, pages arXiv–2304.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Samuel R. Bowman and George Dahl. 2021a. What will it take to fix benchmarking in natural language understanding? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4843–4855, Online. Association for Computational Linguistics.

Samuel R Bowman and George E Dahl. 2021b. What will it take to fix benchmarking in natural language understanding? In *2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 4843–4855. Association for Computational Linguistics (ACL).

Samuel R Bowman, Jeeyoon Hyun, Ethan Perez, Edwin Chen, Craig Pettit, Scott Heiner, Kamilė Lukošiūtė, Amanda Askell, Andy Jones, Anna Chen, et al. 2022. Measuring progress on scalable oversight for large language models. *arXiv preprint arXiv:2211.03540*.

Jordan Boyd-Graber, Shi Feng, and Pedro Rodriguez. 2018. *Human-Computer Question Answering: The Case for Quizbowl*. Springer Verlag.

Jordan Boyd-Graber, Brianna Satinoff, He He, and Hal Daumé III. 2012. Besting the quiz master: Crowdsourcing incremental classification games. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1290–1301, Jeju Island, Korea. Association for Computational Linguistics.

Petter Bae Brandtzaeg, Asbjørn Følstad, and Marita Skjuve. 2025. Emerging ai individualism: how young people integrate social ai into everyday life. *Communication and Change*, 1(1):11.

Leo Breiman, Jerome Friedman, Richard A Olshen, and Charles J Stone. 1986. *Classification and Regression Trees*. Wadsworth.

Eleftheria Briakou and Marine Carpuat. 2020. Detecting fine-grained cross-lingual semantic divergences without supervision by learning to rank. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1563–1580.

Glenn W Brier. 1950. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yuyan Bu, Qiang Sheng, Juan Cao, Peng Qi, Danding Wang, and Jintao Li. 2023. Online misinformation video detection: A survey. *arXiv e-prints*, pages arXiv–2302.

Shuyang Cao and Lu Wang. 2021. Cliff: Contrastive learning for improving faithfulness and factuality in abstractive summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6633–6649.

Richard Caruana. 2019. Friends don't let friends deploy black-box models: The importance of intelligibility in machine learning. In *Proceedings of the 25th ACM SIGKDD International*

Conference on Knowledge Discovery & Data Mining, KDD '19, page 3174, New York, NY, USA. Association for Computing Machinery.

Jesse Chandler, Pam Mueller, and Gabriele Paolacci. 2014. Nonnaïveté among amazon mechanical turk workers: Consequences and solutions for behavioral researchers. *Behavior research methods*, 46:112–130.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45.

Yimin Chen, Niall J Conroy, and Victoria L Rubin. 2015. Misleading online content: recognizing clickbait as" false news". In *Proceedings of the 2015 ACM on workshop on multimodal deception detection*, pages 15–19.

Yuheng Cheng, Ceyao Zhang, Zhengwen Zhang, Xiangrui Meng, Sirui Hong, Wenhao Li, Zihao Wang, Zekai Wang, Feng Yin, Junhua Zhao, et al. 2024. Exploring large language model based intelligent agents: Definitions, methods, and prospects. *https://arxiv.org/abs/2401.03428*.

Sophie Chesney, Maria Liakata, Massimo Poesio, and Matthew Purver. 2017. Incongruent headlines: Yet another way to mislead your readers. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 56–61.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *https://arxiv.org/pdf/2110.14168*.

David R Cox. 1958. The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, 20(2):215–242.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*.

Hal Daume III and Daniel Marcu. 2005. Bayesian summarization at duc and a suggestion for extrinsic evaluation. In *Proceedings of the Document Understanding Conference, DUC-2005, Vancouver, USA*.

Aida Mostafazadeh Davani, Mark Díaz, and Vinodkumar Prabhakaran. 2022. Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *Transactions of the Association for Computational Linguistics*, 10:92–110.

Jiawen Deng, Kiyan Heybati, and Hemang Yadav. 2025. Development and validation of machine-learning models for predicting the risk of hypertriglyceridemia in critically ill patients receiving propofol sedation using retrospective data: a protocol. *BMJ open*, 15(1):e092594.

Bharat Dhiman. 2023. Does artificial intelligence help journalists: A boon or bane?

Sameer Dhoju, Md Main Uddin Rony, Muhammad Ashad Kabir, and Naeemul Hassan. 2019. Differences in health news from reliable and unreliable media. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 981–987.

Laura Edelson, Minh-Kha Nguyen, Ian Goldstein, Oana Goga, Damon McCoy, and Tobias Lauinger. 2021. Understanding engagement with us (mis) information news sources on facebook. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 444–463.

Julian Eisenschlos, Bhuwan Dhingra, Jannis Bulian, Benjamin Börschinger, and Jordan Boyd-Graber. 2021a. Fool me twice: Entailment from Wikipedia gamification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 352–365, Online. Association for Computational Linguistics.

Julian Eisenschlos, Bhuwan Dhingra, Jannis Bulian, Benjamin Börschinger, and Jordan Boyd-Graber. 2021b. Fool me twice: Entailment from wikipedia gamification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 352–365.

Susan E Embretson and Steven P Reise. 2013. *Item response theory for psychologists*. Psychology Press.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. 2020. Adversarial robustness as a prior for learned representations. *arXiv preprint arXiv:1906.00945*.

Shi Feng, Eric Wallace, and Jordan Boyd-Graber. 2019. Misleading failures of partial-input baselines. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5533–5538.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber.

2018. Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics.

William Ferreira and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*. ACL.

David A Ferrucci. 2012. Introduction to "this is watson". *IBM Journal of Research and Development*, 56(3.4):1–1.

Yoav Freund and Robert E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

Josh Gardner. 2024. *Toward Robust, Reliable, and Generalizable Models for Tabular Data*. Ph.D. thesis, University of Washington.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna Clinciu, Dipanjan Das, Kaustubh Dhole, et al. 2021. The gem benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 96–120.

Banda Gerald. 2018. A brief review of independent, dependent and one sample t-test. *International journal of applied mathematics and theoretical physics*, 4(2):50–54.

Malik Ghallab, Dana Nau, and Paolo Traverso. 2004. *Automated Planning: theory and practice*. Elsevier-Morgan Kauffman.

Salvatore Giorgi, Tingting Liu, Ankit Aich, Kelsey Jane Isman, Garrick Sherman, Zachary Fried, João Sedoc, Lyle Ungar, and Brenda Curtis. 2024. Modeling human subjectivity in LLMs using explicit and implicit human factors in personas. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7174–7188, Miami, Florida, USA. Association for Computational Linguistics.

Josh A. Goldstein, Girish Sastry, Micah Musser, Renee DiResta, Matthew Gentzel, and Katerina Sedova. 2023. Generative language models and automated influence operations: Emerging threats and potential mitigations.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. *stat*, 1050:20.

Maharshi Gor, Hal Daumé III, Tianyi Zhou, and Jordan Boyd-Graber. 2024. Do great minds think alike? investigating human-ai complementarity in question answering with caimira. *arXiv preprint arXiv:2410.06524*.

Nitesh Goyal, Minsuk Chang, and Michael Terry. 2024. Designing for human-agent alignment: Understanding what humans want from their agents. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–6.

Gary Graham, Tahir M Nisar, Guru Prabhakar, Royston Meriton, and Sadia Malik. 2025. Chatbots in customer service within banking and finance: Do chatbots herald the start of an ai revolution in the corporate world? *Computers in Human Behavior*, 165:108570.

Madeleine Grunde-McLaughlin, Michelle S. Lam, Ranjay Krishna, Daniel Weld, and Jeffrey Heer. 2025a. Designing llm chains by adapting techniques from crowdsourcing workflows.

Madeleine Grunde-McLaughlin, Michelle S Lam, Ranjay Krishna, Daniel S Weld, and Jeffrey Heer. 2025b. Designing llm chains by adapting techniques from crowdsourcing workflows. *ACM Transactions on Computer-Human Interaction*, 32(3):1–57.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Yui Ha, Jeongmin Kim, Donghyeon Won, Meeyoung Cha, and Jungseock Joo. 2018. Characterizing clickbaits on instagram. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12.

Andreas Hanselowski, Christian Stab, Claudia Schulz, Zile Li, and Iryna Gurevych. 2019. A richly annotated corpus for different tasks in automated fact-checking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 493–503.

Moshe Hazoom, Vibhor Malik, and Ben Bogin. 2021. Text-to-sql in the wild: a naturally-occurring dataset based on stack exchange data. *arXiv preprint arXiv:2106.05006*.

He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé, III. 2016. Opponent modeling in deep reinforcement learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1804–1813, New York, New York, USA. PMLR.

Wanrong He, Andrew Mao, and Jordan Boyd-Graber. 2022. Cheater's bowl: Human vs. computer

search strategies for open-domain QA. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3627–3639, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sebastian Hofstätter, Jiecao Chen, Karthik Raman, and Hamed Zamani. 2023. Fid-light: Efficient and effective retrieval-augmented text generation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1437–1447.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with mace. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130.

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2023a. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*.

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2023b. Look before you leap: An exploratory study of uncertainty measurement for large language models.

Lujain Ibrahim, Saffron Huang, Lama Ahmad, and Markus Anderljung. 2024. Beyond static ai evaluations: advancing human interaction evaluations for llm harms and risks. *arXiv preprint arXiv:2405.10632*.

Evgenia Ilia and Wilker Aziz. 2024. Predict the next word:< humans exhibit uncertainty in this task and language models _>. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2, pages 234–255.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32.

Laure Jaeger, Tom Jorquera, Sylvain Lemouzy, Christian Gogu, Stéphane Segonds, and Christian Bes. 2013. Uncertainty propagation in multi-agent systems for multidisciplinary optimization problems. In *10th World Congress on Structural and Multidisciplinary Optimization (WCSMO 10)*, pages pp–1.

Ken Jennings. 2007. *Brainiac: adventures in the curious, competitive, compulsive world of trivia buffs*. Villard.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023a. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023b. Towards mitigating llm hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Geoffrey H John and Pat Langley. 1995. Estimating continuous distributions in bayesian classifiers. In *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI 1995)*, pages 338–345. Morgan Kaufmann.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. 2020. Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–14.

Divyansh Kaushik, Douwe Kiela, Zachary C. Lipton, and Wen-tau Yih. 2021. On the efficacy of adversarial data collection for question answering: Results from a large-scale randomized study. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6618–6633, Online. Association for Computational Linguistics.

Kian Kenyon-Dean, Eisha Ahmed, Scott Fujimoto, Jeremy Georges-Filteau, Christopher Glasz, Barleen Kaur, Auguste Lalande, Shruti Bhanderi, Robert Belfer, Nirmal Kanagasabai, et al. 2018. Sentiment analysis: It's complicated! In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1886–1895.

Urja Khurana, Eric Nalisnick, Antske Fokkens, and Swabha Swayamdipta. 2024. Crowd-calibrator: Can annotator disagreement inform calibration in subjective tasks?

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.

Tae Soo Kim, Yoonjoo Lee, Jamin Shin, Young-Ho Kim, and Juho Kim. 2024. Evallm: Interactive evaluation of large language model prompts on user-defined criteria. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '24, New York, NY, USA. Association for Computing Machinery.

Lea Krause, Wondimagegnhue Tufa, Selene Báez Santamaría, Angel Daza, Urja Khurana, and Piek Vossen. 2023. Confidently wrong: exploring the calibration and expression of (un) certainty of large language models in a multilingual setting. In *Proceedings of the workshop on multimodal, multilingual natural language generation and multilingual WebNLG Challenge (MM-NLG 2023)*, pages 1–9.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav

Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

John P. Lalor, Hao Wu, and Hong Yu. 2016. Building an evaluation scale using item response theory. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 648–657, Austin, Texas. Association for Computational Linguistics.

John P Lalor, Hao Wu, and Hong Yu. 2019. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4249–4259.

LangChain. 2013. Langchain. `https://github.com/langchain-ai/langchain`.

Mina Lee, Percy Liang, and Qian Yang. 2022a. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA. Association for Computing Machinery.

Mina Lee, Megha Srivastava, Amelia Hardy, John Thickstun, Esin Durmus, Ashwin Paranjape, Ines Gerard-Ursin, Xiang Lisa Li, Faisal Ladhak, Frieda Rong, et al. 2022b. Evaluating human-language model interaction. *arXiv preprint arXiv:2212.09746*.

Jingshu Li, Yitian Yang, Renwen Zhang, and Yi-chieh Lee. 2024a. Overconfident and unconfident ai hinder human-ai collaboration. *arXiv preprint arXiv:2402.07632*.

Zongxia Li, Ishani Mondal, Huy Nghiem, Yijun Liang, and Jordan Lee Boyd-Graber. 2024b. PEDANTS: Cheap but effective and interpretable answer equivalence. In *Findings of the*

Association for Computational Linguistics: EMNLP 2024, pages 9373–9398, Miami, Florida, USA. Association for Computational Linguistics.

Zongxia Li, Ishani Mondal, Huy Nghiem, Yijun Liang, and Jordan Lee Boyd-Graber. 2024c. PEDANTS: Cheap but effective and interpretable answer equivalence. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9373–9398, Miami, Florida, USA. Association for Computational Linguistics.

Q Vera Liao and Ziang Xiao. 2023. Rethinking model evaluation as narrowing the socio-technical gap. *arXiv preprint arXiv:2306.03100*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Chinnadhurai Sankar Liu, Shuyang Chen, Daniel Khashabi, Esin Durmus, Mohit Bansal, and Hannaneh Hajishirzi. 2023a. G-eval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2305.13287*.

Fang Liu, Yang Liu, Lin Shi, Houkun Huang, Ruifeng Wang, Zhen Yang, Li Zhang, Zhongqi Li, and Yuchi Ma. 2024a. Exploring and evaluating hallucinations in llm-powered code generation. *arXiv preprint arXiv:2404.00971*.

Ollie Liu, Deqing Fu, Dani Yogatama, and Willie Neiswanger. 2024b. Dellma: A framework for decision making under uncertainty with large language models. *arXiv preprint arXiv:2402.02392*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023b. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

FM Lord, MR Novick, and Allan Birnbaum. 1968. Statistical theories of mental test scores.

Renze Lou, Kai Zhang, and Wenpeng Yin. 2024. Large language model instruction following: A survey of progresses and challenges. volume 50, pages 1053–1095.

Zhiyi Ma, Kawin Ethayarajh, Tristan Thrush, Somya Jain, Ledell Wu, Robin Jia, Christopher Potts, Adina Williams, and Douwe Kiela. 2021a. Dynaboard: An evaluation-as-a-service platform for holistic next-generation benchmarking. *Advances in Neural Information Processing Systems*, 34:10351–10367.

Zhiyi Ma, Kawin Ethayarajh, Tristan Thrush, Somya Jain, Ledell Yu Wu, Robin Jia, Christopher Potts, Adina Williams, and Douwe Kiela. 2021b. Dynaboard: An evaluation-as-a-service platform for holistic next-generation benchmarking. In *Neural Information Processing Systems*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.

Irene Martín-Morató, Manu Harju, and Annamaria Mesaros. 2021. Crowdsourcing strong labels for sound event detection. In *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 246–250. IEEE.

Fernando Martínez-Plumed, Ricardo BC Prudêncio, Adolfo Martínez-Usó, and José Hernández-Orallo. 2019. Item response theory in ai: Analysing machine learning classifiers at the instance level. *Artificial intelligence*, 271:18–42.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2020. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *57th Annual Meeting of the Association*

for Computational Linguistics, ACL 2019, pages 3428–3448. Association for Computational Linguistics (ACL).

Timothy R McIntosh, Teo Susnjak, Nalin Arachchilage, Tong Liu, Dan Xu, Paul Watters, and Malka N Halgamuge. 2025. Inadequacies of large language model benchmarks in the era of generative artificial intelligence. *IEEE Transactions on Artificial Intelligence*.

Marco Melis, Ambra Demontis, Battista Biggio, Gavin Brown, Giorgio Fumera, and Fabio Roli. 2017. Is deep learning safe for robot vision? adversarial examples against the icub humanoid. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 751–759.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Sewon Min, Luke Zettlemoyer, Hannaneh Hajishirzi, et al. 2022. Crepe: Open-domain question answering with false presuppositions. *arXiv e-prints*, pages arXiv–2211.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th Inter-*

national Conference on Computational Linguistics, pages 2340–2353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Khanh Nguyen and Brendan O'Connor. 2015. Posterior calibration and exploratory analysis for natural language processing models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1587–1598, Lisbon, Portugal. Association for Computational Linguistics.

Yixin Nie, Yicheng Wang, and Mohit Bansal. 2018. Analyzing compositionality-sensitivity of nli models. *ArXiv*, abs/1811.07033.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

OpenAI. 2023. Chatgpt (mar 14 version). Large language model.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Vishakh Padmakumar and He He. 2022. Machine-in-the-loop rewriting for creative image captioning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 573–586, Seattle, United States. Association for Computational Linguistics.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Aakash Parmar, Rakesh Katariya, and Vatsal Patel. 2019. A review on random forest: An ensemble classifier. In *International conference on intelligent data communication technologies and internet of things (ICICI) 2018*, pages 758–763. Springer.

Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel R Bowman. 2021. Bbq: A hand-built bias benchmark for question answering. *arXiv preprint arXiv:2110.08193*.

Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. 2018. Comparing bayesian models of annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia

Glaese, Nat McAleese, and Geoffrey Irving. 2022. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3419–3448.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2016. Credibility assessment of textual claims on the web. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 2173–2178, New York, NY, USA. Association for Computing Machinery.

Martin Potthast, Tim Gollub, Matthias Hagen, and Benno Stein. 2018. The clickbait challenge 2017: Towards a regression model for clickbait strength. *arXiv preprint arXiv:1812.10847*.

Archiki Prasad, Alexander Koller, Mareike Hartmann, Peter Clark, Ashish Sabharwal, Mohit Bansal, and Tushar Khot. 2024. ADaPT: As-needed decomposition and planning with language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4226–4252, Mexico City, Mexico. Association for Computational Linguistics.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Jessica Quaye, Alicia Parrish, Oana Inel, Charvi Rastogi, Hannah Rose Kirk, Minsuk Kahng, Erin Van Liemt, Max Bartolo, Jess Tsang, Justin White, et al. 2024. Adversarial nibbler: An open red-teaming method for identifying diverse harms in text-to-image generation. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, pages 388–406.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena,

Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Inioluwa Deborah Raji and Joy Buolamwini. 2019. Actionable auditing: Investigating the impact of publicly naming biased performance results of commercial ai products. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 429–435.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Arijit Ray, Giedrius Burachas, Yi Yao, and Ajay Divakaran. 2019. Lucid explanations help: Using a human-ai image-guessing game to evaluate machine explanation helpfulness. *CoRR*, abs/1904.03285.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2019. Do ImageNet classifiers generalize to ImageNet? In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5389–5400. PMLR.

Mark Reckase. 1998. Item response theory: Parameter estimation techniques. *Applied Psychological Measurement*, 22:89–91.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Philip Resnik and Jimmy Lin. 2010. Evaluation of nlp systems. *The handbook of computational linguistics and natural language processing*, pages 271–295.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Julio Cesar Soares dos Rieis, Fabrício Benevenuto de Souza, Pedro Olmo S Vaz de Melo, Raquel Oliveira Prates, Haewoon Kwak, and Jisun An. 2015. Breaking the news: First impressions matter on online news. In *Ninth International AAAI Conference on Web and Social Media*.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P. Lalor, Robin Jia, and Jordan Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change NLP leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4486–4503, Online. Association for Computational Linguistics.

Pedro Rodriguez, Shi Feng, Mohit Iyyer, He He, and Jordan Boyd-Graber. 2019a. Quizbowl: The case for incremental question answering. *arXiv preprint arXiv:1904.04792*.

Pedro Rodriguez, Shi Feng, Mohit Iyyer, He He, and Jordan L. Boyd-Graber. 2019b. Quizbowl: The case for incremental question answering. *CoRR*, abs/1904.04792.

Md Main Uddin Rony, Naeemul Hassan, and Mohammad Yousuf. 2017. Diving deep into clickbaits: Who use them to what extents in which topics with what effects? In *Proceedings of the 2017 IEEE/ACM international conference on advances in social networks analysis and mining 2017*, pages 232–239.

Alexis Ross, Ana Marasović, and Matthew E Peters. 2021. Explaining nlp models via minimal contrastive editing (mice). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3840–3852.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Barbara Rychalska, Dominika Basaj, Alicja Gosiewska, and Przemysław Biecek. 2019. Models in the wild: On corruption robustness of neural nlp systems. In *Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part III*, page 235–247, Berlin, Heidelberg. Springer-Verlag.

Paul Röttger, Bertie Vidgen, Dirk Hovy, and Janet B. Pierrehumbert. 2021. Two contrasting data annotation paradigms for subjective nlp tasks. *ArXiv*. ArXiv:2112.07475.

Mattia Samory, Vartan Kesiz Abnousi, and Tanushree Mitra. 2020. Characterizing the social media news sphere through user co-sharing practices. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pages 602–613.

Marta Sandri, Elisa Leonardelli, Sara Tonelli, and Elisabetta Ježek. 2023. Why don't you do it right? analysing annotators' disagreement in subjective tasks. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2420–2433.

Robert E. Schapire. 2013. *Explaining AdaBoost*. Springer Berlin Heidelberg, Berlin, Heidelberg.

Patrícia Schmidtová, Saad Mahamood, Simone Balloccu, Ondřej Dušek, Albert Gatt, Dimitra Gkatzia, David M Howcroft, Ondřej Plátek, and Adarsa Sivaprasad. 2024. Automatic metrics in natural language generation: A survey of current evaluation practices. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 557–583.

Lanyu Shang, Daniel Yue Zhang, Michael Wang, Shuyue Lai, and Dong Wang. 2019. Towards reliable online clickbait video detection: A content-agnostic approach. *Knowledge-Based Systems*, 182:104851.

Shreya Shankar, JD Zamfirescu-Pereira, Björn Hartmann, Aditya Parameswaran, and Ian Arawjo. 2024. Who validates the validators? aligning llm-assisted evaluation of llm outputs with human preferences. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pages 1–14.

Sasha Sheng, Amanpreet Singh, Vedanuj Goswami, Jose Alberto Lopez Magana, Wojciech Galuba, Devi Parikh, and Douwe Kiela. 2021. Human-adversarial visual question answering. *CoRR*, abs/2106.02280.

Chenglei Si, Navita Goyal, Sherry Tongshuang Wu, Chen Zhao, Shi Feng, Hal Daumé III, and

Jordan Boyd-Graber. 2023. Large language models help humans verify truthfulness–except when they are convincingly wrong. *arXiv preprint arXiv:2310.12558*.

Chenglei Si, Chen Zhao, Sewon Min, and Jordan Boyd-Graber. 2022. Re-examining calibration: The case of question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2814–2829.

Valeriya Slovikovskaya. 2019. Transfer learning from transformers to fake news challenge stance detection (fnc-1) task. *arXiv preprint arXiv:1910.14353*.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii. Association for Computational Linguistics.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Johan Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret

Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, Cesar Ferri, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Christopher Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, C. Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodolà, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Xinyue Wang, Gonzalo Jaimovitch-Lopez, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Francis Anthony Shevlin, Hinrich Schuetze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocon, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason

Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh Dhole, Kevin Gimpel, Kevin Omondi, Kory Wallace Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros-Colón, Luke Metz, Lütfi Kerem Senel, Maarten Bosma, Maarten Sap, Maartje Ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramirez-Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael Andrew Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Andrew Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter W

Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan Le Bras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Russ Salakhutdinov, Ryan Andrew Chi, Seungjae Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel Stern Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima Shammie Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven Piantadosi, Stuart Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsunori Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Venkatesh Ramasesh, vinay uday prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding

Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Kate Starbird, Ahmer Arif, and Tom Wilson. 2019. Disinformation as collaborative work: Surfacing the participatory nature of strategic information operations. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–26.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. 2024. On the self-verification limitations of large language models on reasoning and planning tasks. *arXiv e-prints*, pages arXiv–2402.

Elias Stengel-Eskin, Peter Hase, and Mohit Bansal. 2024. Lacie: Listener-aware finetuning for confidence calibration in large language models. *ArXiv*, abs/2405.21028.

Elias Stengel-Eskin and Benjamin Van Durme. 2023. Calibrated interpretation: Confidence estimation in semantic parsing. *Transactions of the Association for Computational Linguistics*, 11:1213–1231.

Theodore Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas Griffiths. 2023. Cognitive architectures for language agents. *Transactions on Machine Learning Research*.

Haotian Sun, Yuchen Zhuang, Lingkai Kong, Bo Dai, and Chao Zhang. 2023. Adaplanner: Adaptive planning from feedback with language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Simone Tedeschi, Johan Bos, Thierry Declerck, Jan Hajič, Daniel Hershcovich, Eduard Hovy, Alexander Koller, Simon Krek, Steven Schockaert, Rico Sennrich, Ekaterina Shutova, and Roberto Navigli. 2023. What's the meaning of superhuman performance in today's NLU? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12471–12491, Toronto, Canada. Association for Computational Linguistics.

Samuel Gbli Tetteh, Lois Azupwah, Atta Yaw Agyemana, Stephen Kwasi Adjei, Ankrah Prince Twumasi, and Sofo Mohammed-Nurudeen. 2025. Artificial intelligence in healthcare: A systematic review of virtual healthcare assistants. *Asian Journal of Probability and Statistics*, 27(7):43–62.

Jesse Thomason, Daniel Gordon, and Yonatan Bisk. 2019. Shifting the baseline: Single modality performance on visual navigation & QA. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1977–1983, Minneapolis, Minnesota. Association for Computational Linguistics.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea

Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Assaf Toledo, Shai Gretz, Edo Cohen-Karlik, Roni Friedman, Elad Venezian, Dan Lahav, Michal Jacovi, Ranit Aharonov, and Noam Slonim. 2019. Automatic argument quality assessment-new datasets and methods. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5625–5635.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothee Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2019. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*.

Jonathan Uesato, Brendan O'donoghue, Pushmeet Kohli, and Aaron Oord. 2018. Adversarial risk and the dangers of evaluating against weak attacks. In *International conference on machine learning*, pages 5025–5034. PMLR.

Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. 2024. Calibrating large language models using their generations only. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15440–15459, Bangkok, Thailand. Association for Computational Linguistics.

et al. Uma. 2024. Annotator-centric active learning for subjective nlp tasks. In *Proceedings of EMNLP 2024*, pages 1031–1042.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Towards debiasing nlu models from unknown biases. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7597–7610.

Clara Vania, Phu Mon Htut, William Huang, Dhara Mungra, Richard Yuanzhe Pang, Jason Phang, Haokun Liu, Kyunghyun Cho, and Samuel R. Bowman. 2021. Comparing test sets with item response theory. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1141–1158, Online. Association for Computational Linguistics.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *science*, 359(6380):1146–1151.

Jane Wakefield. 2016. Social media 'outstrips tv' as news source for young people. *BBC News*.

Mason Walker and Katerina Eva Matsa. 2021. News consumption across social media in 2021. *Pew Research Center*.

Eric Wallace, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. 2019. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*, 7:387–401.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. 2024a. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023a. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091*.

Rui Wang, Fei Mi, Yi Chen, Boyang Xue, Hongru Wang, Qi Zhu, Kam-Fai Wong, and Ruifeng Xu. 2024b. Role prompting guided domain adaptation with general capability preserve for large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2243–2255.

Shuting Ada Wang, Min-Seok Pang, and Paul A Pavlou. 2021a. Seeing is believing? how including a video in fake news influences users' reporting the fake news to social media platforms. *How*

Including a Video in Fake News Influences Users' Reporting the Fake News to Social Media Platforms (August 23, 2021).

Xingyao Wang, Zihan Wang, Jiateng Liu, Yangyi Chen, Lifan Yuan, Hao Peng, and Heng Ji. 2023b. Mint: Evaluating llms in multi-turn interaction with tools and language feedback. *arXiv preprint arXiv:2309.10691*.

Zijie J Wang, Dongjin Choi, Shenyu Xu, and Diyi Yang. 2021b. Putting humans in the natural language processing loop: A survey. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 47–52.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H Chi, Quoc V Le, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*.

Wei Wei and Xiaojun Wan. 2017. Learning to identify ambiguous and misleading news headlines. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4172–4178.

Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022a. Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA. Association for Computing Machinery.

Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022b. Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pages 1–22.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, et al. 2025. The rise and potential of large language model based agents: A survey. *Science China Information Sciences*, 68(2):121101.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *The Twelfth International Conference on Learning Representations*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024a. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. 2024b. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. 2021a. Vlm: Task-agnostic video-language model pre-training for video understanding. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4227–4239.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021b. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6787–6800.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain

question answering. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2013–2018.

Gal Yona, Roee Aharoni, and Mor Geva. 2024. Can large language models faithfully express their intrinsic uncertainty in words? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7764. Association for Computational Linguistics.

Wencong You and Daniel Lowd. 2022. Towards stronger adversarial baselines through human-AI collaboration. In *Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP*, pages 11–21, Dublin, Ireland. Association for Computational Linguistics.

Xinyan Yu, Sewon Min, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. CREPE: Open-domain question answering with false presuppositions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10457–10480, Toronto, Canada. Association for Computational Linguistics.

Quan Yuan, Mehran Kazemi, Xin Xu, Isaac Noble, Vaiva Imbrasaite, and Deepak Ramachandran. 2023. Tasklama: Probing the complex task understanding of language models. *arXiv preprint arXiv:2308.15299*.

Matei Zaharia, Omar Khattab, Lingjiao Chen, Jared Quincy Davis, Heather Miller, Chris Potts, James Zou, Michael Carbin, Jonathan Frankle, Naveen Rao, and Ali Ghodsi. 2024. The shift from models to compound ai systems. `https://bair.berkeley.edu/blog/2024/02/18/compound-ai-systems/`.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. Using "annotator rationales" to improve machine learning for text categorization. In *Human Language Technologies 2007: The*

Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, pages 260–267, Rochester, New York. Association for Computational Linguistics.

Savvas Zannettou, Sotirios Chatzis, Kostantinos Papadamou, and Michael Sirivianos. 2018. The good, the bad and the bait: Detecting and characterizing clickbait on youtube. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 63–69. IEEE.

Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua Tenenbaum, Tianmin Shu, and Chuang Gan. Building cooperative embodied agents modularly with large language models. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.

Kaitlyn Zhou, Jena Hwang, Xiang Ren, and Maarten Sap. 2024. Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3623–3643, Bangkok, Thailand. Association for Computational Linguistics.

Kaitlyn Zhou, Dan Jurafsky, and Tatsunori Hashimoto. 2023. Navigating the grey area: How expressions of uncertainty and overconfidence affect language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5506–5524, Singapore. Association for Computational Linguistics.

Yanbin Zhuang. 2025. The influence of artificial intelligence on labor markets. In *SHS Web of Conferences*, volume 218, page 03030. EDP Sciences.