DECISIVE: Guiding User Decisions with Optimal Preference Elicitation from Unstructured Documents

Akriti Jain[1*] **Anish Mulay**[2*†] **Divyansh Verma**[3*†] **Aishani Pandey**[4†]
Pritika Ramu[5‡] **Aparna Garimella**[1]

[1]Adobe Research, India [2]IIT Madras [3]IIT Roorkee
[4]IIIT Hyderabad [5]University of Maryland, College Park
{akritij, garimell}@adobe.com

Abstract

Decision-making is a cognitively intensive task that requires synthesizing relevant information from multiple unstructured sources, weighing competing factors, and incorporating subjective user preferences. Existing methods, including large language models and traditional decision-support systems, fall short: they often overwhelm users with information or fail to capture nuanced preferences accurately. We present DECISIVE, an interactive decision-making framework that combines document-grounded reasoning with Bayesian preference inference. Our approach grounds decisions in an objective option-scoring matrix extracted from source documents, while actively learning a user's latent preference vector through targeted elicitation. Users answer pairwise tradeoff questions adaptively selected to maximize information gain over the final decision. This process converges efficiently, minimizing user effort while ensuring recommendations remain transparent and personalized. Through extensive experiments, we demonstrate that our approach significantly outperforms both general-purpose LLMs and existing decision-making frameworks achieving up to **20%** improvement in decision accuracy over strong baselines across domains.

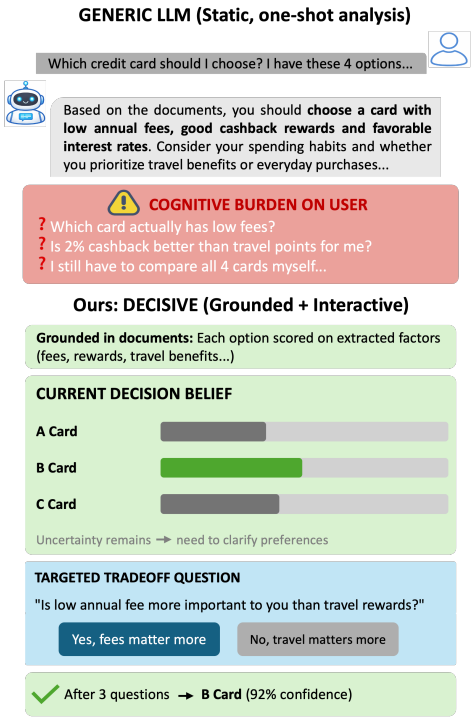

Figure 1: Comparison of decision support approaches. **Top**: A generic LLM provides abstract advice (e.g., "choose a plan with low fees and high rewards"), forcing the user to manually verify which option fits. **Bottom**: DECISIVE grounds recommendations in document-extracted scores and actively elicits preferences through targeted tradeoff questions to find the optimal choice.

1 Introduction

Making informed decisions in high-stakes domains, such as selecting a suitable credit card, choosing an educational program, or defining a corporate strategy, requires synthesizing vast amounts of unstructured information from diverse document collections (e.g., financial disclosures, user reviews, technical reports). These decisions are inherently multi-objective, demanding that users weigh competing factors like cost versus quality or short-term gain versus long-term stability. While retrieving relevant information is a necessary first step, a sound decision-making process requires more than just locating passages. It requires the decision-relevant factors to be systematically extracted and consistently evaluated, enabling users to weigh trade-offs that align with their specific, often latent, goals.

To manage such complexity, AI-powered decision support is now widespread. In finance, models optimize investment portfolios (Li et al., 2024); in healthcare, they assist with clinical diagnosis (Nazi and Peng, 2024); and in strategic planning, they support complex reasoning (Changeux and Mon-

[*]Equal Contribution
[†]Work done during internship at Adobe Research
[‡]Work done while at Adobe Research

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 31774–31786
July 2-7, 2026

tagnier, 2024). Large Language Models (LLMs) have transformed this landscape, offering powerful capabilities to process the vast quantities of unstructured text central to these decisions. Yet despite this promise, current LLM-based solutions face significant limitations. Off-the-shelf LLMs typically act as "static information providers", delivering generic, one-shot analyses that fail to incorporate the user's unique context or latent preferences (Fig.1). Consequently, the cognitive burden of synthesis and alignment is forced back onto the human, who must interpret the AI's output and manually map it to their personal objectives. This not only negates potential efficiency gains but also fosters mistrust in systems that remain opaque and misaligned with user goals. The failure to actively model user preferences is a critical gap; indeed, recent work identifies the user's mental model as a key determinant for improving both the quality of and reliance on LLM-assisted decisions (Eigner and Händler, 2024). Conversely, recent frameworks like DeLLMa (Liu et al., 2024) and DecisionFlow (Chen et al., 2025) assume idealized scenarios where all relevant information is explicitly stated and a single objective governs the choice. This overlooks two critical realities of complex decision-making: (1) evidence is often scattered across multiple unstructured documents, and (2) the optimal choice depends heavily on user-specific constraints (e.g., budget limits or career goals) that are frequently latent and must be actively elicited. We argue that effective decision support must explicitly decouple two distinct problems. First, **factual evaluation**, where options are objectively scored against decision-relevant factors extracted from documents, a process that is constant across users. Second, **preference modeling**, where the system actively learns the user's subjective priorities, a process unique to each individual. By separating objective evidence from subjective goals, we can build systems that are both robust and highly personalized.

To this end, we introduce DECISIVE, an interactive framework that combines document-grounded reasoning with Bayesian preference inference. DECISIVE first extracts an objective Option-Scoring Matrix from unstructured documents. It then employs an active elicitation strategy, asking targeted tradeoff questions to refine a probabilistic model of the user's preferences. This allows the system to converge on a high-utility recommendation. To summarize, our primary contributions are:

1. We introduce DECISIVE, a framework that decouples decision-making into objective document grounding (via an Option-Scoring Matrix) and subjective preference modeling. This separation allows for robust handling of multi-objective trade-offs.
2. We propose a decision-aware active elicitation strategy that selects questions to maximize information gain over the final decision, ensuring efficient convergence with minimal user effort.
3. We curate a challenging, realistic benchmark spanning Finance, Education, and Hiring domains, requiring synthesis from unstructured documents. Extensive experiments show that DECISIVE significantly outperforms strong LLM baselines and structured frameworks in both accuracy and efficiency.

2 Related Works

2.1 Decision-making using LLMs

Large Language Models (LLMs) have demonstrated significant potential as decision support tools across diverse fields like business (Changeux and Montagnier, 2024; Simchi-Levi et al., 2025), finance (Li et al., 2024; Yu et al., 2024), and medicine (Nazi and Peng, 2024; Li et al., 2025; Maity and Saikia, 2025; Kim et al., 2024; Gumilar et al., 2024; Foo et al., 2025). However, research indicates that direct prompting often yields poor results as problem complexity increases, as models may fixate on specific information without adequately balancing evidence or aligning with user goals. To address these limitations, several structured frameworks have been proposed to ground LLM reasoning in formal theory.

One prominent direction involves integrating decision theory and utility maximization. The DeLLMa framework (Liu et al., 2024) guides LLMs through a multi-step process involving state enumeration, probabilistic forecasting, and utility elicitation to identify decisions that maximize expected utility. Building on this, DecisionFlow (Chen et al., 2025) transforms natural language scenarios into structured representations of actions, attributes, and constraints, inferring a latent utility function in a transparent manner. While these frameworks represent a significant step forward, they predominantly operate in idealized scenarios with explicit information and single objectives (e.g., profit maximization). As discussed ear-

lier, this simplifies the dual challenge of real-world decision-making and highlights the need for systems that can both navigate complex information landscapes and actively elicit the nuanced, latent preferences necessary to resolve these trade-offs.

The interactive and collaborative nature of decision-making is also a critical area of exploration. Systems like ChoiceMates (Park et al., 2025) use multi-agent conversational interactions to help users discover diverse perspectives and construct personalized preference spaces. They emphasize keeping the user in the loop to prevent the loss of agency that often accompanies full automation. However, purely conversational approaches often struggle to provide precise, opinionated recommendations, yielding vague answers that fail to satisfy specific user needs (Park et al., 2025). Without explicit preference modeling, these systems lack a mechanism to ensure recommendations reflect the user's priorities rather than the model's implicit tendencies (Jia et al., 2024). DECISIVE addresses these challenges by grounding decisions in real-world unstructured documents (for precision) while keeping the user in the loop to elicit latent preferences (for alignment).

2.2 Preference Elicitation and Clarification Questions

The ability of LLMs to resolve ambiguity through interaction is well-studied in literature. Early work focused on reactive clarification, where models were trained or prompted to identify underspecified queries and ask follow-up questions to resolve implicit assumptions (Zhang and Choi, 2025; Chang, 2025). While effective for simple disambiguation, these approaches often struggle in complex decision-making scenarios where the missing information (the user's latent preference structure) is abstract and high-dimensional.

Recent research has shifted towards proactive information gathering. Several benchmarks, such as InfoQuest (de Oliveira et al., 2025) and latent preference discovery tasks (Tsaknakis et al., 2025), have highlighted that off-the-shelf LLMs are inefficient at this process. They often fail to select the most informative questions, leading to long, unfocused interactions that result in preference dilution rather than convergence. To address this, newer methods employ specialized training paradigms (Andukuri et al., 2024; Dou and Liu, 2025) that use self-improvement and trajectory optimization to reward effective questioning strategies, while other works leverage diffusion-inspired denoising via sequential funnel questions (Montazeralghaem et al., 2025). However, these approaches optimize for reconstructing a general user profile rather than resolving the specific decision at hand. Consequently, they may spend interaction effort on preferences that do not differentiate the available options.

A complementary direction, most relevant to our work, augments LLMs with explicit probabilistic reasoning. Foundational work in Bayesian preference elicitation (Guo and Sanner, 2010) established multi-attribute preference learning with pairwise comparison queries, and recent methods have built on this by combining LLMs with information-theoretic objectives: Active Preference Inference (Piriyakulkij et al., 2024) selects questions that maximize entropy reduction, the OPEN framework (Handa et al., 2024) uses Bayesian Optimal Experimental Design to track user persona distributions, and Austin et al. (2024) combine Bayesian optimization with LLM-based acquisition functions for natural language elicitation. However, these methods are often limited to binary Yes/No queries and frequently default to direct option comparisons (e.g., "Do you prefer Option A or B?"), which assumes the user already understands the trade-offs the system is meant to surface. The Multi-Attribute Decision Making literature (Pu and Chen, 2005, 2008) reinforces this concern, showing that users construct preferences *through* interaction rather than arriving with fixed ones, and that interactive tradeoff support can improve decision accuracy by up to 57%. DECISIVE builds on these insights but introduces a key distinction: rather than reducing uncertainty over preferences alone, it optimizes for reducing uncertainty in the final *decision*, focusing elicitation on the specific trade-offs that distinguish the top candidates.

3 Dataset Curation

Existing benchmarks such as MTA (Hu et al., 2024) and DeLLMa (Liu et al., 2024) frame tasks around a single objective (e.g., maximizing profit) and provide contexts where all necessary information is explicitly stated. This sidesteps two core difficulties of real-world decisions: (1) locating relevant details scattered across multiple documents, and (2) weighing competing attributes (e.g., a university's research rankings or program structure) against personal constraints (e.g., tuition limits). To address

this gap, we curate a dataset spanning three domains: **Finance**, **Education**, and **Hiring**. These correspond to multi-objective decisions commonly faced by knowledge workers, students, and HR professionals, respectively. As identified by prior surveys, these domains serve as representative settings for AI-assisted decision support, with applications in financial advising, educational planning, and career-related decision making (Ma et al., 2025; Albashrawi, 2025; Yin et al., 2025). Each domain requires synthesizing information across multiple documents and reasoning about trade-offs that depend on user-specific goals and constraints.

3.1 Data Sourcing

Curating coherent document collections at scale presents significant challenges. Scraping documents from the web (e.g., tens of university brochures or loan policies) is impractical due to inconsistent formatting, access restrictions, and highly variable information granularity across sources. Instead, we adopt a hybrid sourcing strategy. For the **Hiring** domain, we use resumes from a public Kaggle dataset (Bhawal, 2021), also used by other works (Veldanda et al., 2023; Wang et al., 2024). In this dataset, resumes are organized by category of occupation (e.g., engineer, consultant). For each scenario, we randomly select 10 resumes from the relevant occupational category and generate the decision question taking into account that particular role. For **Finance** and **Education**, we develop a synthetic generation pipeline grounded in manually scraped seed documents and decision scenarios. These seeds, drawn from authentic sources (e.g., real loan brochures, university prospectuses), serve as few-shot examples that anchor the generation in realistic structures and terminology.

3.2 Synthetic Data Generation Pipeline

Stage 1: Question and Schema Generation. We first manually curate a set of seed decision questions representative of each domain. Using these as few-shot examples, we prompt GPT-4o to generate diverse decision scenarios (e.g., "Which graduate program best aligns with my career goals?"). A total of 500 questions are generated per domain (1,500 across three domains). For the Finance and Education domains, the model also produces a structured document schema along with the decision question, specifying the necessary documents for making an informed decision, such as program borchures etc. Each scenario includes M=10 candidate options, deliberately increasing the decision complexity beyond pre-existing benchmarks (which typically use 2-4 options) and ensuring that simple heuristics are insufficient. Since our preference elicitation objective operates over factors rather than options, the framework scales naturally with M. We further verify that performance generalizes to smaller candidate sets through an ablation at M=5 (Appendix A.3).

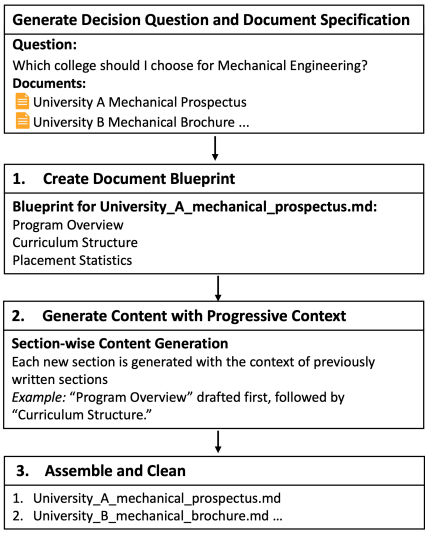

Figure 2: Overview of the data generation process.

Stage 2: Context-Aware Document Synthesis. The second stage synthesizes the documents defined in the schema. To ensure intra-document consistency, the pipeline performs a hierarchical, context-aware generation process: first generating a structural blueprint, then sequentially producing content where each section is conditioned on prior sections, and finally refining the compiled document for stylistic uniformity (see Appendix A.1 for details). The output is a dataset where each instance consists of a decision question and a set of option documents.

Validation. To ensure realism, we recruit three domain experts per field from a freelancing platform,[1] to validate whether the decision questions reflect scenarios encountered in professional practice and whether the generated documents contain the information necessary to make an informed decision. They were compensated at a rate of $15/hour. Annotators also provide the key factors they would consider in such decisions, which informs our prompt design for automatic parameter extraction discussed in detail in Section 5.1. After

[1] https://www.upwork.com

filtering, our final dataset comprises of 499 samples in Education, 491 in Finance, and 500 in Hiring, totaling to 1,490 decision scenarios.

4 Problem Formulation

Given a decision query q and a set of candidate options $\mathcal{O} = \{o_1, \ldots, o_M\}$, where each option is described by a collection of unstructured documents $\mathcal{D}$, our goal is to identify the option o^* that best aligns with the user's latent preferences. We assume the utility of an option depends on K decision factors derived from the documents. We decompose the decision into two components: **(1) Option-Scoring Matrix (S):** An $M \times K$ matrix where S_{ij} quantifies the performance of option i on factor j. Derived directly from source documents, this matrix transforms unstructured text into a structured, comparable representation of evidence for each option. **(2) User Preference Vector ($\mathbf{w}^*$):** A K-dimensional vector where w_j^* reflects the user's subjective importance for factor j. This vector captures the user's unique trade-offs and priorities, which are unknown a priori and must be elicited through interaction.

The recommended option is the one that maximizes the weighted utility:

$$o^* = \arg\max_i (\mathbf{S}\mathbf{w}^*)_i$$

$\mathbf{S}$ is document-derived and user-agnostic: it can be pre-computed once and remains constant across users, whereas $\mathbf{w}^*$ is latent and must be elicited. This formulation cleanly separates factual evaluation from preference modeling, allowing robust decision-making even with complex, multi-objective trade-offs.

5 Methodology

Our methodology centers on a Bayesian model of user preferences that is iteratively refined using an active learning policy. This approach allows the system to intelligently explore the preference space and converge on a high-utility recommendation with minimal user effort.

5.1 Parameter Extraction and Option-Scoring Matrix Construction

Given a decision query q and documents $\mathcal{D}$, an LLM (GPT-5) first extracts K decision-relevant factors $\mathcal{F}$ such as tuition cost or risk tolerance. K is not fixed globally but is extracted per scenario, tailored to the specific decision question at hand. In practice, we observe an overall average of K=11.2 factors per scenario, with domain-level averages of 12.0 for Education (range 8-17), 13.4 for Finance (range 8-21), and 8.3 for Hiring (range 5-14).

For each option-factor pair (o_i, f_j), the system generates a qualitative assessment on an 8-point ordinal scale ("Very Low", "Low", "Low to Medium", "Medium", "Medium to High", "High", "High to Very High", "Very High") grounded in the source documents, which is then mapped to a numerical value in $[0, 1]$ to form the matrix $\mathbf{S}$. This graded, comparative scoring is distinct from standard binary relevance judgment (Upadhyay et al., 2024), as our task requires nuanced relative comparison across options. To ensure robustness and mitigate individual model bias, we generate these scores using three distinct LLMs (GPT-5, Gemini-3-Pro, Claude-4.5-Sonnet) and combine them via median aggregation on the ordinal scale.

5.2 Modeling User Preferences

With $\mathbf{S}$ fixed, the system must discover the user's latent preference vector $\mathbf{w}^*$. A single LLM estimate of user preferences can be unreliable. To address this, we model uncertainty over preferences using a uniform Dirichlet prior ($\alpha_j = 1$), the standard uninformative prior in Bayesian inference, which assumes no initial bias toward any particular preference structure. We then sample P candidate preference vectors $\{\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(P)}\}$ from this prior, each representing a plausible user "persona" with an equal initial likelihood weight $\pi^{(p)} = 1/P$. This particle-based approximation (Doucet and Johansen, 2011; Arulampalam et al., 2002) allows us to tractably maintain and update a rich distribution over possible user preferences. This set of weighted personas represents our initial belief state about the user's preferences.

5.3 Interactive Preference Refinement

To refine our belief state, the system interacts with the user through simple pairwise tradeoff questions. When a user indicates that factor a is preferred over factor b ($f_a \succ f_b$), we perform a Bayesian update on the likelihoods of our sampled personas. The update reinforces personas consistent with the user's feedback and down-weights those that are not. Importantly, inconsistent personas are down-weighted but retained, ensuring that an occasional noisy or contradictory user response does not destroy information but simply redistributes belief across the

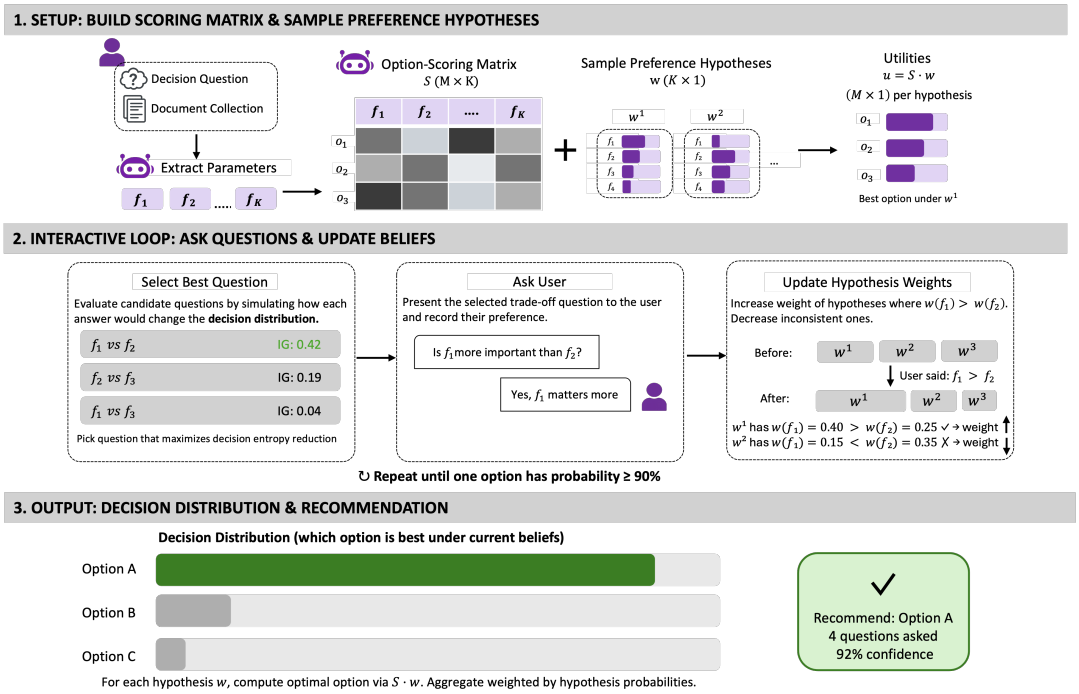

Figure 3: Overview of the DECISIVE pipeline. The system first extracts K preference factors from documents and constructs the Option-Scoring Matrix $\mathbf{S}$, which quantifies how each option performs on each factor. A Dirichlet prior models uncertainty over the user's preference weights, with particle samples representing plausible preference vectors. Decision-aware questions are adaptively selected to maximize information gain over the *decision distribution*. User responses trigger Bayesian updates until decision confidence exceeds a threshold.

preference space. Formally, this is achieved using a sigmoid update rule:

$$\pi^{(p)} \leftarrow \pi^{(p)} \cdot \sigma\Big(\kappa \cdot (w_a^{(p)} - w_b^{(p)})\Big)$$

where $\sigma(\cdot)$ is the sigmoid function and κ controls the sharpness of the update. After each interaction, the likelihoods are re-normalized. This iterative process gradually concentrates the probability mass on the subset of personas most consistent with the user's cumulative responses.

5.4 Decision-Aware Question Selection

Not all questions are equally informative. To maximize efficiency, we select questions that reduce uncertainty in the *final decision*, rather than just in the preference weights. We define the system's current belief as a probability distribution $\boldsymbol{\chi} \in \mathbb{R}^M$ over the available options, where χ_i represents the probability that option i is optimal, marginalized over all sampled preference personas:

$$\chi_i = \sum_{p=1}^{P} \pi^{(p)} \cdot \mathbb{I}\left[\arg\max_{i'} (\mathbf{S}\mathbf{w}^{(p)})_{i'} = i\right]$$

We quantify the uncertainty of this distribution using its entropy $H(\boldsymbol{\chi})$. For each candidate question comparing factors a and b, we simulate both possible user responses ($f_a \succ f_b$ and $f_b \succ f_a$) and compute the expected posterior entropy of the decision distribution. The system selects the question that maximizes the Expected Information Gain (EIG), defined as the reduction in decision entropy:

$$\text{EIG}(a, b) = H(\boldsymbol{\chi}) - \mathbb{E}_{resp}[H(\boldsymbol{\chi}^{(resp)})]$$

This formulation ensures the system focuses only on preference trade-offs that actually differentiate the top contenders, effectively ignoring factors that do not impact the final recommendation.

5.5 Stopping and Recommendation

Instead of asking a fixed number of questions, we employ a dynamic stopping criterion based on decision stability. The dialogue terminates when the confidence in the most likely option exceeds a threshold ($\max_i \chi_i \geq \tau$, with $\tau = 0.85$). Upon stopping, it recommends the option with the highest expected utility under the final posterior belief:

$$o^* = \arg\max_i \sum_{p=1}^{P} \pi_{\text{final}}^{(p)} \cdot (\mathbf{S}\mathbf{w}^{(p)})_i$$

This ensures the final recommendation is grounded in both the objective evidence from the documents

and the user's interactively refined preferences.

6 Experimental Setup

6.1 Evaluation Protocol and Metrics

For each decision scenario, we simulate user interactions by sampling ground-truth preference vectors $\mathbf{w}^*$ from a symmetric Dirichlet distribution. This ensures diverse preference profiles across trials. The ground-truth decision is computed as $o^* = \arg\max_i(\mathbf{S}\mathbf{w}^*)_i$. During interaction, a deterministic user simulator answers tradeoff questions by comparing the corresponding weights: given a question "f_a vs f_b?", it returns f_a if $w_a^* > w_b^*$, else f_b. For our method, we use $P = 500$ preference personas sampled from the same Dirichlet prior, which our ablation study (Sec. 7.1) shows provides a strong accuracy-efficiency trade-off.

Our evaluation focuses on three aspects: decision accuracy, ranking quality, and interaction efficiency. For decision accuracy, we measure both *Top-1 Accuracy* (percentage of times the top recommendation matches ground truth) and *Top-2 Accuracy*, also known as Hit Rate@2, (ground truth appears in top two). For ranking quality, we use *NDCG@3* and *Mean Reciprocal Rank (MRR)*. For efficiency, we measure *Average Questions* asked before reaching a decision, which serves as a proxy for user effort.

6.2 Baselines

We use two categories of baselines:

Decision Frameworks. We compare against three recent works: two frameworks for structured decision-making and one for interactive preference elicitation. **DeLLMa** (Liu et al., 2024): A multi-step reasoning framework that optimizes decisions under uncertainty. We adapt it to our setting by providing the ground-truth preference profile $\mathbf{w}^*$ as part of the context. **DecisionFlow** (Chen et al., 2025): A system that structures decisions by modeling actions and attributes. Similar to DeLLMa, it is provided with the ground-truth preference profile. **Active Preference Inference** (Piriyakulkij et al., 2024): An interactive method that enables an LLM to ask questions to infer user preferences. Like our method, it starts without knowledge of $\mathbf{w}^*$ and must elicit it through dialogue.

Prompting Baselines. These baselines assess the capability of LLMs (GPT-4o) to solve the task directly with varying degrees of assistance. **LLM-Direct (B1):** A single-shot baseline that receives all inputs (documents, decision question, extracted parameters, and the option-scoring matrix, along with the ground-truth user preference vector $\mathbf{w}^*$). It is prompted to directly output its recommended decision. **LLM-CoT (B2):** Identical to B1 but utilizes Chain-of-Thought (CoT) prompting to encourage reasoning before the final decision. **LLM-Structured Dialogue (B3):** An interactive baseline where the LLM does *not* receive $\mathbf{w}^*$. Instead, it must discover user preferences by asking up to 10 yes/no questions before making a final recommendation. This mirrors the setting of our method but relies on the LLM's implicit ability to infer preferences. **LLM-Free Dialogue (B4):** Similar to B3 but the LLM decides what questions to ask, how many, and when to stop. This tests whether unconstrained dialogue improves preference inference. Exact prompts for all baselines are provided in Appendix A.2.

7 Results and Discussion

Performance vs. Prompting Methods. Given the ground-truth preference vector, LLM baselines (B1, B2) perform reasonably well, achieving 62-67% accuracy on Education and Finance, and ∼87% on the Hiring domain where candidate differentiations are more apparent (Table 1). That said, their accuracy is capped by the need to reason over a $10 \times K$ scoring matrix (where $K \approx 11$) and compute weighted preference combinations, which remains challenging due to known LLM limitations in multi-step numerical reasoning. When preferences are unknown (B3, B4), performance collapses across all domains (9-30%), confirming that LLMs struggle to implicitly infer and track user preferences through dialogue (Zhao et al., 2025; Tsaknakis et al., 2025). DECISIVE addresses this by offloading preference tracking to an explicit Bayesian model, achieving ∼79% on Education and Finance, and 90% on Hiring, **consistently outperforming** the fully-informed baselines.

Performance vs. Decision Frameworks. Existing structured frameworks struggle on our benchmark. DeLLMa and DecisionFlow achieve only 9-28% accuracy despite being provided with the ground-truth preference profile. These frameworks assume a simplified decision structure where a single metric dominates, reducing the decision to a direct optimization problem. Our benchmark, by contrast, requires reasoning over multi-objective trade-offs where users must balance competing factors. Ac-

Method	Dialogue?	Top-1 Acc.	Top-2 Acc.	NDCG@3	MRR	Avg. Qs
Education Domain						
DECISION FRAMEWORKS						
DeLLMa	✗	28.1	44.3	0.733	0.480	–
DecisionFlow	✗	20.2	34.3	0.706	0.403	–
Active Pref. Inference	✓	13.0	25.9	0.679	0.336	9.7
PROMPTING BASELINES						
LLM-Direct (B1)	✗	64.8	82.4	0.799	0.782	–
LLM-CoT (B2)	✗	65.5	83.5	0.804	0.786	–
LLM-Structured Dialogue (B3)	✓	9.4	22.2	0.225	0.302	4.7
LLM-Free Dialogue (B4)	✓	19.4	34.7	0.345	0.398	2.8
DECISIVE (Ours)	✓	78.8	91.6	0.893	0.879	5.1
Finance Domain						
DECISION FRAMEWORKS						
DeLLMa	✗	24.0	39.3	0.719	0.446	–
DecisionFlow	✗	17.1	30.1	0.694	0.373	–
Active Pref. Inference	✓	9.8	18.9	0.667	0.276	9.7
PROMPTING BASELINES						
LLM-Direct (B1)	✗	62.3	83.5	0.793	0.773	–
LLM-CoT (B2)	✗	67.0	85.5	0.827	0.803	–
LLM-Structured Dialogue (B3)	✓	13.4	26.3	0.265	0.336	4.1
LLM-Free Dialogue (B4)	✓	20.2	35.2	0.364	0.410	2.3
DECISIVE (Ours)	✓	79.0	91.6	0.887	0.893	6.1
Hiring Domain						
DECISION FRAMEWORKS						
DeLLMa	✗	9.8	20.0	0.667	0.288	–
DecisionFlow	✗	9.2	20.4	0.665	0.292	–
Active Pref. Inference	✓	32.0	56.6	0.749	0.554	7.5
PROMPTING BASELINES						
LLM-Direct (B1)	✗	87.2	97.2	0.942	0.931	–
LLM-CoT (B2)	✗	87.8	97.8	0.949	0.935	–
LLM-Structured Dialogue (B3)	✓	11.4	21.8	0.229	0.303	6.9
LLM-Free Dialogue (B4)	✓	31.6	40.6	0.416	0.472	2.4
DECISIVE (Ours)	✓	90.2	97.6	0.958	0.947	1.9

Table 1: Performance comparison on Education, Finance, and Hiring domains. We compare DECISIVE against LLM baselines and existing decision-making frameworks. The "Dialogue" column indicates whether the method interactively elicits preferences from the user. Green highlights best performance; Red highlights second-best.

tive Preference Inference, the only other interactive method, shows highly variable performance (10-32% across domains), highlighting the limitations of generic entropy-based question selection. Our *decision-aware* strategy, which focuses on factors that differentiate the top options, yields more consistent and substantially higher accuracy.

7.1 Ablation Studies

Effect of Number of Preference Profiles. We analyze how the number of sampled preference profiles (P) affects accuracy and inference time (Table 2). P can be seen as a resolution parameter for our Monte Carlo estimator, where higher values yield a finer-grained approximation of the posterior. Accuracy improves steadily as P increases: from 66.9% at $P = 10$ to 82.7% at $P = 500$. Beyond this point, gains become marginal (83.9% at $P = 2000$) while inference time continues to grow, validating that the EIG estimates at $P = 500$ are stable enough to select effective questions.

Profiles (P)	Top-1 Acc.	Top-2 Acc.	Time (s)	Avg. Qs
10	66.9	84.5	0.021	2.41
100	77.5	89.5	0.086	3.23
500	82.7	93.7	0.533	4.24
1000	82.9	92.6	0.849	4.01
2000	83.9	93.4	1.289	4.54

Table 2: Effect of number of preference profiles on accuracy and inference time averaged across all domains.

We use $P = 500$ for our main experiments, which achieves strong accuracy in under 600ms per scenario averaged across all domains.

Robustness to Noisy Responses. To validate robustness to inconsistent user responses, we conduct an ablation using a Bradley-Terry response model where the simulated user occasionally gives inconsistent answers. We compare deterministic responses against noisy responses (temperature =

0.05) across all three domains (Table 3).

Temp	Domain	Top-1	Top-2	NDCG@3	MRR	Avg Qs
None	Education	78.8	91.6	0.893	0.879	5.1
None	Finance	79.0	91.6	0.887	0.861	6.1
None	Hiring	90.2	97.8	0.958	0.947	1.9
0.05	Education	76.6	91.0	0.881	0.866	5.5
0.05	Finance	76.8	90.4	0.880	0.861	6.2
0.05	Hiring	89.6	97.4	0.955	0.943	1.8

Table 3: Noise robustness ablation. Performance under deterministic (Temp=None) vs. noisy (Temp=0.05) Bradley-Terry user responses.

Performance remains largely stable under noise: average Top-1 drops only $\sim$2-2.5% and Top-2 drops $\sim$1%, with Hiring remaining particularly robust. These results confirm that even when user responses deviate from the model's assumptions, the soft reweighting mechanism prevents catastrophic information loss.

Extensibility to Nuanced Responses. A key advantage of our probabilistic formulation is its extensibility to nuanced user feedback. Because we model preferences as distributions over weight vectors, the framework can naturally accommodate responses beyond simple binary choices. For instance, when a user indicates two factors are *equally important* or they don't have any strong preference for either (neutral), the update rule can preserve the prior over those factors. When *both* are highly valued, both can be upweighted proportionally. This flexibility is not possible in methods like Active Preference Inference, which rely on hard binary elimination.

7.2 Computational Efficiency

Question Efficiency. DECISIVE requires only 5-6 questions on average to reach a confident decision in Education and Finance, and under 2 questions in Hiring where candidates are more clearly differentiated. This is fewer than Active Preference Inference ($\sim$7.5-9.7 questions) and comparable to B3 ($\sim$4-7 questions), but with substantially higher accuracy. Our dynamic stopping criterion terminates the dialogue only when the *decision distribution* is confident, avoiding wasted effort on preferences that do not impact the final choice.

LLM Call Efficiency. DecisionFlow's multi-step pipeline (extraction, attribute mapping, weight computation per option-attribute pair) requires dozens of LLM calls per scenario. DeLLMa and the dialogue baselines (B3, B4, Active Preference) each require 7-10 calls. In contrast, DECISIVE requires LLM calls only for initial parameter extraction and Option-Scoring Matrix construction. The subsequent Bayesian inference and question selection phases are purely computational, requiring **zero LLM calls** during user interaction. This design makes our approach significantly more efficient and cost-effective at inference time.

8 Conclusion

We present DECISIVE, a framework that transforms the passive retrieval of unstructured information into an active, personalized decision-making process. By explicitly decoupling factual evidence extraction from subjective preference modeling, our approach addresses the twin challenges of information overload and preference ambiguity. Our results reveal three key findings: (1) Structured preference integration significantly improves accuracy over text-based reasoning. (2) Dialogue alone is insufficient for preference discovery without explicit state tracking. (3) Decision-aware elicitation enables efficient convergence with fewer questions than entropy-based methods. We believe DECISIVE represents a step toward next-generation decision engines that do not merely present information, but actively collaborate with users to assist them in complex, real-world decision-making scenarios.

Limitations

Our framework relies on the quality of the Option-Scoring Matrix, which depends on the extraction capabilities of the underlying LLM. Errors or hallucinations in scoring can propagate to the final decision; we mitigate this by aggregating scores from multiple models, but residual noise may still persist. Additionally, our current utility model assumes that decision factors are independent. Real-world preferences can exhibit interdependencies (e.g., a user might prioritize quality only if price falls below a certain threshold), which a linear model may not fully capture. The framework also extracts a fixed set of K factors per scenario; when users have goals that fall outside the extracted factors (e.g., personal connections or emotional considerations not reflected in documents), the system's recommendations may not fully capture their preferences. Finally, our evaluation uses simulated users with known ground-truth preferences. While we validate robustness under noisy responses via a Bradley-Terry ablation (Section 7.1),aspects such as usability and user satisfaction remain to be validated through human studies.

Ethics Statement

We recognize that automated decision support, particularly in high-stakes domains like hiring, finance, and education, carries significant ethical responsibilities. DECISIVE is designed as an assistive tool to augment, not replace, human agency. By explicitly separating objective evidence extraction from subjective preference modeling, our framework aims to increase transparency and allow users to understand the basis of recommendations. However, we acknowledge that the underlying LLMs used for option scoring may exhibit latent biases, which could propagate to the decision process. While we mitigate this by aggregating scores across multiple models, users should interpret recommendations as data-driven suggestions rather than definitive judgments. Furthermore, all data used in our experiments is either synthetically generated or sourced from public, anonymized repositories (e.g., Kaggle resumes), ensuring that no private individual data was compromised.

References

Mousa Albashrawi. 2025. Generative ai for decision-making: A multidisciplinary perspective. *Journal of Innovation Knowledge*, 10(4):100751.

Chinmaya Andukuri, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah D. Goodman. 2024. Star-gate: Teaching language models to ask clarifying questions.

M. Sanjeev Arulampalam, Simon Maskell, Neil J. Gordon, and Tim Clapp. 2002. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. Signal Process.*, 50:174–188.

David Austin, Anton Korikov, Armin Toroghi, and Scott Sanner. 2024. Bayesian optimization with llm-based acquisition functions for natural language preference elicitation. In *Proceedings of the 18th ACM Conference on Recommender Systems*, RecSys '24, page 74–83, New York, NY, USA. Association for Computing Machinery.

Snehaan Bhawal. 2021. Resume dataset.

Shane Chang. 2025. Teaching AI to Clarify: Handling Assumptions and Ambiguity in Language Models.

Alexander Changeux and Stephen Montagnier. 2024. Strategic decision-making support using large language models (llms). *Management Journal for Advanced Research*, 4(4):102–108.

Xiusi Chen, Shanyong Wang, Cheng Qian, Hongru Wang, Peixuan Han, and Heng Ji. 2025. Decisionflow: Advancing large language model as principled decision maker.

Bryan L. M. de Oliveira, Luana G. B. Martins, Bruno Brandão, and Luckeciano C. Melo. 2025. Infoquest: Evaluating multi-turn dialogue agents for open-ended conversations with hidden context.

Yulin Dou and Jiangming Liu. 2025. To-gate: Clarifying questions and summarizing responses with trajectory optimization for eliciting human preference.

Arnaud Doucet and Adam M. Johansen. 2011. A tutorial on particle filtering and smoothing : fiteen years later. In Dan Crisan and Boris Rozovskii, editors, *The Oxford handbook of nonlinear filtering*, Oxford handbooks in mathematics, pages 656–705. Oxford University Press, Oxford ; N.Y.

Eva Eigner and Thorsten Händler. 2024. Determinants of llm-assisted decision-making.

Charisse Foo, Pin Sym Foong, Camille Nadal, Natasha Ureyang, Thant Naylin, and Gerald Choon Huat Koh. 2025. The benefits and risks of llms for facilitating medical decision-making among laypersons. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference*, DIS '25, page 3173–3191, New York, NY, USA. Association for Computing Machinery.

Khanisyah Erza Gumilar, Birama R. Indraprasta, Ach Salman Faridzi, Bagus M. Wibowo, Aditya Herlambang, Eccita Rahestyningtyas, Budi Irawan, Zulkarnain Tambunan, Ahmad Fadhli Bustomi, Bagus Ngurah Brahmantara, Zih-Ying Yu, Yu-Cheng Hsu, Herlangga Pramuditya, Very Great E. Putra, Hari Nugroho, Pungky Mulawardhana, Brahmana A. Tjokroprawiro, Tri Hedianto, Ibrahim H. Ibrahim, Jingshan Huang, Dongqi Li, Chien-Hsing Lu, Jer-Yen Yang, Li-Na Liao, and Ming Tan. 2024. Assessment of large language models (llms) in decision-making support for gynecologic oncology. *Computational and Structural Biotechnology Journal*, 23:4019–4026.

Shengbo Guo and Scott Sanner. 2010. Real-time multi-attribute bayesian preference elicitation with pairwise comparison queries. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 289–296, Chia Laguna Resort, Sardinia, Italy. PMLR.

Kunal Handa, Yarin Gal, Ellie Pavlick, Noah Goodman, Jacob Andreas, Alex Tamkin, and Belinda Z. Li. 2024. Bayesian preference elicitation with language models.

Brian Hu, Bill Ray, Alice Leung, Amy Summerville, David Joy, Christopher Funk, and Arslan Basharat. 2024. Language models are alignable decision-makers: Dataset and application to the medical triage domain.

Jingru Jia, Zehua Yuan, Junhao Pan, Paul E. McNamara, and Deming Chen. 2024. Decision-making behavior evaluation framework for llms under uncertain context.

Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik Siu Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae Won Park. 2024. Mdagents: An adaptive collaboration of llms for medical decision-making.

Haohang Li, Yupeng Cao, Yangyang Yu, Shashidhar Reddy Javaji, Zhiyang Deng, Yueru He, Yuechen Jiang, Zining Zhu, Koduvayur Subbalakshmi, Guojun Xiong, Jimin Huang, Lingfei Qian, Xueqing Peng, Qianqian Xie, and Jordan W. Suchow. 2024. Investorbench: A benchmark for financial decision-making tasks with llm-based agent.

Jia Li, Zichun Zhou, Han Lyu, and Zhenchang Wang. 2025. Large language models-powered clinical decision support: enhancing or replacing human expertise? *Intelligent Medicine*, 5(1):1–4.

Ollie Liu, Deqing Fu, Dani Yogatama, and Willie Neiswanger. 2024. Dellma: Decision making under uncertainty with large language models.

Shuai Ma, Qiaoyi Chen, Xinru Wang, Chengbo Zheng, Zhenhui Peng, Ming Yin, and Xiaojuan Ma. 2025. Towards human-ai deliberation: Design and evaluation of llm-empowered deliberative ai for ai-assisted decision-making. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, New York, NY, USA. Association for Computing Machinery.

S. Maity and M. J. Saikia. 2025. Large language models in healthcare and medical applications: A review. *Bioengineering (Basel)*, 12(6):631.

Ali Montazeralghaem, Guy Tennenholtz, Craig Boutilier, and Ofer Meshi. 2025. Asking clarifying questions for preference elicitation with large language models.

Zabir Al Nazi and Wei Peng. 2024. Large language models in healthcare and medical domain: A review.

Jeongeon Park, Bryan Min, Kihoon Son, Jean Y. Song, Xiaojuan Ma, and Juho Kim. 2025. Choicemates: Supporting unfamiliar online decision-making with multi-agent conversational interactions.

Wasu Top Piriyakulkij, Volodymyr Kuleshov, and Kevin Ellis. 2024. Active preference inference using language models and probabilistic reasoning.

Pearl Pu and Li Chen. 2005. Integrating tradeoff support in product search tools for e-commerce sites. In *Proceedings of the 6th ACM Conference on Electronic Commerce*, EC '05, page 269–278, New York, NY, USA. Association for Computing Machinery.

Pearl Pu and Li Chen. 2008. User-involved preference elicitation for product search and recommender systems. *AI Mag.*, 29(4):93–103.

David Simchi-Levi, Konstantina Mellou, Ishai Menache, and Jeevan Pathuri. 2025. Large language models for supply chain decisions.

Ioannis Tsaknakis, Bingqing Song, Shuyu Gan, Dongyeop Kang, Alfredo Garcia, Gaowen Liu, Charles Fleming, and Mingyi Hong. 2025. Do llms recognize your latent preferences? a benchmark for latent information discovery in personalized interaction.

Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Nick Craswell, and Jimmy Lin. 2024. Umbrela: Umbrela is the (open-source reproduction of the) bing relevance assessor.

Akshaj Kumar Veldanda, Fabian Grob, Shailja Thakur, Hammond Pearce, Benjamin Tan, Ramesh Karri, and Siddharth Garg. 2023. Are emily and greg still more employable than lakisha and jamal? investigating algorithmic hiring bias in the era of chatgpt.

Ze Wang, Zekun Wu, Xin Guan, Michael Thaler, Adriano Koshiyama, Skylar Lu, Sachin Beepath, Ediz Ertekin, and Maria Perez-Ortiz. 2024. JobFair: A framework for benchmarking gender hiring bias in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3227–3246, Miami, Florida, USA. Association for Computational Linguistics.

Shi Yin, Raiha Imran, Kifayat Ullah, Zeeshan Ali, and Izatmand Haleemzai. 2025. Responsible AI in student management: preventing misdecision in career choice of university students under inaccurate guidance. *Scientific Reports*, 15(1):38177.

Yangyang Yu, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yupeng Cao, Zhi Chen, Jordan W. Suchow, Rong Liu, Zhenyu Cui, Zhaozhuo Xu, Denghui Zhang, Koduvayur Subbalakshmi, Guojun Xiong, Yueru He, Jimin Huang, Dong Li, and Qianqian Xie. 2024. Fincon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making.

Michael JQ Zhang and Eunsol Choi. 2025. Clarify when necessary: Resolving ambiguity through interaction with LMs. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5526–5543, Albuquerque, New Mexico. Association for Computational Linguistics.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. 2025. Do llms recognize your preferences? evaluating personalized preference following in llms.

A Appendix

A.1 Document Generation Pipeline Details

Our context-aware document synthesis (Stage 2) proceeds in three steps for each document:

1. **Blueprint Generation:** The system generates a hierarchical JSON blueprint outlining the document's structure, including its sections and subsections (e.g., Program Overview, Fee Structure, Career Prospects for a university prospectus).

2. **Sequential Content Generation:** Content is generated section by section, where each new section is explicitly conditioned on all previously generated sections. This sequential conditioning maintains factual consistency (e.g., ensuring fee amounts mentioned in one section match those in another) and stylistic coherence across the document.
3. **Compilation and Refinement:** All generated sections are compiled, and a concluding LLM call refines the complete document to ensure stylistic uniformity, remove generation artifacts, and verify internal consistency.

This hierarchical approach ensures that generated documents exhibit the coherence characteristics of real-world documents, avoiding the inconsistencies that often arise from single-shot generation.

A.2 Baseline Prompts

Below we provide the exact prompts used for each prompting baseline. In all prompts, <QUESTION>, <OPTIONS>, <PARAMS>, <G_MATRIX>, <DOCUMENTS>, and <USER_PREFS> are placeholders populated from the scenario data.

A.2.1 B1: LLM-Direct

```
You are a decision-making assistant.

=== DECISION QUESTION ===
<QUESTION>

=== OPTIONS ===
<OPTIONS>

=== USER PREFERENCES ===
<USER_PREFS>

=== PARAMETERS ===
<PARAMS>

=== G-MATRIX (How each option scores on each
    parameter, 0-1 scale) ===
<G_MATRIX>

=== DOCUMENTS ===
<DOCUMENTS>

=== TASK ===
Based on the user's preferences and the
information above, select the best option
for this user.

OUTPUT FORMAT (JSON only):
{
  "ranking": ["best_option", "second_best", ...],
  "decision": "best_option"
}
```

A.2.2 B2: LLM-CoT

Identical to B1, with the following task section replaced:

```
=== TASK ===
Based on the user's preferences and the
information provided, determine which option
is best for this user.

Let's think step by step:
- First, identify which parameters matter most
  to this user based on their preference weights.
- Then, analyze how each option scores on those
  key parameters.
- Finally, weigh the trade-offs and make your
  decision.

Respond with JSON in this format:
{
  "reasoning": "Your step-by-step analysis...",
  "ranking": ["best_option", "second_best", ...],
  "decision": "best_option"
}
```

A.2.3 B3: LLM-Structured Dialogue

The LLM receives the decision context (question, options, parameters, G-matrix, abbreviated documents) but *not* the user's preference vector. It is instructed to ask up to 10 yes/no questions:

```
You are a helpful decision assistant having a
conversation with a user.

=== CONTEXT ===
Decision Question: <QUESTION>
Options: <OPTIONS>
Parameters that matter: <PARAMS>
G-Matrix: <G_MATRIX>
Documents (abbreviated): <DOCUMENTS>

=== YOUR TASK ===
You need to help the user choose the best option.
To do this:
1. Ask the user YES/NO questions to understand
   their preferences
2. After gathering enough information (3-5
   questions), make a recommendation

=== RULES ===
- Ask ONE question at a time
- Questions must be answerable with YES or NO
- Focus on understanding which parameters matter
  most to the user
- When ready to recommend, provide your decision

=== OUTPUT FORMAT ===
For each turn, output JSON:
{
  "type": "question" or "decision",
  "content": "Your question" (if asking),
  "ranking": ["best", "2nd best", ...] (if
    deciding),
  "reasoning": "Brief reasoning"
}
```

A.2.4 B4: LLM-Free Dialogue

Similar to B3, but the LLM has full autonomy over question format, count, and stopping:

```
You are a decision assistant helping a user
make a choice.

=== DECISION CONTEXT ===
Question: <QUESTION>
Options: <OPTIONS>
```

```
Relevant Factors: <PARAMS>
How options score on these factors: <G_MATRIX>
Documents (abbreviated): <DOCUMENTS>

=== YOUR TASK ===
Help the user make the best decision for THEIR
specific needs.

You have FULL AUTONOMY over your strategy:
- You may ask the user questions to understand
  their preferences
- You may ask ANY type of question (open-ended,
  comparison, yes/no, etc.)
- You decide how many questions to ask (could
  be 0 if you think you have enough info)
- You decide when you're ready to make a
  recommendation

The user knows their own preferences but hasn't
studied the options in detail. Your job is to
match their preferences to the best option.

=== OUTPUT FORMAT ===
Respond with JSON:
{
  "action": "ask" or "decide",
  "content": "Your question" (if asking),
  "ranking": ["best", "2nd best", ...] (if
    deciding),
  "reasoning": "Why you're asking / why this
    recommendation"
}
```

A.3 Candidate Set Size Ablation

We ablate the effect of candidate set size by reducing M from 10 to 5, comparing DECISIVE against the two strongest baselines (B1 and B2) on a representative subset (Table 4).

Domain	Method	Top-1	Top-2	NDCG@3	MRR	Avg Qs
Education	DECISIVE	81.0	93.8	0.930	0.881	3.7
Education	B1	75.0	91.0	0.908	0.856	–
Education	B2	78.2	89.0	0.909	0.881	–
Finance	DECISIVE	79.0	95.0	0.922	0.878	3.4
Finance	B1	76.0	91.0	0.911	0.872	–
Finance	B2	77.4	92.3	0.896	0.873	–
Hiring	DECISIVE	93.4	98.0	0.974	0.964	1.2
Hiring	B1	91.0	97.4	0.967	0.955	–
Hiring	B2	92.0	97.6	0.970	0.958	–

Table 4: Effect of candidate set size (M=5). DECISIVE outperforms both fully-informed baselines across all domains, with fewer interaction questions required compared to M=10.

DECISIVE outperforms both baselines even at M=5 while requiring fewer questions. This is expected because our preference elicitation operates over decision factors rather than individual options, allowing the framework to scale naturally with M. In contrast, prompting-based baselines must fit the entire option set and associated documents into the context window, which becomes increasingly limiting as M grows. Notably, the gains over prompting baselines are more pronounced at larger M (Table 1), where simple prompting-based approaches struggle to jointly reason over all candidates.