CEUR-WS.org/Vol-3993/paper3.pdf

Agentic CBR in Action: Empowering Loan Approvals Through Interactive, Counterfactual Explanations*

Pedram Salimi[1,*], Nirmalie Wiratunga[2] and David Corsar[3]

Robert Gordon University, Garthdee House, Garthdee Rd, Aberdeen AB10 7AQ

Abstract

Large Language Models (LLMs) have demonstrated impressive conversational capabilities, yet their susceptibility to hallucinations and inconsistent recommendations poses significant risks in high-stakes domains such as finance. This paper presents an interactive chatbot for loan application guidance that leverages a case-based reasoning (CBR) approach to generate actionable counterfactual explanations within an agentic framework. Our system employs a supervisor agent, built using the LangGraph framework, to orchestrate four specialised agents: a classifier agent that provides an initial loan prediction, a causally-aware counterfactual explanation agent that proposes minimal yet feasible modifications to reverse an unfavourable decision, a Feature Actionability Taxonomy (FAT) agent that updates user-specific immutability constraints based on feedback, and a template-based natural language generation (NLG) agent that transforms counterfactual suggestions into clear, user-friendly explanations. A key strength of our design is the automated feedback loop: when users indicate that certain suggestions are unworkable, the FAT agent revises the constraints and instructs the counterfactual generation agent to produce a refined explanation. We detail the system architecture and workflow and outline an experimental plan that compares our full agentic chatbot to ablated variants and a LLM-Only Baseline. And finally we outline a planned user study to evaluate how controlled reasoning affects trust in high-stakes lending.

Keywords

Conversational AI, Counterfactual explanations, Agentic workflow, CBR, Large Language Models, Hallucinations

1. Introduction

Large language models (LLMs) such as GPT-4 have demonstrated remarkable conversational abilities, making them attractive for use as automated assistants in decision-making domains. However, in high-stakes applications like financial loan assessments, the unreliability of LLM outputs, especially their tendency to generate hallucinations, i.e., plausible-sounding but incorrect or unfounded information, poses a significant risk [1]. Users seeking loan advice or explanations for an AI-driven loan decision require accurate and trustworthy information; any incorrect guidance could lead to poor financial decisions or loss of trust [2]. This creates a need for techniques to ground LLM responses in verifiable logic and data.

Explainable AI (XAI) is crucial in lending and other domains to justify algorithmic decisions and provide users with recourse options [3]. Among XAI methods, counterfactual explanations have gained prominence: they answer "What if?" scenarios by describing how a small change in the input features could alter the decision outcome. For example, an applicant might be told, *"If your annual income were $5,000 higher, then your loan would be approved."* Such explanations not only reveal key factors behind a decision but also inform the user of actionable steps to potentially achieve a desired outcome in the future [4].

Despite their usefulness, counterfactual explanations in practice face challenges regarding feasibility and personalisation [5, 6]. Many methods assume any input feature can be freely changed, which is not true for real users, as some features (like age or credit history) cannot be altered, and others can be changed only indirectly or with great effort. Recent work has begun to formalise these constraints via feature actionability taxonomy [7], and to incorporate causal relationships to avoid suggesting

---

2nd Workshop on Case-Based Reasoning and Large Language Model Synergies (CBR-LLM) at ICCBR-2025, June 30, 2025, Biarritz, France

[*] Corresponding author.

p.salimi@rgu.ac.uk (P. Salimi); n.wiratunga@rgu.ac.uk (N. Wiratunga); d.corsar1@rgu.ac.uk (D. Corsar)
0000-0003-4040-2496 (N. Wiratunga); 0000-0001-7059-4594 (D. Corsar)

© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

implausible changes [8]. Another challenge is how to present and interact with such explanations: a static list of numerical feature changes can be overwhelming or confusing to users without additional context or an opportunity to ask questions [9]. In other words, an explanation interface should ideally guide the user through understanding and possibly acting on the recommendations, in a sensitive and interactive manner.

In this paper, we address these challenges by combining case-based reasoning (CBR) with LLM capabilities to create a conversational agent for loan application guidance. CBR, an approach where reasoning is grounded in specific instances or "cases" naturally complements LLMs by providing concrete examples or analogies that the generative model can use to formulate explanations. By retrieving and adapting similar past cases (or generating counterfactual cases), a CBR component can supply factual anchors that reduce the risk of hallucination from the LLM. Meanwhile, the LLM enables a flexible dialogue with the user, clarifying their context and preferences and presenting explanations in fluent natural language.

We present a novel interactive conversational AI system that implements this CBR-LLM synergy in the context of loan application decisions. In this synergy, CBR anchors explanations in concrete cases while the LLM contextualises those cases into conversational, user-specific advice which using the strengths of both symbolic retrieval and generative language. Our system is built with an agentic workflow using a framework called LangGraph, [1] which allows the LLM to act as a controller orchestrating multiple specialised modules. The supervisor loan agent integrates four key agents: (1) an agent with trained loan approval classifier that provides an initial prediction, (2) a causally-aware instance-based counterfactual explanation discovery agent that finds how the unsuccessful input case could change to yield a positive outcome, (3) a Feature Actionability Taxonomy (FAT) agent that encodes user-specific immutability and ethical constraints on feature changes, and (4) a template-based natural language generation (NLG) agent that guide the LLM to convert explanations in a proper format considering fairness and actionability then putting it into user-friendly dialogue responses. By decomposing the task among these modules, we use the strengths of each (e.g., reliable numeric computation and causal logic in the structured modules, and language expressiveness in the LLM) while mitigating the weaknesses of an unconstrained LLM.

In summary, our contributions are as follows:

- We introduce a novel interactive CBR-enhanced LLM agentic framework that ensures modularity: the supervisor agent interacts with dedicated agents for prediction, explanation, updating user constraints and NLG, improving the handling of numerical computations, counterfactual generation, and ethical constraints compared to a monolithic LLM approach.
- We integrate a Feature Actionability Taxonomy into the explanation process, enabling the system to respect user-specific immutable features as well as generating personalised feasible, respectful suggestions.
- We demonstrate how an LLM can serve as a dialogue orchestrator to parsing user intent, invoking specialised tools via LangGraph, and tailoring the final wording, thereby contributing reasoning, tool-use, and adaptive NLG beyond simple template filling
- We outline an evaluation strategy comparing our full system against a LLM-Only Baseline and an ablated version, to quantify the benefits in terms of explanation quality, user trust, and the reduction of hallucinated or infeasible recommendations.

To our knowledge, this is one of the first works to systematically combine case-based reasoning with large language models in an agentic setting, aiming to enhance explainability and reliability in high-stakes decision support. Next, we discuss related literature before detailing the methodology of our system, experimental setup, followed by our planned user study, and conclusions.

---

[1] https://www.langchain.com/langgraph

2. Related Work

Conversational XAI and Interactive CBR. Researchers have increasingly recognised the need for making AI explanations more conversational and interactive. Traditional explanation interfaces (e.g., static texts or visualisations) do not allow users to seek clarification or explore "what-if" scenarios in depth. Conversational XAI attempts to fill this gap by using dialogue to deliver and refine explanations. For instance, Wijekoon et al. [10] propose a CBR-driven interactive XAI approach where users can iteratively query an explanation system and receive case-based clarifications. Their work indicates that users may benefit from explanations that reference similar prior cases or counterfactual examples in a dialogue, which can enhance understanding. Our chatbot follows this paradigm, enabling back-and-forth interaction about a model's decision and potential changes. In contrast to purely scripted dialogues, however, our approach leverages an LLM for flexible natural conversation management, guided by a structured workflow to keep the dialogue grounded.

Counterfactual explanations and actionable recourse. Counterfactual explanations have become a cornerstone of interpretable machine learning [3, 11]. They provide individuals subject to an automated decision with a description of how things could be different to obtain a desirable outcome. Numerous algorithms exist to generate counterfactuals. Some optimise feature perturbations to achieve minimal changes while flipping the model's prediction [12]; others search within a database of past instances for a nearest neighbor with a different outcome [13, 14]. Ensuring these counterfactuals are not only technically valid but also actionable and realistic has been a focus of recent work. Ustun et al. [5] introduced the notion of actionable recourse, emphasising that recommendations should correspond to feasible interventions a user can actually perform. Poyiadzi et al. [6] similarly proposed FACE, which finds feasible counterfactuals that lie on a manifold of plausible data. The idea of incorporating causal constraints is explored by Mahajan et al. [8], who argue that counterfactuals should not violate known causal relationships in the domain (e.g., one should not suggest altering a feature in a way that is causally impossible). Our system builds on these principles by generating counterfactual explanations that are informed by real instances and by filtering or adjusting them according to feasibility constraints (drawing on ideas from [5, 6, 8]). Importantly, we embed this capability within an interactive dialogue, whereas most prior methods assume a single-shot explanation delivery.

LLM augmentation and hallucination mitigation. The advent of powerful LLMs has led to explorations of how they can be integrated with existing AI models or knowledge sources to improve reliability. A key concern is the phenomenon of hallucination in generative models, where the model outputs incorrect information with high confidence [15]. One line of work to mitigate this is retrieval-augmented generation [16], in which the language model is provided with pertinent documents or data retrieved from a knowledge base, ensuring it has factual grounding for its responses. Another line is enabling LLMs to use external tools or calculators for tasks outside their core language abilities [17, 18]. For example, the ReAct framework [17] and Toolformer [18] show that by interleaving reasoning steps with tool invocations, an LLM can solve problems more accurately and avoid making up facts that could be computed. Similarly, Shen et al. [19] present HuggingGPT, where an LLM orchestrates various AI models (for vision, speech, etc.) to tackle complex multi-modal tasks. These approaches inspire our design: we use the LLM not as an isolated decision-maker, but as a coordinator that queries a dedicated classifier for the actual loan decision and a reasoning module for valid counterfactuals, thereby anchoring the conversation in truthful, model-verified information. In essence, our approach can be seen as applying the tool-augmented LLM paradigm specifically for XAI: the LLM "tool" here is a case-based reasoner that supplies concrete instance-based counterfactual explanations to ensure fidelity and robustness in the dialogue.

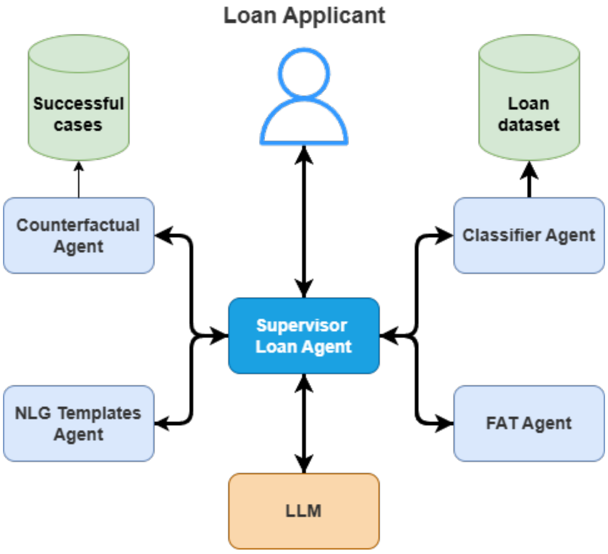


Figure 1: Agentic workflow of the proposed chatbot architecture. The supervisor agent orchestrates specialised agents to classify the application, generate counterfactuals (CF), apply the FAT constraints, and generate a natural-language explanation for the user. The supervisor agent by using the LLM can parse user queries, select the branch to follow, invoke each specialised agent, and rewrite their combined output into conversational text.

3. Methodology

3.1. Agentic System Workflow with LangGraph

For our interactive chatbot, we developed a workflow using the LangGraph framework, which allows us to explicitly define the sequence and branching of actions the agent (driven by an LLM) should perform. The architecture is illustrated in Figure 1. At a high level, the system involves a supervisor agent (which is powered by an LLM) interacting with external modules through a structured graph of operations. This design ensures that the supervisor agent consults the right agent or tool at the right time, rather than relying on the LLM to do everything in one prompt.

At the heart of this system lies a large language model (LLM) supervisor, responsible for more than just template polishing. It acts as **(i)** a *conversational router* that understands free-form user utterances and maps them onto the appropriate branch of the LangGraph, **(ii)** an *orchestrator* that conditionally calls external tools (classifier, counterfactual module, FAT checker, template NLG) and stitches their outputs together, and **(iii)** an *adaptive explainer* that revises template text to match the user's tone, education level, and follow-up questions while strictly preserving the factual content returned by the downstream agents. These three capabilities are essential: without them the workflow could neither decide which agent to invoke next nor maintain a coherent, context-aware dialogue.

The interaction begins with the user either providing their loan application details or asking a question about their application status. The supervisor agent orchestrates the following steps:

1. **Classification:** The agent first calls a classifier Agent, passing in the user's application features (e.g., income, loan amount, credit score, etc.). This agent has a classifier as a tool (a neural network with 4 linear layers) that has been trained on German Credit dataset contains 32,581 instances with 11 features (9 mutable) to predict approval or rejection of the loan status, [2]. It returns a prediction (approved/rejected).
2. **Counterfactual Generation:** If the loan is predicted to be rejected (or if the user asks "How can I get approved?" or "How can I improve my loan application?"), then the supervisor agent triggers the counterfactual explanation agent. This agent uses an instance-based approach to identify one or more plausible modifications to the user's input that would result in an approval

---

[2] https://archive.ics.uci.edu/datasets

(see algorithm 1). In our case-based approach, the module searches a database of past successful applications for a case similar to the user's to find the nearest decision boundary crossing [13]. The output is a set of candidate counterfactual changes (e.g., *increase income by $5,000; reduce loan amount by $10,000*).

3. **Actionability Check and Constraints Update:** The agent then passes the proposed counterfactual changes through the Feature Actionability Taxonomy Agent (FAT Agent). Each suggested feature change is assessed for feasibility: this agent categorises the feature as mutable or immutable (and if immutable, whether it's sensitive). Sensitive features are protected attributes such as *age*, *gender*, or *ethnicity* which must not influence credit-worthiness in order to have equal opportunity for everyone. Based on this, the module may filter out or annotate certain suggestions. For example, if a counterfactual generator naively suggested *"be 5 years older"*, the FAT Agent would label **age** as immutable (and likely sensitive) and prevent this suggestion from being presented as an action for the user. For changes that are actionable but indirect (e.g., *increase credit score*, which typically requires other actions), FAT can mark them to be phrased differently. Another important task of FAT agent is updating user constraints enabling more personalised suggestion. This aligns with preference-based case-based reasoning which explicitly models user constraints during retrieval [20].
4. **Explanation Synthesis (NLG):** Finally, the supervisor agent invokes the NLG template agent. This agent takes the refined set of counterfactual recommendations and constructs a conversational explanation. It uses predefined sentence templates that incorporate the user's data, the model's decision, and the suggested changes. The template choice and phrasing are informed by the FAT categories for each feature to ensure the explanation is polite, understandable, and appropriately hedged (especially for sensitive factors). The LLM may also be used at this stage in a fill-in-the-blank manner to smooth out the text or adjust it to the user's tone (e.g., formal vs. informal), without altering the factual content.
5. **User Interaction:** The compiled explanation is presented to the user by supervisor agent. The conversation can continue: the user might ask follow-up questions (e.g., "Why is my credit score considered low?" or "I can't change X, what else can I do?"). The supervisor agent handles these by invoking different branches of the agentic workflow as needed. For instance, if the user asks *"I cannot decrease my loan amount by $5,000"*, then FAT agent will update Feature Actionability Taxonomy for that feature in the user's profile and loop back to the counterfactual generation agent to provide another solution tailored to user's specific situation.

Algorithm 1 Iterative Adaptation for Counterfactual Generation (adapted from [21])

Input: Query instance x, nearest unlike neighbour n, constraint set C

Output: Counterfactual instance x' or **FAIL** if none found

```
$x' \leftarrow x$
foreach feature $f$ in priority order do
    if $f \notin C$ and $x'_f \neq n_f$ then
        $x'_f \leftarrow n_f$
        if Classifier($x'$) $= approved$ then
            return $x'$
return FAIL
```

This agentic loop continues until the user's queries are resolved. By explicitly encoding the workflow graph, we ensure that at each turn the supervisor agent knows which agent outputs to incorporate, maintaining consistency and factuality across the dialogue. One benefit of this modularisation is that the LLM never has to internally compute or assume the outcome of changes so that it always queries the actual model, thereby eliminating a potential source of error or hallucination. Similarly, knowledge about what is changeable is codified in FAT rather than left to the LLM's general knowledge (which might be incomplete or biased regarding what a user can or should change).

Our use of LangGraph differentiates from simpler pipeline approaches by allowing conditional branching and iterative flows. For example, based on whether the model outcome is positive or negative, different branches in the graph are followed; based on whether a user-proposed change is feasible or not, the agent can decide how to respond (perhaps by invoking the FAT logic again or by politely explaining limitations). The graph structure thus offers flexibility in the conversation flow beyond a fixed sequence, which is crucial for interactive dialogue. This modular architecture means improvements or updates to one component (say, a better classifier or a more nuanced FAT categorisation) can be integrated without retraining the entire system.

3.2. Communication between Agents

In our agentic framework, every arrow in the figure 1 represents a directed communication act which is an intentional message exchange that drives the system through its reasoning, acting, and interacting phases. Each specialised agent, including the supervisor, is configured with its own system prompt. These prompts guide the agents on how to interact with one another in a cost-efficient and purpose-driven manner. For instance, when the supervisor agent receives a user's request, it issues a directive to the classifier agent. The classifier, informed by its system prompt, responds with a brief result (e.g., "accepted" or "rejected") to efficiently update the supervisor's internal belief state.

This structured communication mirrors the principles of speech act theory, where the expressive act (such as a request or assertion) not only transmits information but also performs an action. Here, the supervisor's directive (a "request" act) and the classifier's corresponding response (an "inform" act) are essential steps in the internal virtuous cycle [22]. This cycle is designed to integrate reasoning (through chain-of-thought [23]), acting (by executing targeted decisions), and interacting (via feedback loops that enable self-reflection and data augmentation), all while reducing computational cost and ensuring clarity in each module's role.

Together, these communication acts are guided by predefined system prompts to ensure that the agentic LLM operates as a cohesive, efficient, and adaptive system.

3.3. Causally-Aware Counterfactual Explanation Agent

The counterfactual explanation agent addresses the question "What minimal changes would flip the decision to a positive outcome?" by integrating instance-based search with causal reasoning. In a high-level process, the agent first performs a nearest unlike neighbor (NUN) search in the training dataset. This search uses a weighted feature distance where the weights are computed from two components: (1) Individual Treatment Effect (ITE) values that obtained via DECI which is a causal discovery framework [24] on a Directed Acyclic Graph (DAG) that encapsulates the global causal relationships, and (2) feature actionability weights from a Feature Actionability Taxonomy (FAT) that categorises features based on their mutability. The resulting hybrid weights ensure that features with strong causal influence and practical potential for change are prioritised.

After selecting the nearest unlike neighbour, the agent generates a counterfactual explanation by iteratively adapting the query instance. For each prioritised feature, the agent updates its value to match that of the selected instance; importantly, any modification to a feature also triggers a recursive adaptation of its causally dependent (child) features as defined by the DAG. This parent/child adaptation ensures that the proposed changes maintain the inherent causal structure, thereby generating counterfactuals that are both causally effective and practically actionable.

Once a valid counterfactual is obtained (i.e., one that reverses the model's decision to the desired outcome) the counterfactual explanation agent communicates this result back to the supervisor agent, completing its role within the overall agentic workflow.

3.4. Feature Actionability Taxonomy (FAT) Agent Integration

The FAT agent is responsible for ensuring that counterfactual explanations are personalised and practically actionable by updating the user's feature constraints based on their feedback. Conceptually, the FAT agent analyses counterfactual suggestions and identifies if any proposed changes involve features that the user deems immutable (e.g. the *loan Amount*, normally adjustable by applicants, becomes immutable if the user specifically needs exactly \$5,000 to cover a planned expense). When such features are detected, the FAT agent revises its internal FAT profile to reflect these constraints.

Once the FAT agent updates this information, it first communicates the revised FAT details back to the supervisor agent. The supervisor agent, acting as the central coordinator, then relays the updated FAT guidelines to the counterfactual generation agent, instructing it to generate a new counterfactual explanation that respects the user's specified immutability. This dynamic exchange among the FAT agent, supervisor agent, and counterfactual generation agent ensures that the final counterfactual recommendations are both causally informed and tailored to the user's real-world capabilities. If, under these constraints, no valid counterfactual can be found, the system will explicitly inform the user: "Given the constraints, no feasible changes can overturn the decision."

3.5. Template-Based Natural Language Generation (NLG) Agent

In our agentic framework the LLM co-authors the final explanation; it fills the slots of a safe template and then dynamically re-voices the text (e.g., adjusting formality, adding clarifications requested in the previous turn), something a rigid template engine alone cannot do [7]. Template conditioning constrains the LLM's free-form generation which significantly reduces factual drift; this mirrors the findings of Upadhyay *et al.* [25], who reported higher factual accuracy when using templates for obituary generation. The process unfolds in several coordinated steps: First, the counterfactual generation agent generates a counterfactual suggestion, which it communicates to the supervisor agent. The supervisor then forwards this counterfactual to the NLG agent. The NLG agent, governed by its system prompt and provided tool, fills in predefined templates, such as an introductory sentence, actionable recommendation templates for mutable or indirectly mutable features, templates that neutrally acknowledge sensitive and non-sensitive immutable factors, and a concluding sentence to construct a clear and consistent natural language explanation.

For example, a final output might look like:

> Your loan application was declined due to several factors. To improve your chances:
>
> Take steps to increase your monthly income to at least \$5,000 (currently \$4,200).
>
> Reduce your requested loan amount closer to \$10,000 (currently \$15,000).
>
> Unfortunately, you cannot change your age (25), and younger applicants often have shorter credit histories.
>
> These changes would address the risk factors identified by the model. I hope this helps, good luck with your next application!

After generating its explanation, the NLG agent returns its output to the supervisor agent. The supervisor agent then refines the explanation as needed, enhancing its naturalness and realism, before presenting the final, user-tailored message. This chain of communication ensures that every counterfactual recommendation is accurately translated into clear, actionable, and contextually appropriate advice.

3.6. Summary of Agent Interactions

The proposed agentic workflow forms a cohesive, feedback-driven system in which each specialised agent contributes to delivering personalised and actionable explanations. Initially, the classifier agent predicts the loan decision. If the loan is rejected or if a user requests guidance, the counterfactual explanation agent identifies the nearest unlike neighbour and adapts its features to generate a counterfactual explanation for the query.

If a participant indicates that a suggested change is unworkable (e.g., stating "I cannot change X"), the FAT agent is activated. In response, the FAT agent updates its internal profile to mark the specified feature as immutable and promptly communicates these updated constraints to the supervisor agent. The supervisor agent then relays the revised FAT information to the counterfactual explanation agent, which regenerates a personalised counterfactual that adheres to the updated constraints. This revised explanation is forwarded to the template-based NLG agent, which transforms it into a clear and structured natural language message. Finally, after further refinement by the supervisor agent to ensure naturalness and realism, the final explanation is presented to the user.

This iterative process, whereby the system continuously refines its counterfactual based on user feedback via the FAT agent, is central to ensuring that the provided explanations are both causally consistent and tailored to the user's real-world constraints. In the next section, we describe the experimental setup and planned user study that will evaluate the impact of our integrated agentic framework on explanation quality and user trust.

4. Experimental setup

In order to assess the effectiveness of the proposed system, we outline experiments focusing on two aspects: (1) the quality and feasibility of the explanations, and (2) the impact on user understanding and trust.

We compare three versions of the chatbot:

- **Full Agentic System:** the complete system as described, using the classifier, counterfactual reasoning, FAT, and template NLG (the CBR-LLM agent).
- **No-FAT Variant:** an ablation where the FAT module is disabled. The agent still generates counterfactuals, but it cannot update user's specific constraints to provide a better counterfactual tailored to the user.
- **LLM-Only Baseline:** a baseline using a state-of-the-art LLM (e.g., GPT-4) with only classifier tool. This baseline is prompted to perform the entire task on its own. For example, we provide it with the user's input and ask it to give an explanation or advice. The prompt can be engineered with few-shot examples to encourage counterfactual-style outputs. However, the LLM will not have access to the FAT agent or counterfactual agent to generate responses and it relies only on its general knowledge and reasoning.

For the LLM we use GPT-4o-mini for all three versions.

5. User Study Evaluation

In order to assess the effectiveness of our proposed system, we will conduct a user study focusing on two main aspects: the quality and feasibility of the counterfactual explanations, and the overall impact on user understanding and trust. In this future study, participants will engage in an interactive chat environment implemented using Streamlit, [3], where the supervisor agent coordinates among internal agents, namely, the classifier agent, the counterfactual generation agent, the FAT agent, and the natural language generation (NLG) agent.

The central focus of our evaluation will be the performance of the automated feedback loop. Specifically, the FAT agent will update the user's constraints based on their feedback regarding unworkable suggestions, and these updated constraints will prompt the counterfactual generation agent to generate refined and more actionable counterfactual explanations. This iterative process, which continually updates the explanation based on user input, is expected to significantly enhance the personalisation and practical utility of the system's advice.

Following this interactive phase, participants will complete a detailed survey measuring dimensions such as overall satisfaction, clarity, helpfulness, fairness, and trustworthiness of the provided explanations. This comprehensive evaluation approach, informed by recent best practices in user-centred XAI research [26, 27, 28], will allow us to rigorously assess the impact of our integrated agentic framework on user experience.

---

[3] https://streamlit.io: an open-source Python framework for rapidly building data-driven web applications with minimal boilerplate.

5.1. Participant Recruitment and Screening

Participants will be recruited from local university networks and professional online communities. We will use a screening process to select individuals who can realistically assume the role of a loan applicant and have an elementary understanding of financial decision-making. The required sample size will be determined using G*Power analysis, targeting a power of 0.80, an alpha level of 0.05, and an anticipated medium effect size (Cohen's d = 0.5), which suggests a minimum of 34 participants. To account for potential dropouts and to enhance the robustness of our findings, approximately 40 participants will be recruited. All participants will be provided with an information sheet and will give informed consent prior to their involvement.

5.2. Study Design and Procedure

Participants are presented with the following scenario:

- **Scenario:** You are John Doe, a 36-year-old applicant whose loan request for purchasing a new car has been rejected. Your task is to interact with the intelligent assistant and explore ways to improve your loan application.

In this study, participants interact with each of the three system variants described earlier for up to five turns (one user utterance plus one system reply constitutes a turn). For each of the three chatbot variants, we collect the complete five-turn dialogue, five per-variant Likert-scale ratings, and optional free-form feedback, and analyse these data jointly to assess how specific dialogue features (e.g., invoked modules) influence user evaluations. We adopt a within-subject Latin-square counter-balancing: the order of the three variants ($3! = 6$ possible sequences) is randomly assigned, and the same decision scenario is reused across variants to keep task difficulty constant. The examples of these chat interactions and UI screenshots will be provided in Appendix I.

During these interactions, the supervisor agent orchestrates responses to both general queries and specific requests for guidance on improving a loan application. For general questions, the assistant provides prompt, context-aware answers; however, when a participant seeks concrete solutions or strategies to improve their rejected loan application, the counterfactual generation agent is activated. This agent first evaluates the current scenario (i.e., a loan rejection) and generates a counterfactual suggestion that details minimal modifications to the input features.

This generated counterfactual is then communicated to the supervisor agent, which passes it on to the template-based NLG agent. The NLG agent transforms the counterfactual suggestion into a clear and structured explanation using predefined templates. If the resulting explanation includes any suggestions that the participant finds immutable, for example, due to personal constraints, the FAT agent is triggered. The FAT agent updates the user's feature constraints (e.g., marking features as non-negotiable) and communicates these updated FAT details back to the supervisor agent. The supervisor agent subsequently instructs the counterfactual generation agent to generate a revised counterfactual explanation that adheres to the updated constraints.

After further refinement by the supervisor to enhance naturalness and realism, the final, personalised explanation is returned to the participant. This coordinated multi-agent interaction ensures that the counterfactual explanations are not only feasible and actionable but also tailored to the user's real-world circumstances. To safeguard data quality, we embed an instructed-response attention check ("Select *Disagree*"), a recall question on the chatbot's last suggestion, and a 5 minutes interaction-time requirement per variant; any participant failing two of these criteria is removed *a priori.*

5.3. Evaluation Survey and Measures

After the interaction phase, they will complete a detailed survey designed to capture their perceptions of the system's explanations.

The survey will include Likert-scale items measuring:

- **Overall Satisfaction:** How satisfied the participant is with the overall assistance provided by the chatbot.
- **Clarity:** The degree to which the counterfactual explanations are clear and understandable.
- **Helpfulness:** The extent to which the suggestions aid the participant in understanding what changes could improve their loan application.
- **Fairness:** The perceived fairness and ethical appropriateness of the recommendations.
- **Trustworthiness:** The degree to which the participant trusts the assistant's advice [29].

For example, an item measuring clarity might ask, "How clear and understandable were the counterfactual explanations provided by the assistant?" with response options ranging from 1 (not clear at all) to 5 (extremely clear). Similar items will be framed for the other dimensions.

In addition to these Likert-scale measures, the survey will include open-ended questions where participants outline the specific steps they would take based on the suggestions they received. This qualitative feedback will offer further insight into the actionability and practical utility of the explanations.

Because each participant uses all three variants, we will compare their mean Likert ratings with paired t-tests for the three variant pairs (full agentic vs. no-FAT, full agentic vs. LLM-only, and no-FAT vs. LLM-only). We will first check the normality of the rating differences with the Shapiro-Wilk test; if this assumption is violated, we will use the Wilcoxon signed-rank test instead. The GPower sample-size estimate (medium effect $d = 0.5$, $\alpha = 0.05$, power = 0.80) was calculated for this paired-test design.

6. Conclusions and Future Work

We introduced an agentic framework for loan-application guidance that combines case-based reasoning with large language models. By orchestrating specialised agents including a classifier, a causally aware counterfactual generator, a Feature Actionability Taxonomy (FAT) checker, and a template-based NLG module. The system is designed to generate explanations that respect the user's real-world constraints and causal dependencies.

Although empirical validation is still to come, the modular architecture is intended to reduce common LLM issues such as hallucination and to support a feedback loop that should improve clarity and trust. These hypotheses will be tested in a forthcoming user study.

We posit that this agentic-CBR approach can provide personalised, actionable recourse in high-stakes finance scenarios and, once validated, could generalise to other domains where trustworthy decision support is required.

Declaration on Generative AI

During preparation of this work, the authors used ChatGPT for the purpose of: grammar and spelling check, paraphrase and reword. After using this tool, the authors reviewed and edited the content and take full responsibility for the publication's content.

References

[1] E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the dangers of stochastic parrots: Can language models be too big?, in: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 610–623. URL: https://doi.org/10.1145/3442188.3445922. doi:10.1145/3442188.3445922.

[2] P. Salimi, Addressing trust and mutability issues in xai utilising case based reasoning., ICCBR Doctoral Consortium 1613 (2022) 0073.

[3] S. Wachter, B. Mittelstadt, C. Russell, Counterfactual explanations without opening the black box: Automated decisions and the gdpr, Harv. JL & Tech. 31 (2017) 841.

[4] A.-H. Karimi, B. Schölkopf, I. Valera, Algorithmic recourse: from counterfactual explanations to interventions, in: Proceedings of the 2021 ACM conference on fairness, accountability, and transparency, 2021, pp. 353–362.

[5] B. Ustun, A. Spangher, Y. Liu, Actionable recourse in linear classification, in: Proceedings of the conference on fairness, accountability, and transparency, 2019, pp. 10–19.

[6] R. Poyiadzi, K. Sokol, R. Santos-Rodriguez, T. De Bie, P. Flach, Face: feasible and actionable counterfactual explanations, in: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, 2020, pp. 344–350.

[7] P. Salimi, N. Wiratunga, D. Corsar, A. Wijekoon, Towards feasible counterfactual explanations: A taxonomy guided template-based nlg method, in: ECAI 2023, IOS Press, 2023, pp. 2057–2064.

[8] D. Mahajan, C. Tan, A. Sharma, Preserving causal constraints in counterfactual explanations for machine learning classifiers, arXiv preprint arXiv:1912.03277 (2019).

[9] A. Wijekoon, D. Corsar, N. Wiratunga, K. Martin, P. Salimi, Tell me more: Intent fulfilment framework for enhancing user experiences in conversational xai, arXiv preprint arXiv:2405.10446 (2024).

[10] A. Wijekoon, N. Wiratunga, K. Martin, D. Corsar, I. Nkisi-Orji, C. Palihawadana, D. Bridge, P. Pradeep, B. D. Agudo, M. Caro-Martínez, Cbr driven interactive explainable ai, in: International conference on case-based reasoning, Springer, 2023, pp. 169–184.

[11] S. Verma, V. Boonsanong, M. Hoang, K. Hines, J. Dickerson, C. Shah, Counterfactual explanations and algorithmic recourses for machine learning: A review, ACM Computing Surveys 56 (2024) 1–42.

[12] R. K. Mothilal, A. Sharma, C. Tan, Explaining machine learning classifiers through diverse counterfactual explanations, in: Proc. Conf. on Fairness, Accountability, and Transparency, 2020, pp. 607–617.

[13] N. Wiratunga, A. Wijekoon, I. Nkisi-Orji, K. Martin, C. Palihawadana, D. Corsar, Discern: Discovering counterfactual explanations using relevance features from neighbourhoods, in: 2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, 2021, pp. 1466–1473.

[14] D. Brughmans, P. Leyman, D. Martens, Nice: an algorithm for nearest instance counterfactual explanations, Data mining and knowledge discovery (2023) 1–39.

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

[16] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al., Retrieval-augmented generation for knowledge-intensive nlp tasks, Advances in neural information processing systems 33 (2020) 9459–9474.

[17] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, Y. Cao, React: Synergizing reasoning and acting in language models, in: International Conference on Learning Representations (ICLR), 2023.

[18] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, T. Scialom, Toolformer: Language models can teach themselves to use tools, Advances in Neural Information Processing Systems 36 (2023) 68539–68551.

[19] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, Y. Zhuang, Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face, Advances in Neural Information Processing Systems 36 (2023) 38154–38180.

[20] E. Hüllermeier, P. Schlegel, Preference-based cbr: First steps toward a methodological framework, in: International Conference on Case-Based Reasoning, Springer, 2011, pp. 77–91.

[21] B. Smyth, M. T. Keane, A few good counterfactuals: generating interpretable, plausible and diverse counterfactual explanations, in: International Conference on Case-Based Reasoning, Springer, 2022, pp. 18–32.

[22] Z. Wang, S. Moriyama, W.-Y. Wang, B. Gangopadhyay, S. Takamatsu, Talk structurally, act hierarchically: A collaborative framework for llm multi-agent systems, arXiv preprint arXiv:2502.11098 (2025).

[23] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., Chain-of-thought

prompting elicits reasoning in large language models, Advances in neural information processing systems 35 (2022) 24824–24837.
[24] T. Geffner, J. Antoran, A. Foster, W. Gong, C. Ma, E. Kiciman, A. Sharma, A. Lamb, M. Kukla, A. Hilmkil, et al., Deep end-to-end causal inference, in: NeurIPS 2022 Workshop on Causality for Real-world Impact, 2022.
[25] A. Upadhyay, S. Massie, S. Clogher, Case-based approach to automated natural language generation for obituaries, in: International Conference on Case-Based Reasoning, Springer, 2020, pp. 279–294.
[26] Y. Rong, T. Leemann, T.-T. Nguyen, L. Fiedler, P. Qian, V. Unhelkar, T. Seidel, G. Kasneci, E. Kasneci, Towards human-centered explainable ai: A survey of user studies for model explanations, IEEE transactions on pattern analysis and machine intelligence 46 (2023) 2104–2122.
[27] J. Kim, H. Maathuis, D. Sent, Human-centered evaluation of explainable ai applications: a systematic review, Frontiers in Artificial Intelligence 7 (2024) 1456486.
[28] L. Gates, D. Leake, K. Wilkerson, Cases are king: a user study of case presentation to explain cbr decisions, in: International Conference on Case-Based Reasoning, Springer, 2023, pp. 153–168.
[29] R. R. Hoffman, A taxonomy of emergent trusting in the human–machine relationship, Cognitive systems engineering (2017) 137–164.

A. Supplementary Material

The code, data and the user study materials can be find here:
https://github.com/pedramsalimi/CBRAgent

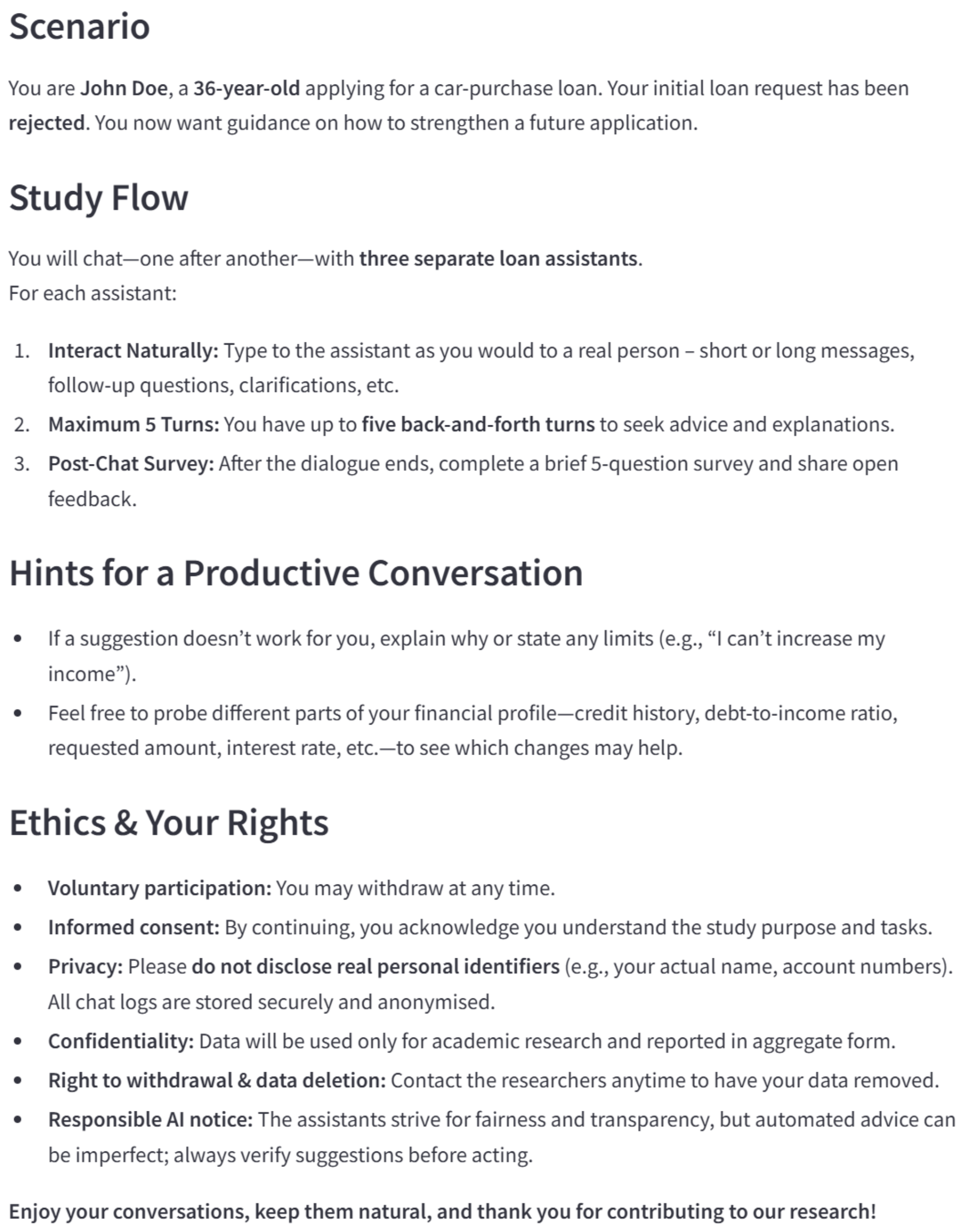
A.1. Study Instructions and Interactions

Figures 2 and 3 present two screenshots of the instructions that will be provided to participants at the beginning of the user study. These images depict the comprehensive guidelines for the study scenario, including task descriptions, interaction guidelines, and post-interaction procedures.

A.2. Example of System Interaction

Figures 4 and 5 illustrate an example screenshot of the interactive chat interface. This image shows a typical conversation between the user and the assistant, demonstrating how the system handles queries, generates counterfactual explanations, and communicates results back to the user.

Figure 6 provides a screenshot of the survey interface that participants will complete after their interactions with the assistant. The survey employs Likert-scale questions to assess the clarity, helpfulness, fairness, and trustworthiness of the explanations, as well as open-ended questions to capture qualitative feedback on the actionability of the suggestions.



Scenario

You are **John Doe**, a **36-year-old** applying for a car-purchase loan. Your initial loan request has been **rejected**. You now want guidance on how to strengthen a future application.

Study Flow

You will chat—one after another—with **three separate loan assistants**.
For each assistant:

1. **Interact Naturally:** Type to the assistant as you would to a real person – short or long messages, follow-up questions, clarifications, etc.
2. **Maximum 5 Turns:** You have up to **five back-and-forth turns** to seek advice and explanations.
3. **Post-Chat Survey:** After the dialogue ends, complete a brief 5-question survey and share open feedback.

Hints for a Productive Conversation

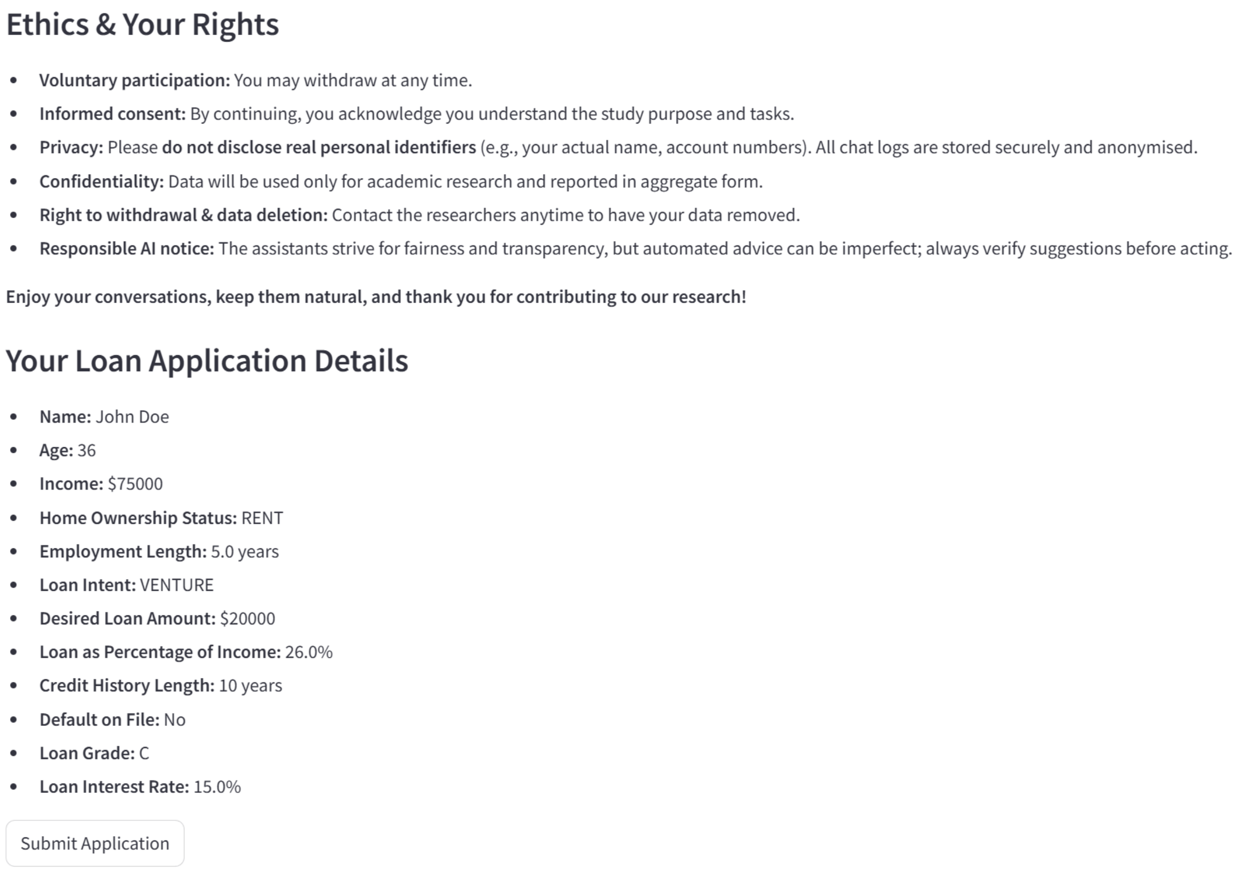
- If a suggestion doesn't work for you, explain why or state any limits (e.g., "I can't increase my income").
- Feel free to probe different parts of your financial profile—credit history, debt-to-income ratio, requested amount, interest rate, etc.—to see which changes may help.

Ethics & Your Rights

- **Voluntary participation:** You may withdraw at any time.
- **Informed consent:** By continuing, you acknowledge you understand the study purpose and tasks.
- **Privacy:** Please **do not disclose real personal identifiers** (e.g., your actual name, account numbers). All chat logs are stored securely and anonymised.
- **Confidentiality:** Data will be used only for academic research and reported in aggregate form.
- **Right to withdrawal & data deletion:** Contact the researchers anytime to have your data removed.
- **Responsible AI notice:** The assistants strive for fairness and transparency, but automated advice can be imperfect; always verify suggestions before acting.

Enjoy your conversations, keep them natural, and thank you for contributing to our research!

Figure 2: Study instructions (Part I) for participants.



Ethics & Your Rights

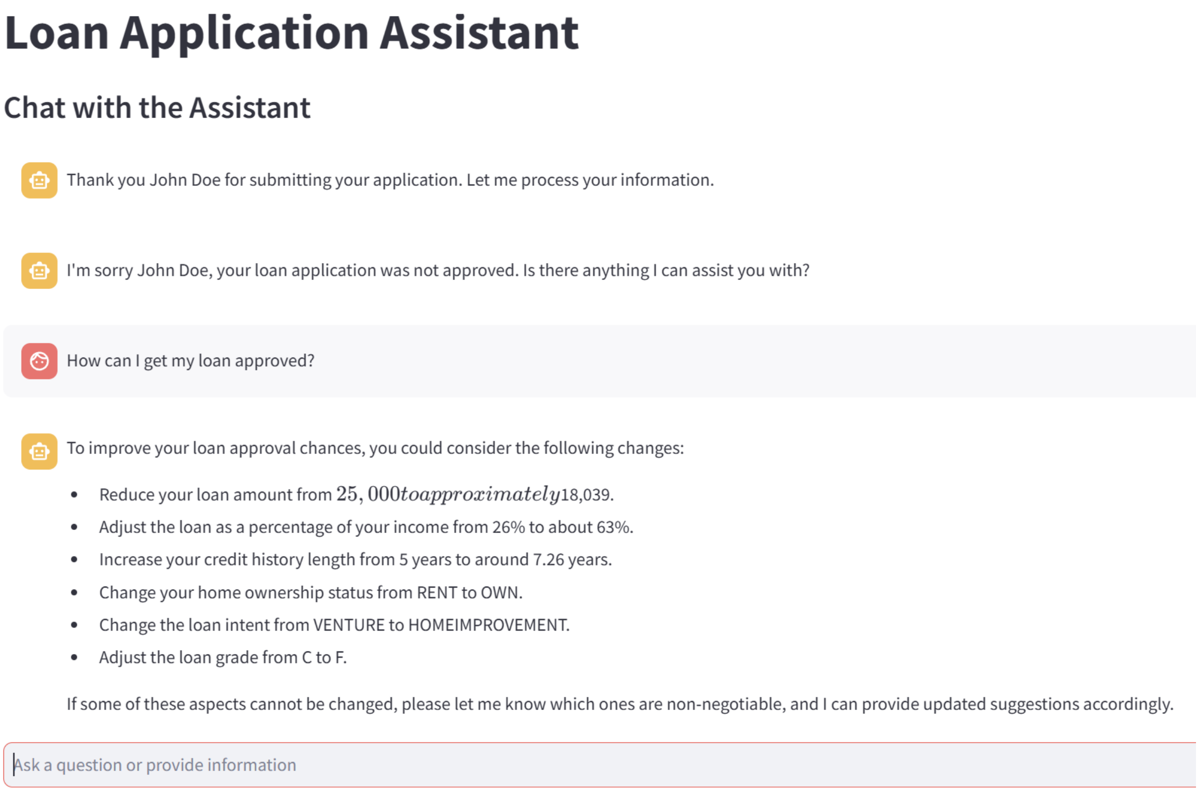
- **Voluntary participation:** You may withdraw at any time.
- **Informed consent:** By continuing, you acknowledge you understand the study purpose and tasks.
- **Privacy:** Please **do not disclose real personal identifiers** (e.g., your actual name, account numbers). All chat logs are stored securely and anonymised.
- **Confidentiality:** Data will be used only for academic research and reported in aggregate form.
- **Right to withdrawal & data deletion:** Contact the researchers anytime to have your data removed.
- **Responsible AI notice:** The assistants strive for fairness and transparency, but automated advice can be imperfect; always verify suggestions before acting.

Enjoy your conversations, keep them natural, and thank you for contributing to our research!

Your Loan Application Details

- **Name:** John Doe
- **Age:** 36
- **Income:** $75000
- **Home Ownership Status:** RENT
- **Employment Length:** 5.0 years
- **Loan Intent:** VENTURE
- **Desired Loan Amount:** $20000
- **Loan as Percentage of Income:** 26.0%
- **Credit History Length:** 10 years
- **Default on File:** No
- **Loan Grade:** C
- **Loan Interest Rate:** 15.0%

Figure 3: Study instructions (Part II) for participants.



Loan Application Assistant

Chat with the Assistant

Thank you John Doe for submitting your application. Let me process your information.

I'm sorry John Doe, your loan application was not approved. Is there anything I can assist you with?

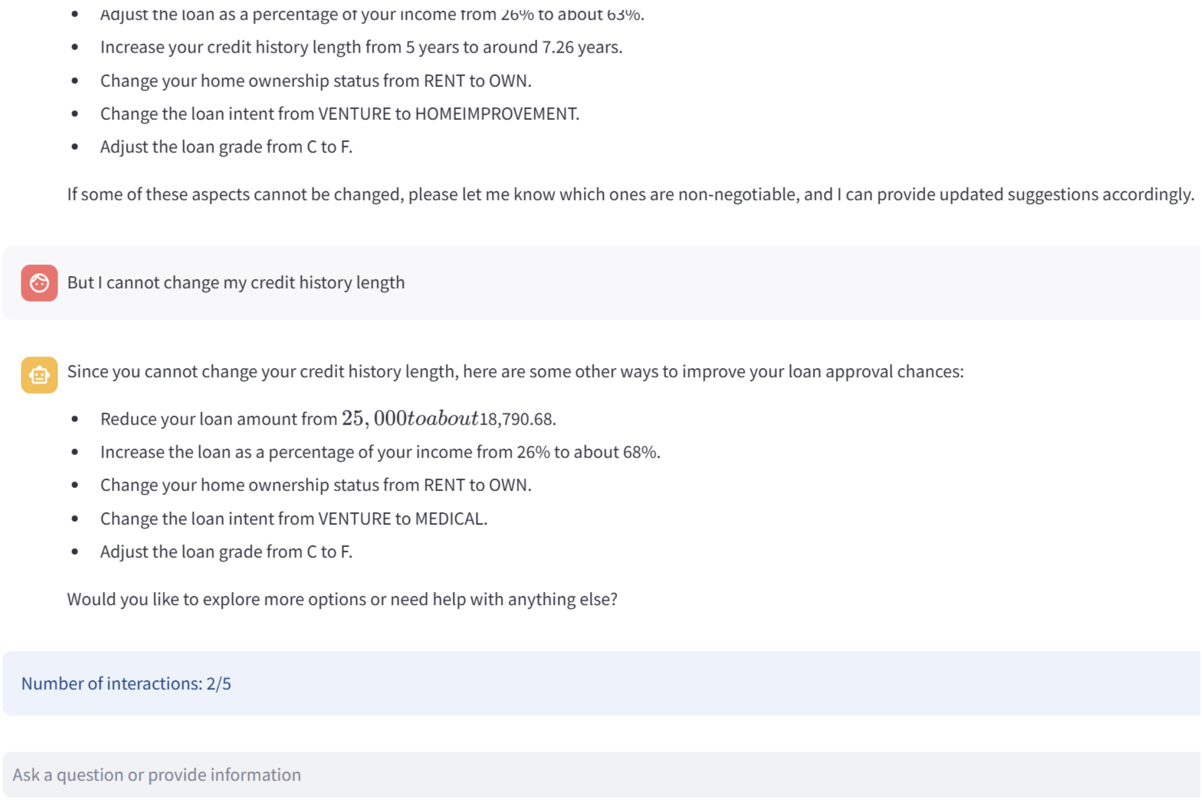
How can I get my loan approved?

To improve your loan approval chances, you could consider the following changes:

- Reduce your loan amount from $25,000toapproximately$18,039.
- Adjust the loan as a percentage of your income from 26% to about 63%.
- Increase your credit history length from 5 years to around 7.26 years.
- Change your home ownership status from RENT to OWN.
- Change the loan intent from VENTURE to HOMEIMPROVEMENT.
- Adjust the loan grade from C to F.

If some of these aspects cannot be changed, please let me know which ones are non-negotiable, and I can provide updated suggestions accordingly.

Figure 4: Example screenshot of system interaction.



- Adjust the loan as a percentage of your income from 26% to about 63%.
- Increase your credit history length from 5 years to around 7.26 years.
- Change your home ownership status from RENT to OWN.
- Change the loan intent from VENTURE to HOMEIMPROVEMENT.
- Adjust the loan grade from C to F.

If some of these aspects cannot be changed, please let me know which ones are non-negotiable, and I can provide updated suggestions accordingly.

But I cannot change my credit history length

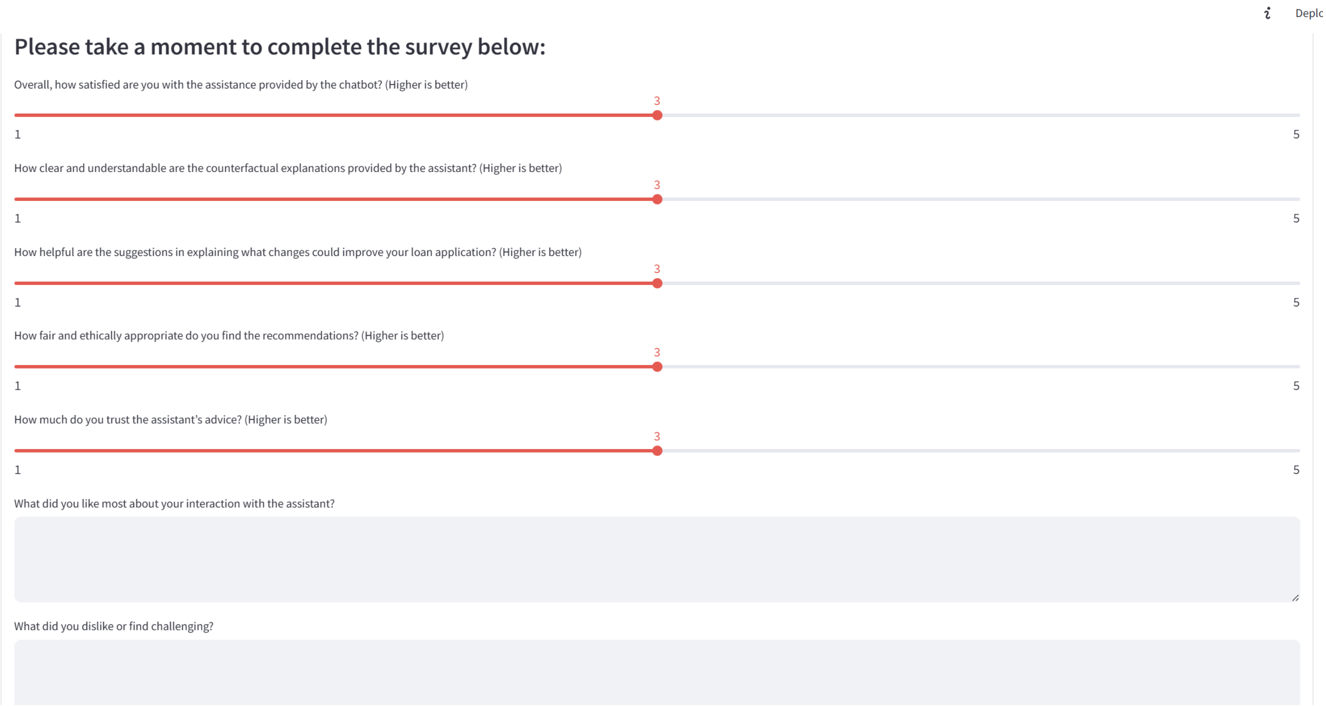
Since you cannot change your credit history length, here are some other ways to improve your loan approval chances:

- Reduce your loan amount from $25, 000 to about$18,790.68.
- Increase the loan as a percentage of your income from 26% to about 68%.
- Change your home ownership status from RENT to OWN.
- Change the loan intent from VENTURE to MEDICAL.
- Adjust the loan grade from C to F.

Would you like to explore more options or need help with anything else?

Number of interactions: 2/5

Figure 5: Example screenshot of system interaction.



Please take a moment to complete the survey below:

Overall, how satisfied are you with the assistance provided by the chatbot? (Higher is better)

3

1 5

How clear and understandable are the counterfactual explanations provided by the assistant? (Higher is better)

3

1 5

How helpful are the suggestions in explaining what changes could improve your loan application? (Higher is better)

3

1 5

How fair and ethically appropriate do you find the recommendations? (Higher is better)

3

1 5

How much do you trust the assistant's advice? (Higher is better)

3

1 5

What did you like most about your interaction with the assistant?

What did you dislike or find challenging?

Figure 6: Screenshot of the evaluation survey interface with Likert-scale items and open-ended questions. Higher scores are better