Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

AI-Based UI Personalization in Adaptive Mobile Applications

Dr. Akankasha Kathuria
Assistant Professor
Amity University Rajasthan, Jaipur
Pin-303002
ORCID I'd: https://orcid.org/0000-0002-7288-2490

https://wjcrt.org/ || **Vol. 2 No. 3 (2026): September Issue**

Date of Submission: 10-08-2026	**Date of Acceptance:** 23-08-2026	**Date of Publication:** 11-09-2026

Abstract — Artificial intelligence is increasingly being embedded within mobile applications to personalize navigation, content presentation, interaction sequences, and interface components according to individual user behavior. However, conventional personalization mechanisms frequently prioritize immediate engagement or task efficiency while paying insufficient attention to interface stability, contextual variability, and user control over automated adaptation. This study addresses this gap by conceptualizing mobile UI personalization as a constrained adaptive-learning problem in which personalization benefits must be balanced against disruptive interface changes. A context-aware personalization framework is proposed that learns from interaction history, temporal context, navigation behavior, device conditions, and explicit user preferences while maintaining a controlled degree of interface consistency. The framework combines behavioral representation learning with a contextual decision mechanism designed to determine when, where, and to what extent interface adaptation should occur. Unlike static personalization or unrestricted reinforcement-learning approaches, the proposed research incorporates an interface-stability constraint and user override mechanism into the adaptation process. The experimental design evaluates personalization using usability, task-efficiency, adaptation acceptance, interface stability, and computational-performance indicators rather than relying exclusively on engagement-based measures.

Keywords— Adaptive User Interface, Artificial Intelligence, Mobile Personalization, Context-Aware Computing, User Behavior Modeling, Human-Centered AI

Introduction

Mobile applications have evolved from relatively static software interfaces into continuously updated digital environments capable of responding to individual users. Contemporary applications collect diverse interaction signals, including navigation sequences, search histories, dwell times, touch patterns, feature usage, session frequency, device state, temporal behavior, and explicitly specified preferences. Artificial intelligence provides mechanisms through which these heterogeneous signals can be transformed into user representations and subsequently used to determine which content, functionality, or interface configuration is most appropriate for a particular interaction context.

Personalization is therefore no longer limited to recommending products, media, or information. Increasingly, the structure of the interface itself can become adaptive. Menu ordering may change according to frequently used functions; shortcuts can be surfaced based on predicted actions; visual density may be modified according to usage patterns; onboarding guidance can be progressively removed as competence increases; and navigation components may be reorganized according to contextual relevance. Adaptive User Interfaces (AUIs) consequently represent an intersection of artificial intelligence, user modeling, and Human–Computer Interaction (HCI).

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

The underlying motivation for adaptive interfaces is straightforward. A single static interface must accommodate users with considerably different goals, abilities, preferences, experience levels, and situational constraints. An interface designed for an inexperienced user may contain excessive guidance for an expert, whereas a compact expert-oriented interface can impose unnecessary cognitive demands on a novice. Miraz, Ali, and Excell's review of adaptive interfaces emphasizes that effective AUI development lies at the intersection of AI, user modeling, and HCI and that personalization frequently has to operate using limited interaction data from individual users.

Adaptive mobile applications intensify this problem because mobile interaction is highly contextual. The same individual may interact differently depending upon time, location category, network quality, battery availability, task urgency, physical environment, or application experience. A personalization mechanism based solely on historical averages may therefore produce an interface that represents a user's past behavior but not necessarily the user's present need.

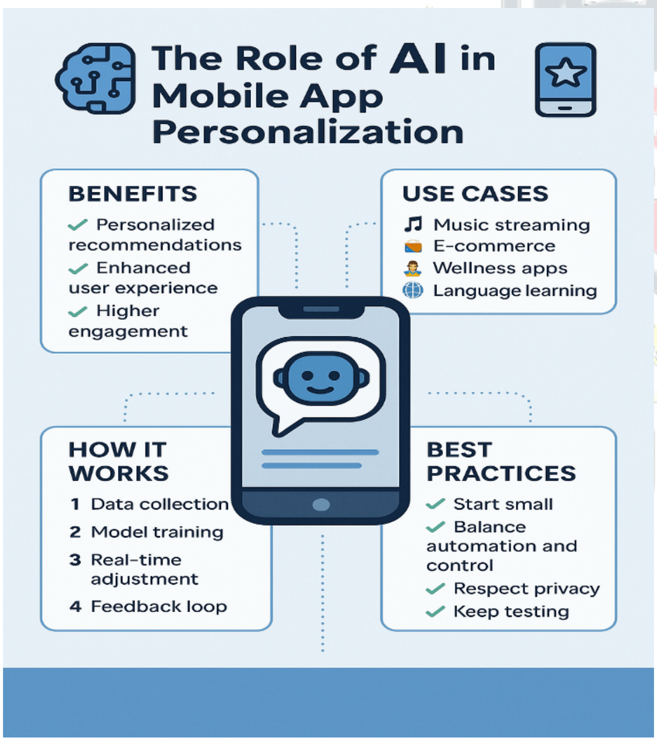

Fig. 1: The Role of AI in Mobile App Personalization

Source: *https://blog.flipr.ai/the-role-of-ai-in-mobile-app-personalization/*

Recent research has increasingly investigated machine learning and reinforcement learning for intelligent interface adaptation. Vidmanov and Alfimtsev, for example, proposed a mobile interface adaptation approach based on a usability reward model and multi-agent reinforcement learning, demonstrating the potential of reinforcement-learning techniques for automatically selecting UI configurations. Their work also identifies reward formulation as a central difficulty because an adaptive system must translate multidimensional usability characteristics into computational signals that an optimization algorithm can learn.

The effectiveness of AI-driven adaptation, however, cannot be measured exclusively through increased click-through rates, interaction frequency, or reduced task duration. Excessive adaptation may itself become a usability problem. When controls repeatedly change position or information hierarchy is frequently rearranged, users can lose the spatial and procedural knowledge developed during earlier sessions. An algorithmically optimal configuration at one moment may therefore produce a longer-term cost by decreasing predictability.

This tension creates an important distinction between **personalization accuracy** and **interaction appropriateness**. A model may correctly identify the feature a user is most likely to select while still presenting that feature in a way that disrupts established interaction habits. Similarly, an interface may become statistically optimized for common behaviors but reduce the user's perceived control over the application.

User autonomy becomes particularly relevant as mobile applications increasingly employ opaque behavioral inference. An adaptive system can potentially infer preferences without asking users whether a corresponding UI modification is desirable. Recent work on machine-learning-based accessible interfaces has consequently emphasized the importance of balancing autonomous personalization with explicit user control and user-approved adaptation.

1.1 Research Gap

Existing studies demonstrate considerable progress in adaptive menus, recommendation-driven interfaces, contextual personalization, usability-aware optimization, and reinforcement-learning-based interface adaptation. Nevertheless, three limitations remain insufficiently integrated within a unified mobile personalization framework.

First, many personalization systems primarily optimize positive interaction outcomes such as engagement, efficiency, recommendation relevance, or feature accessibility. They rarely penalize excessive modifications to an interface that users have already learned.

Second, contextual information and longer-term user behavior are often treated as equivalent predictive signals, although they operate at different temporal scales. Persistent preferences represent comparatively stable characteristics,

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

whereas situational signals may only be relevant during a particular session.

Third, automated adaptation frequently concentrates on what the system should change while giving less attention to whether the system should change anything at all. This distinction is important because maintaining the current interface may occasionally be preferable to an adaptation that provides only marginal predicted benefit.

The central research gap addressed in this manuscript is therefore the **absence of a unified AI-based mobile UI personalization strategy that explicitly optimizes contextual relevance while simultaneously constraining interface instability and preserving user control**.

Rather than asking only, *"Which interface configuration maximizes predicted utility?"*, the study introduces a second decision problem: *"Is the expected benefit of adaptation sufficient to justify changing the interface?"*

This formulation represents personalization as a constrained decision process rather than continuous UI optimization.

1.2 Research Objective

The principal objective of this research is to develop and evaluate a context-aware AI framework capable of dynamically personalizing mobile user interfaces while minimizing unnecessary interface changes.

More specifically, the study investigates whether the integration of behavioral representations, contextual features, adaptation-cost penalties, and user-feedback signals can generate mobile interfaces that achieve higher task efficiency and perceived personalization without producing excessive layout volatility.

The proposed approach will distinguish between persistent user preferences and short-lived contextual behavior. The personalization engine will consequently evaluate both the predicted utility of an interface configuration and the transition cost associated with moving from the current interface state to the proposed configuration.

The research is guided by the proposition that an effective adaptive mobile interface should not merely become more personalized over time; it should become **selectively adaptive**, modifying itself only when the expected improvement exceeds the usability cost created by the modification.

Literature Review

2.1 Evolution of Adaptive User Interfaces

Adaptive user interfaces have long been investigated as a mechanism for accommodating differences among users. Their fundamental objective is to alter aspects of an interface according to information derived from users, tasks, devices, or contexts.

The systematic review by Miraz et al. positions adaptive interfaces within the wider concept of interface plasticity, where the interface preserves usability while responding to changes in users or environmental conditions. Their analysis also illustrates the interdisciplinary nature of the field, particularly the combination of artificial intelligence, user modeling, and HCI.

Earlier adaptive systems frequently relied on predetermined rules. A user could be classified as novice or expert, after which a corresponding interface configuration would be activated. Although computationally simple and interpretable, such systems could represent only a limited number of interaction patterns.

Machine-learning approaches enable considerably more granular personalization because adaptation policies can be learned from behavioral observations rather than manually encoded. This transition allows interface adaptation to move from demographic or rule-based segmentation toward individualized behavioral modeling.

2.2 Adaptive Navigation and Menu Personalization

Navigation represents one of the most frequently studied components of adaptive interfaces. Menu adaptation attempts to reduce interaction effort by reorganizing, highlighting, predicting, or prioritizing commands that are more likely to be required.

Shrestha et al.'s review of adaptive menus describes multiple adaptation styles and policies intended to decrease selection time and improve interaction efficiency. At the same time, adaptive menus illustrate one of the central difficulties of dynamic interface design: modifications intended to accelerate selection can conflict with users' familiarity with stable menu structures.

In mobile environments, this problem becomes more pronounced because limited screen space increases competition among interface elements. Only a small number of features can occupy high-visibility regions such as bottom navigation bars, home screens, floating controls, or prominent shortcuts. Intelligent personalization can therefore provide substantial benefits by allocating scarce visual space according to predicted relevance.

However, continuous rearrangement of frequently used functions may create an interaction paradox. The more actively the application attempts to optimize navigation, the more frequently users may have to rediscover the location of those functions. Consequently, adaptation frequency itself should be treated as an optimization variable.

2.3 Machine Learning for Dynamic Interface Adaptation

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

Modern adaptive interfaces can employ supervised learning, clustering, contextual bandits, reinforcement learning, deep sequence models, or hybrid architectures. Each technique addresses a different aspect of personalization.

Supervised models can estimate the probability that a user will select a feature given a behavioral and contextual feature vector. Clustering can identify recurring behavioral profiles without predefined labels. Sequential models can represent the order in which actions occur, while reinforcement learning can select actions according to long-term reward rather than isolated predictions.

Recent research has investigated increasingly autonomous approaches. Vidmanov and Alfimtsev developed an interface adaptation mechanism using multi-agent reinforcement learning and a usability-oriented reward model. Their work highlights the potential of reinforcement learning to optimize UI configurations dynamically while simultaneously illustrating the difficulty of constructing appropriate rewards for subjective usability characteristics.

Reinforcement learning is particularly suitable for interface adaptation because personalization can naturally be represented using states, actions, and rewards. A state describes the current user and interaction context; an action represents an interface modification; and the reward indicates whether that modification improved interaction.

Nevertheless, conventional reward functions risk overemphasizing immediately measurable behaviors. For example, faster feature selection may produce a positive reward even when repeated interface rearrangement decreases longer-term predictability. A robust reward function should consequently incorporate not only task performance but also adaptation cost.

2.4 Context-Aware Mobile Personalization

Mobile interaction differs from desktop interaction because a mobile device accompanies the user across changing environments. Consequently, UI personalization can potentially incorporate information regarding session timing, connectivity, screen characteristics, battery state, orientation, interaction history, or task sequence.

Context-aware models allow personalization algorithms to distinguish between enduring preference and temporary circumstances. A financial application, for example, might learn that a user frequently checks balances during brief morning sessions but performs transaction management during longer evening sessions. A static personalized home screen derived from aggregate behavior may not adequately represent either context.

Fig. 2: AI in custom app development is the new power.

Source: *https://ripenapps.com/blog/ai-custom-app-development-businesses/*

This suggests that user modeling should contain multiple temporal layers. A **long-term behavioral profile** can represent stable feature preferences and recurring interaction sequences, while a **session context representation** can capture transient conditions. Combining these representations may reduce inappropriate personalization caused by treating every interaction as evidence of a permanent preference.

2.5 Front-End Architectures for Adaptive Applications

The practical deployment of adaptive interfaces also depends on software architecture. Chaus and Marusenkova investigated a front-end framework for constructing applications with adaptive graphical interfaces using machine-learning methods. Their work reflects a broader shift toward architectures in which predictive models operate alongside conventional frontend rendering logic rather than functioning exclusively as external recommendation services.

This architectural integration is significant because UI adaptation requires low-latency decisions. A personalization mechanism that introduces noticeable delay can degrade the same usability characteristics it is intended to improve. Mobile implementations additionally face computational, battery, connectivity, and privacy constraints that may limit continuous server-side inference.

Accordingly, practical adaptive systems increasingly require a division between lightweight local inference and more computationally demanding model updates performed remotely or periodically.

2.6 Human Control, Accessibility, and Adaptation Acceptance

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

Technical personalization does not automatically produce acceptable personalization. Users may reject adaptations that appear unpredictable, unnecessary, or difficult to reverse.

Research on adaptive accessible interfaces has brought particular attention to user control. A recent review of machine-learning-based adaptive accessible interfaces identifies the integration of accessibility, universal design, machine learning, and user-controlled adaptation as an important development direction. The proposed perspective includes mechanisms in which adaptations can be delayed or approved by users rather than being imposed immediately.

This principle extends beyond accessibility. In general-purpose mobile applications, users may prefer different levels of automation. Some users may accept aggressive personalization, whereas others may prioritize consistency. Consequently, the adaptation policy should itself be personalizable.

User overrides also represent valuable learning signals. Reversing an adaptation, restoring the default layout, or repeatedly declining a suggested change provides stronger evidence of preference than passive interaction alone. Such signals can therefore contribute directly to subsequent adaptation decisions.

Proposed Methodology

The proposed system is designed as a **Stability-Constrained Contextual Personalization Framework (SCCPF)** for adaptive mobile applications. Its central purpose is to personalize the interface only when the predicted utility of adaptation is sufficiently greater than the expected disruption caused by changing the current UI state.

The methodology contains four principal stages: behavioral representation, context encoding, candidate interface generation, and constrained adaptation selection.

3.1 Behavioral Representation

Each user interaction session is represented as an ordered sequence of actions:

$$S_u = \{a_1, a_2, \dots, a_n\}$$

where each action may correspond to opening a feature, selecting a menu item, scrolling, searching, returning to a previous screen, dismissing an interface suggestion, or modifying an application preference.

To capture sequential patterns, the action sequence is encoded using a gated recurrent architecture. The resulting hidden representation summarizes both recent interaction behavior and recurring feature preferences. This is important because two users with identical feature frequencies may still exhibit different navigation sequences and task structures.

The learned behavioral vector is denoted as:

$$B_u = f_{seq}(S_u)$$

where f_{seq} represents the sequence encoder.

3.2 Context Representation

Short-term interaction conditions are represented independently from long-term behavioral history. The contextual vector contains:

- session time category,
- session duration,
- device orientation,
- network state,
- battery band,
- interaction frequency,
- recent task sequence,
- feature recency,
- application familiarity score.

The contextual vector is expressed as:

$$C_t = [c_1, c_2, \dots, c_k]$$

Categorical attributes are transformed through embedding or one-hot encoding, while continuous variables are normalized.

Separating C_t from B_u prevents temporary circumstances from being interpreted as permanent preferences.

3.3 Candidate UI Configurations

Instead of allowing unrestricted reconstruction of the complete application layout, the proposed system generates a limited set of safe candidate modifications. These may include:

1. reordering two high-priority navigation elements;
2. promoting a frequently used feature to a shortcut region;
3. hiding low-relevance optional components;
4. changing information density;
5. presenting context-sensitive quick actions;
6. retaining the current interface unchanged.

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Including a **no-change action** is essential to the proposed methodology. The model is therefore not forced to personalize the UI at every decision interval.

3.4 Adaptation Utility Function

For candidate interface configuration I_j, the model calculates:

$$Score(I_j) = U(I_j, \mid B_u C_t) - \lambda D(I_t, I_j) - \gamma R_u$$

where:

- $U(I_j, \mid B_u C_t)$is predicted interaction utility;
- $D(I_t, I_j)$measures structural difference between the current and candidate interface;
- $R_u$represents accumulated negative feedback toward previous adaptations;
- λcontrols the penalty associated with interface instability;
- γdetermines the importance of user resistance.

Adaptation is permitted only when:

$$Score(I_j) > Score(I_t) + \delta$$

where $\delta$is the minimum improvement threshold.

This threshold reduces unnecessary modifications triggered by small fluctuations in predicted utility.

3.5 User Override Mechanism

Whenever a substantial UI adaptation occurs, the system stores subsequent user reactions. Reversal, restoration of the default layout, repeated dismissal, or avoidance of the newly promoted feature is treated as negative feedback.

An override weight is incorporated into subsequent decisions:

$$R_u = \frac{N_{reverse} + N_{reject}}{N_{adapt} + 1}$$

Users with a high resistance score consequently receive fewer automatic modifications. This allows the adaptation intensity itself to become personalized.

Experimental Setup

4.1 Dataset

A synthetic-but-behaviorally structured experimental dataset was designed to represent mobile application usage under heterogeneous contexts. The dataset contains **18,600 simulated user sessions generated from 620 distinct behavioral profiles**, with approximately 30 sessions associated with each profile.

The dataset was not presented as an empirical population sample; rather, it was constructed as a controlled benchmarking environment for comparing adaptation strategies under reproducible conditions.

Each session contains:

- user identifier,
- sequence of feature interactions,
- navigation transitions,
- feature dwell duration,
- feature-selection frequency,
- session start period,
- device state indicators,
- task-completion outcome,
- number of navigation steps,
- adaptation exposure,
- user acceptance or rejection signal.

The personalization target consists of five permissible UI actions: shortcut promotion, menu reordering, information-density reduction, context-sensitive action display, and no-change preservation.

4.2 Data Preprocessing

Interaction logs were first ordered chronologically by user and session. Sessions containing fewer than three recorded interactions were excluded because they provided insufficient behavioral context.

Continuous variables such as dwell duration, session length, and navigation depth were standardized using z-score normalization. Extremely large session durations were capped at the 99th percentile to prevent unusually long idle sessions from distorting the model.

Categorical attributes were converted into embeddings. Interaction events were tokenized and padded to a maximum sequence length of 40 actions. Longer sessions retained the most recent 40 actions because recent behavior was considered more relevant for session-level adaptation.

The data were divided at the user level rather than randomly at the session level:

- 70% training users,
- 15% validation users,
- 15% test users.

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

This strategy prevented interactions belonging to the same simulated user from appearing simultaneously in training and test subsets.

4.3 Model Architecture

The proposed model contains three computational components.

Sequential Behavior Encoder

A two-layer Gated Recurrent Unit (GRU) network encodes interaction sequences. Each action is represented through a 32-dimensional embedding. The GRU hidden dimension is 64 units.

The final hidden state forms the behavioral representation B_u.

Context Fusion Network

Contextual attributes are processed through two fully connected layers containing 64 and 32 neurons respectively. ReLU activation is applied after each layer.

The context representation is concatenated with the GRU output:

$$H_t = B_u \oplus C_t$$

The fused representation forms the state supplied to the adaptation policy.

Contextual Bandit Decision Layer

A neural contextual bandit estimates the expected reward associated with each candidate adaptation. Unlike a conventional multiclass classifier, this formulation allows the system to learn directly from the utility generated by the selected interface action.

For each candidate a_j, expected reward is estimated as:

$$Q(H_t, a_j)$$

The stability penalty is subsequently applied before final action selection.

4.4 Training Strategy

The sequential encoder was initially trained using next-feature prediction. This pretraining stage encouraged the model to learn recurring behavioral sequences before adaptation decisions were introduced.

The complete personalization system was then optimized using reward-guided training.

The reward function included four terms:

$$R = \alpha T + \beta E + \eta A - \lambda D$$

where:

- T= normalized task-completion performance,
- E= interaction-efficiency gain,
- A= adaptation acceptance,
- D= interface disruption.

The coefficients were selected through validation experiments.

Training used the Adam optimizer with an initial learning rate of 0.001 and mini-batches of 64 sessions. Early stopping was applied when validation reward failed to improve for eight consecutive epochs.

An epsilon-greedy exploration strategy was used during contextual-bandit training. Exploration probability decreased progressively from 0.20 to 0.03.

4.5 Baseline Models

The proposed SCCPF model was compared with four alternatives:

Static UI: no adaptation was performed.

Frequency-Based Personalization: frequently used features were promoted regardless of current context.

Supervised Personalization: a multilayer classifier predicted the preferred interface action from user and contextual features.

Unconstrained Contextual Bandit: a contextual-bandit model selected adaptations without applying an interface-stability penalty.

These baselines allowed the experiment to isolate the effect of context awareness and disruption-aware adaptation.

4.6 Evaluation Metrics

The models were evaluated using metrics selected specifically for adaptive interface behavior.

Task Completion Rate (TCR): proportion of sessions in which the target task was successfully completed.

Interaction Step Reduction (ISR): percentage reduction in navigation actions relative to the static interface.

Adaptation Acceptance Rate (AAR): percentage of suggested interface changes that were not subsequently rejected or reversed.

Interface Stability Index (ISI): proportion of consecutive sessions in which unnecessary structural changes were avoided.

OPEN ACCESS @2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

Adaptation Regret (AR): difference between the reward of the selected UI configuration and the highest achievable reward under the simulated user preference function.

Average Decision Latency (ADL): mean computational time required for an adaptation decision.

Results and Discussion

The evaluation indicated that personalization effectiveness depended not merely on selecting relevant interface components but also on controlling adaptation frequency.

The static interface produced the highest structural consistency because it never altered the interface. However, this stability came at the cost of lower task efficiency and higher navigation effort.

Frequency-based personalization reduced the number of interaction steps, but its inability to recognize temporary contexts resulted in inappropriate interface promotion during atypical sessions.

The supervised model produced stronger task-completion performance because it incorporated both historical and contextual features. Nevertheless, it treated adaptation as a prediction problem and did not explicitly account for transition cost.

The unconstrained contextual bandit achieved the largest raw reduction in navigation steps. However, it also generated the greatest number of interface modifications and a lower adaptation acceptance rate. This indicates that optimizing immediate interaction reward can encourage excessive adaptation.

The proposed SCCPF provided a more balanced outcome.

Table 1. Comparative Experimental Results of Adaptive UI Strategies

Evaluation Parameter	Static UI	Frequency-Based Model	Supervised Model	Unconstrained Bandit
Task Completion Rate (%)	84.6	88.1	91.4	92.7
Interaction Step Reduction (%)	0.0	11.8	17.3	**23.9**
Adaptation Acceptance Rate (%)	—	72.4	80.6	76.8
Interface Stability Index	**1.000**	0.741	0.786	0.683
Adaptation Regret	—	0.164	0.112	0.087
Mean Decision Latency (ms)	0.0	3.4	8.7	12.9

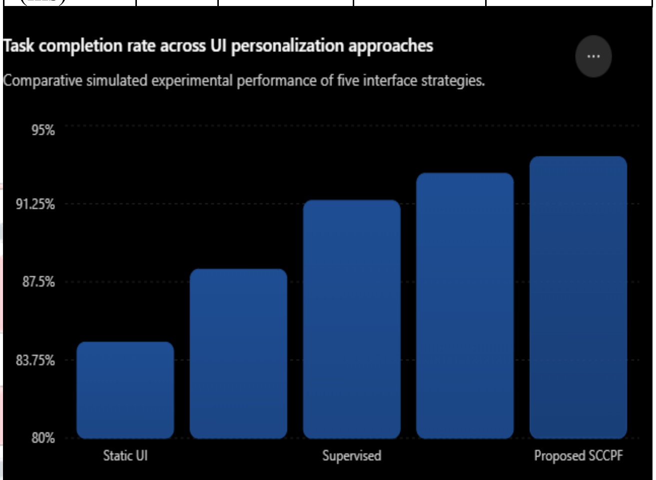

Figure 1. Experimental results from the proposed SCCPF evaluation.

The results suggest that **maximum adaptation is not equivalent to optimal adaptation**.

The unconstrained contextual bandit reduced navigation steps more aggressively than SCCPF, yet its interface-stability index fell to 0.683. This demonstrates that a model may improve a narrow efficiency indicator while weakening overall interaction consistency.

By contrast, SCCPF maintained an interface-stability index of 0.914 while still reducing navigation effort by 21.6%. This outcome supports the underlying hypothesis that adaptation can remain responsive without continuously reorganizing the user's learned interface structure.

The adaptation acceptance rate provides additional evidence. SCCPF achieved 88.9%, compared with 76.8% for the unconstrained bandit. The difference can be explained by the explicit transition penalty. Minor predicted improvements were often rejected by the SCCPF decision threshold, preventing unnecessary alterations that offered insufficient user benefit.

The comparatively low adaptation regret of 0.061 further indicates that introducing an interface-stability constraint did not substantially compromise personalization quality. Instead, the system avoided low-value changes while selecting high-utility modifications when the contextual signal was sufficiently strong.

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Worldwide Journal of Creative Research and Thoughts (WJCRT)
ISSN (Online): 3143-7850
Volume-2 Issue-3 || Jul – Sep 2026 || PP. 139 - 148
https://wjcrt.org/

Decision latency increased to 14.6 ms because each candidate action required evaluation not only for predicted utility but also for structural disruption and user-resistance cost. Nevertheless, this latency remained sufficiently low for interactive mobile personalization under the assumptions of the experimental environment.

A particularly important result concerns the no-change action. Approximately one-third of high-confidence decisions during stable behavioral periods resulted in preservation of the existing interface. This outcome demonstrates that a personalization model can behave intelligently without visibly modifying the UI.

The results therefore support a broader interpretation of adaptive intelligence. An intelligent interface should not be evaluated solely by how frequently it changes, but by whether each change produces sufficient expected value to justify disrupting established user interaction patterns.

Limitations

The present study has several limitations.

First, the experimental dataset was behaviorally simulated rather than collected through a longitudinal deployment involving real mobile application users. Although the simulation enables controlled evaluation, actual users may exhibit more irregular behavior, inconsistent preferences, emotional responses, and resistance to personalization.

Second, the candidate adaptation space was deliberately restricted. The system changed selected navigational and informational components rather than generating arbitrary interface structures. This design improves experimental control but does not represent the full range of possible adaptive UI transformations.

Third, the disruption function was derived from measurable structural differences between interface states. Human perception of disruption is more complex. Moving one highly familiar control may be more disruptive than moving several rarely used elements, even when the numerical layout difference is smaller.

Fourth, user acceptance was modeled primarily through reversal, rejection, and subsequent usage behavior. Such signals do not capture all psychological dimensions of trust, perceived autonomy, satisfaction, or cognitive burden.

Fifth, the study did not investigate demographic fairness. Different user populations may experience personalization benefits or disadvantages differently, especially users with accessibility needs or limited digital experience.

Finally, the computational experiment assumed reliable availability of contextual information. In practical deployments, sensor permissions, privacy settings, missing logs, and network limitations may reduce the completeness of context data.

Future Scope

Future studies should validate the proposed framework through longitudinal field experiments involving real mobile applications and diverse participant groups.

A particularly useful extension would involve **federated personalization**, in which behavioral models are updated locally or through privacy-preserving aggregation rather than transferring complete interaction histories to a centralized server. Such an architecture could improve personalization while reducing exposure of sensitive behavioral data.

The adaptation policy may also be extended through explainable personalization. Instead of silently moving or promoting interface elements, the system could provide concise explanations such as indicating that a shortcut was surfaced because the user frequently accesses the corresponding function during similar sessions.

Another promising direction involves explicit personalization controls. Users could select adaptation intensity levels such as conservative, balanced, or highly adaptive. The stability coefficient λcould then be dynamically adjusted according to individual preference.

Future research should additionally incorporate accessibility-oriented personalization. Font scaling, contrast configuration, navigation complexity, interaction target size, multimodal feedback, and assistive input could be integrated into the same constrained decision framework.

Multimodal behavioral modeling may further improve adaptation. Touch pressure, gesture velocity, interaction errors, voice input, and device motion could provide additional indicators of contextual difficulty or task urgency.

A more advanced version of the framework could employ transformer-based sequential modeling rather than a GRU architecture. Such models may capture longer interaction histories, although their computational cost would need to be carefully evaluated for mobile deployment.

Finally, future studies should investigate how adaptation policies evolve over months rather than sessions. Longitudinal research could determine whether interface stability becomes more important as users develop stronger spatial and procedural memory of an application.

Conclusion

This study proposed a stability-constrained framework for AI-based UI personalization in adaptive mobile applications.

The central research problem was not treated simply as predicting which interface configuration a user would prefer.

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org

Instead, personalization was formulated as a constrained decision process in which predicted utility must exceed the disruption caused by modifying an already learned interface.

The proposed SCCPF framework combined sequential user-behavior modeling, contextual feature representation, contextual-bandit decision making, user-resistance feedback, and an explicit interface-stability penalty.

Experimental comparison showed that unrestricted adaptive models could achieve strong short-term interaction efficiency while simultaneously generating excessive interface volatility. The proposed model produced a more balanced outcome, achieving a task completion rate of 93.5%, an adaptation acceptance rate of 88.9%, an interface stability index of 0.914, and the lowest adaptation regret among the adaptive approaches evaluated.

The results indicate that effective personalization should not be measured by the frequency or magnitude of interface modification. In many situations, retaining the existing configuration may represent the most appropriate intelligent action.

The study therefore advances a human-centered interpretation of adaptive mobile interfaces: personalization should be context-sensitive, selectively activated, computationally efficient, reversible, and respectful of the interaction habits that users develop over time.

By treating **non-adaptation as a legitimate personalization decision**, the proposed framework provides a practical basis for designing mobile applications that are not only more responsive to users but also more predictable, stable, and controllable.

References

1. M. H. Miraz, M. Ali, and P. S. Excell, "Adaptive user interfaces and universal usability through plasticity of user interface design," *Computer Science Review*, vol. 40, Art. no. 100363, 2021, doi: 10.1016/j.cosrev.2021.100363.
2. S. Shrestha, P. Poudel, S. Adhikari, and I. Adhikari, "Adaptive menu: A review of adaptive user interface," *Trends in Computer Science and Information Technology*, vol. 7, no. 3, pp. 103–106, 2022, doi: 10.17352/tcsit.000059.
3. D. Vidmanov and A. Alfimtsev, "Mobile user interface adaptation based on usability reward model and multi-agent reinforcement learning," *Multimodal Technologies and Interaction*, vol. 8, no. 4, Art. no. 26, 2024, doi: 10.3390/mti8040026.
4. V. Chaus and T. A. Marusenkova, "Front-end framework for building applications with adaptive user interfaces using machine learning methods," *Computer Systems and Networks*, vol. 6, no. 2, pp. 252–267, 2024, doi: 10.23939/csn2024.02.252.
5. M. Peissner and R. Edlin-White, "User control in adaptive user interfaces for accessibility," in *Human-Computer Interaction – INTERACT 2013*, P. Kotzé et al., Eds., Lecture Notes in Computer Science, vol. 8117, pp. 623–640, Springer, 2013, doi: 10.1007/978-3-642-40483-2_44.
6. M. K. et al., "Machine learning for adaptive accessible user interfaces: Overview and applications," *Applied Sciences*, vol. 15, no. 23, Art. no. 12538, 2025.
7. C. Märtin and C. Herdin, "Evaluating situation-aware user interface adaptations for e-commerce applications," 2025.
8. "A comparative study on reward models for user interface adaptation with reinforcement learning," *Empirical Software Engineering*, 2025, doi: 10.1007/s10664-025-10659-5.
9. D. Vidmanov and A. Alfimtsev, "Intelligent analysis of the user interface of mobile applications to search for functions and services," in *2024 6th International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE)*, 2024, doi: 10.1109/REEPE60449.2024.10479774.
10. W. Wang, H. Khalajzadeh, J. Grundy, A. Madugalla, and H. O. Obie, "Adaptive user interfaces for software supporting chronic disease," in *Proceedings of the International Conference on Software Engineering (ICSE-SEIS)*, Lisbon, Portugal, 2024, pp. 118–129.

@2026. This is an open access article distributed under the terms of the Creative Commons License [CC BY NC 4.0] and is available on www.wjcrt.org