AURA

Adaptive User-driven Retrieval Architecture

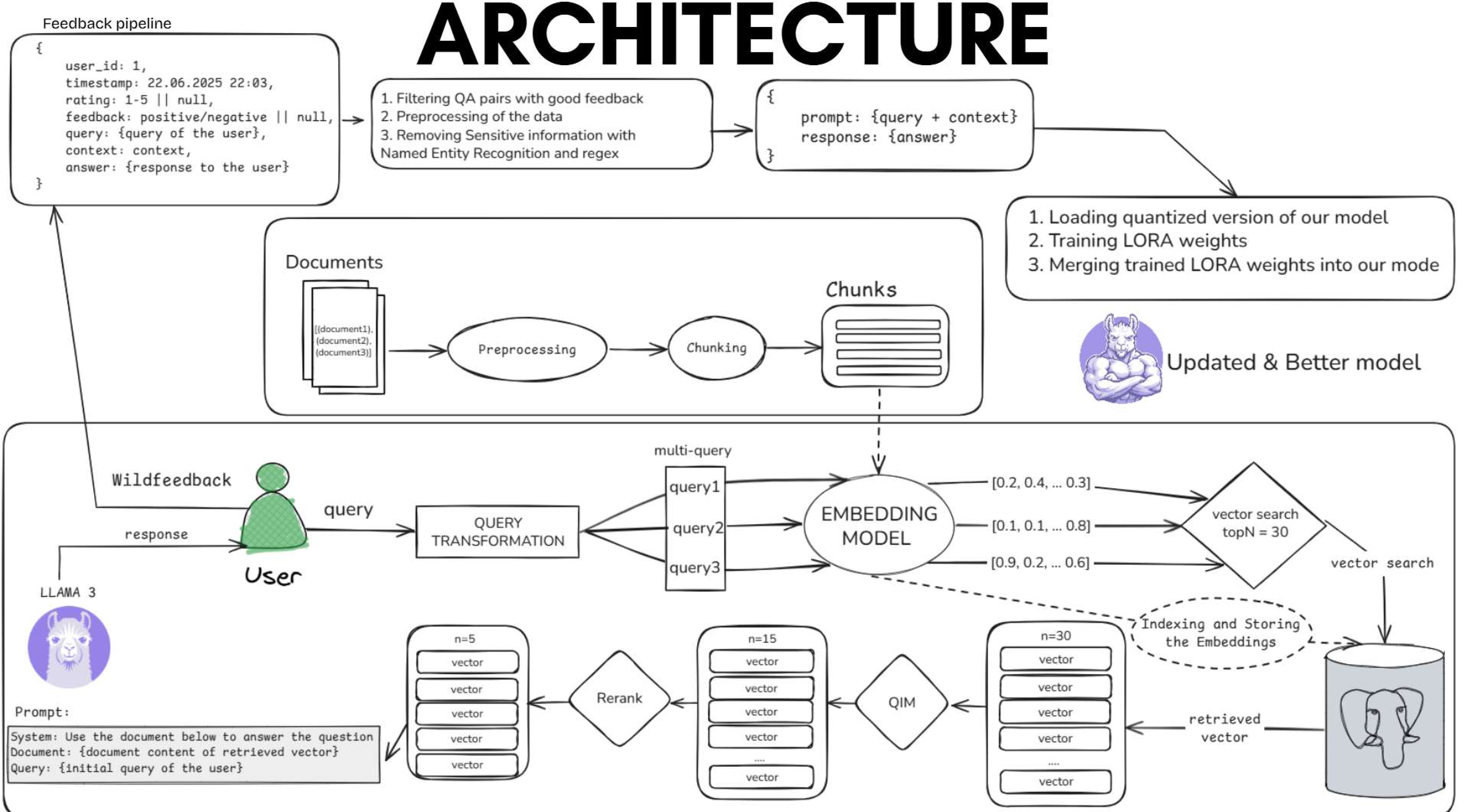
Presented By EWKI

IDEA SUMMARY

AURA is a feedback-powered RAG pipeline that uses a multi-query retriever, reranker and Quantized Influence Measure to pull and rank relevant context for an 8 B-parameter LLaMA, with user reaction gauged by the newly research WildFeedback algorithm.

A modular approach to implementation gives us an extreme competitive edge, because it ensures easy scalability and testing.

ARCHITECTURE



Feedback pipeline

```
{
    user_id: 1,
    timestamp: 22.06.2025 22:03,
    rating: 1-5 || null,
    feedback: positive/negative || null,
    query: {query of the user},
    context: context,
    answer: {response to the user}
}
```

1. Filtering QA pairs with good feedback
2. Preprocessing of the data
3. Removing Sensitive information with Named Entity Recognition and regex

```
{
    prompt: {query + context}
    response: {answer}
}
```

1. Loading quantized version of our model
2. Training LORA weights
3. Merging trained LORA weights into our mode

Updated & Better model

Prompt:

```
System: Use the document below to answer the question
Document: {document content of retrieved vector}
Query: {initial query of the user}
```

TECHNOLOGY STACK

Embeddings

Google LaBSE: multilingual embedding model, which supports kazakh and russian

LLM

Llama-3.1-Sherkala-8B[4]

RAG Pipeline

LangChain

QIM

Custom Implementation

USER FEEDBACK

Automated Feedback based on user reaction (Wildfeedback)

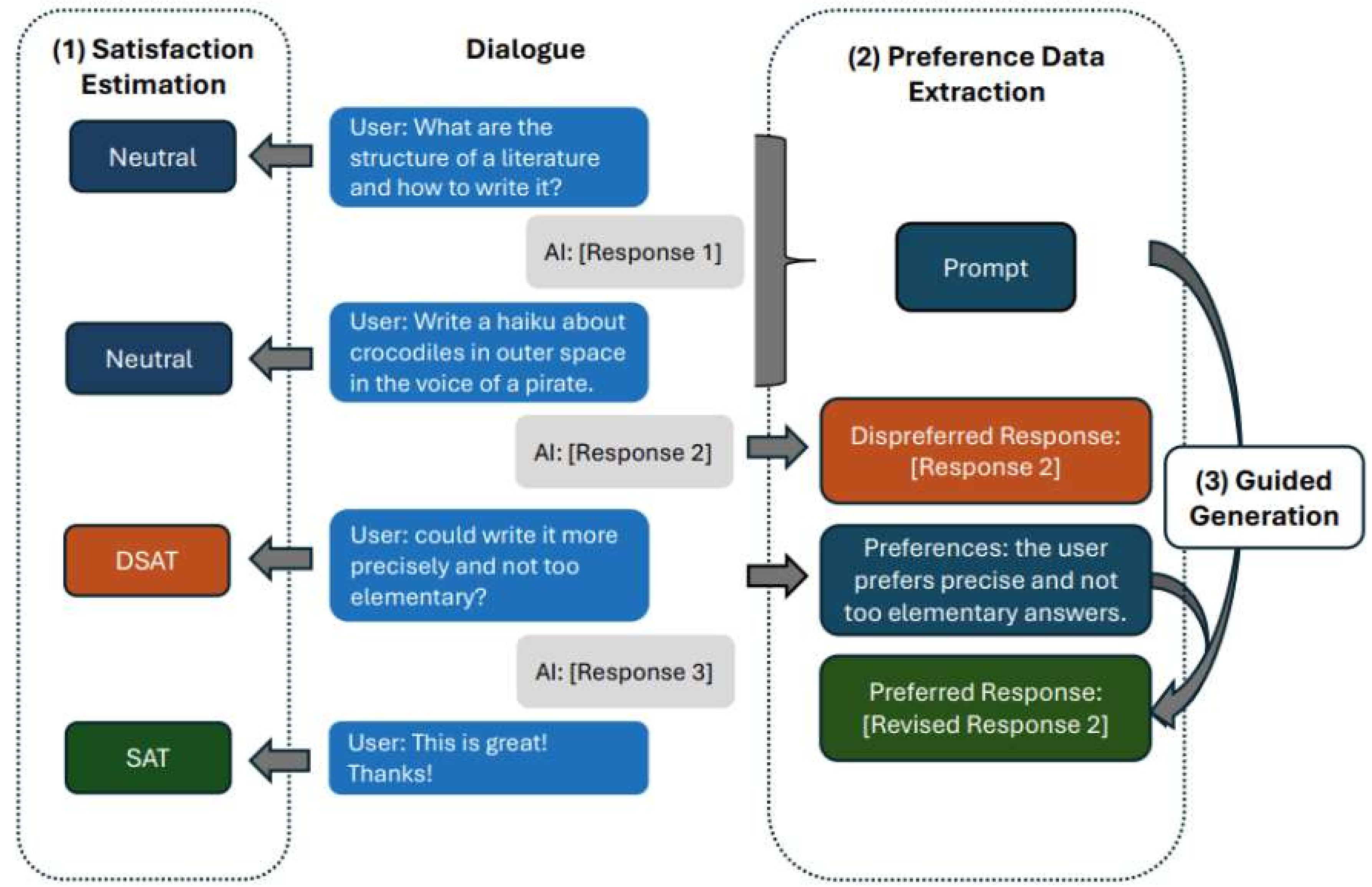
WildFeedback is a framework that fine-tunes language models using real user feedback from conversations — no manual annotation needed.

Input: Real user-model chat logs

Output: A high-quality preference dataset — auto-generated

Core Idea:

- Detects if the user is satisfied or dissatisfied from their messages (e.g., "thanks", "please revise", "not quite right")
- Extracts user preferences (e.g., "more concise", "formal tone"),
- Builds training pairs: preferred vs. dispreferred responses.



(1) Satisfaction Estimation

Neutral

Neutral

DSAT

SAT

Dialogue

User: What are the structure of a literature and how to write it?

AI: [Response 1]

User: Write a haiku about crocodiles in outer space in the voice of a pirate.

AI: [Response 2]

User: could write it more precisely and not too elementary?

AI: [Response 3]

User: This is great! Thanks!

(2) Preference Data Extraction

Prompt

Dispreferred Response: [Response 2]

Preferences: the user prefers precise and not too elementary answers.

Preferred Response: [Revised Response 2]

(3) Guided Generation

WHY WILDFEEDBACK?

OpenAI’s Human-annotated RLHF

15-30% Improvement[1]

Wildfeedback

10-20% Improvement[2]

Pilot Plan

DELIVERABLE	DURATION	SUCCESS METRIC
CORE: Basic RAG and Reranker	10-14 days	1) RAG retrieves 30 related documents 2) Reranker successfuly narrows down to 5
Wildfeedback	14-18 days	50% User Positive and Negative Feedback is automatically identified, collected and sent to dataset for finetuning
LoRA Adapter Integration	6 days	Original model weights remain frozen; additional low-rank matrices are trained with ≤ 50% of the parameters of the full layer
QIM	3 days	QIM successfuly narrows down from 30 related documents to 15
Basic TK UI	2 days	UI Has Chat, shows model responses.

ADVANTAGES

Continuous, automated fine-tuning

Easy testing and integration

Low resource costs

RESOURCE MANAGEMENT | TEAM

Computer Infrastructure with at least 40 GB VRAM GPU(L40 should do)

X1

AI Engineer

X2

Backend Microservices Developer

X1

Data Engineer

X1

[1]
T. Shi, Z. Wang, L. Yang, Y.-C. Lin, Z. He, M. Wan, P. Zhou, S. K. Jauhar, S. Chen, S. Xia, H. Zhang, J. Zhao, X. Xu, X. Song, and J. Neville,
"WildFeedback: Aligning LLMs With In-situ User Interactions and Feedback,"
arXiv preprint arXiv:2408.15549, 2024.

[2]
Y. Chai, H. Sun, H. Fang, S. Wang, Y. Sun, and H. Wu,
"MA-RLHF: Reinforcement Learning from Human Feedback with Macro Actions,"
arXiv preprint arXiv:2410.02743, 2024.

[3]
Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (n.d.). LoRA: Low-Rank Adaptation of Large Language Models (Version 2). Microsoft Corporation. https://github.com/microsoft/LoRA