

ROLE	RAG only	LoRA only	LoRA + RAG + API
User:	Hi, I'm having trouble accessing my dashboard. It says "permission denied".	Hi, I'm having trouble accessing my dashboard. It says "permission denied".	Hi, I'm having trouble accessing my dashboard. It says "permission denied".
Backend:	<i>RAG performs vector search. Retrieves a chunk from the knowledge base that includes: "Permission Denied errors can occur due to lack of access rights. Contact your administrator or request access through the internal portal."</i>	<i>LoRA-fine-tuned model generates a soft, empathetic response based on how it was trained to respond to errors. No document retrieval.</i>	<i>RAG kicks in: vector search looks up internal documentation or past tickets related to "dashboard access" and "permission denied" errors. The LLM, fine-tuned with LoRA, understands company-specific context and phrasing.</i>
Bot:	According to our documentation: "Permission Denied errors can occur due to lack of access rights. Contact your administrator or request access through the internal portal."	I'm sorry to hear that. It sounds like you might not have the necessary access rights. Let's figure it out together.	I'm sorry you're running into that. Just to confirm, are you using your work email to log in?
Outcome:	❌ Sounds robotic and generic. No personalization, no empathy.	<div>✅ Nice tone, more human — learned from fine-tuned support examples</div> <div>❌ But vague — model doesn't know current access policies or dashboard details unless included in training.</div>	LLM uses fine-tuned dialogue style to collect context. LoRA helps model sound helpful and follow internal tone guidelines.
User:	But I thought I should have access?	But I thought I should have access?	Yes, it's john.doe@company.com.
Backend:	<i>RAG does another search using this input. Pulls a document chunk related to dashboard access tiers, which mentions: "Sales team has default access; Marketing requires approval."</i>	<i>Model tries to recall from what it was fine-tuned on. Maybe learned that some departments have different access levels, but it's guessing now.</i>	<i>Model infers from context that permissions may be role-based. LoRA fine-tuning has taught it company-specific access rules, e.g., "Marketing doesn't have dashboard X by default. Sales does."</i>
Bot:	The Sales team has default access to the dashboard. Marketing team members require manager approval.	Are you part of the Sales or Support team? I believe they usually have default access.	Thanks, John. Are you part of the Sales team or another department?
Outcome:	<div>✅ Factual info from docs, but</div> <div>❌ no personalized reasoning — the bot doesn't ask which team the user is in or infer it from earlier messages.</div>	<div>✅ Reasonable guess based on training</div> <div>❌ If access rules changed or are different in your org, model won't know — no dynamic source</div>	✅ Diagnostics with probing questions
User:	I'm in Marketing.	I'm in Marketing.	I'm in Marketing.
Backend:	<i>Now it could re-query RAG and get the same doc again. But there's no logic engine to act on that info — it won't connect "User is in Marketing" + "Marketing needs approval" = "Bot should offer to send approval."</i>	<i>No training data. Probably a guess</i>	<i>LLM creates a summary of the user's issue and queries RAG. For example: RAG retrieves internal KB article: "Marketing users require manager approval for Sales Dashboard access."</i>
Bot:	As mentioned, Marketing team members require manager approval.	Hmm, it's possible that Marketing doesn't have access by default. You may need to contact your admin to request it.	That helps. Based on your department, you might not have access to the Sales Analytics Dashboard by default. I'll check your permissions now. ... (finding article) I found that Marketing users need manager approval to access that dashboard. I can send a request to your team lead if you'd like?
Outcome:	❌ Repeats info instead of offering help or next step. No initiative. No stored context or flow logic.	<div>✅ Follows tone well</div> <div>❌ Uncertain answer; model is relying on memory from training data.</div> <div>❌ No ability to confirm or check policies — could be outdated..</div>	<div>✅ RAG retrieved this rule from internal access policy docs.</div> <div>✅ LoRA helped model phrase this politely and align with support tone.</div>
User:	Can you send the request for me?	Can you send the request for me?	Yes, please do that.
Backend:	<i>RAG finds nothing about sending requests. The document may mention a manual process or a form, but unless it's clearly worded like "support bot sends approval," the bot won't take action.</i>	<i>Model was not trained to trigger real actions or integrate with systems. It can generate the idea of a next step, but can't actually perform one.</i>	<i>Bot sends request via internal API or ticket system. Final output generated by LLM, using LoRA fine-tuned style + prefilled template for such cases.</i>
Bot:	I was not able to find relevant information on sending a request. Please contact your team lead or use the internal portal.	I'm not able to submit the request directly, but I recommend reaching out to your manager or submitting a request through the usual form.	Done. I've sent a request to your manager. You'll get access once they approve — usually within 1 business day. Let me know if you need anything else!
Outcome:	❌ No workflow integration. ❌ No ability to generalize intent. ❌ No soft handoff or helpfulness.	<div>✅ Helpful intent</div> <div>❌ Can't actually help. No API calls. No dynamic routing or automation.</div>	✅ Helpful intent + Resolution