DOI: 10.34185/1991-7848.itmm.2026.01.087

LARGE LANGUAGE MODELS AS A ROUTING LAYER IN MULTI-CHANNEL MESSENGER SYSTEMS

Poliakov O.M. [ORCID]

Co-Founder & CTO Touchly.io, master's degree. Ukraine

Abstract. *Businesses that handle customer communication across multiple messenger channels face a recurring operational problem: getting each incoming message to the right handler without human involvement. Rule-based approaches — keyword filters, button menus, static decision trees — break down quickly when users write freely. This paper describes a middleware architecture where a large language model sits between the incoming message stream and the business logic layer, classifying each message by intent and triggering the appropriate action directly. The model returns structured JSON via a tool use mechanism rather than generating free text, which keeps latency predictable and integration straightforward. The system has been deployed commercially, integrated with platforms including PipeDrive, HubSpot, and Zoho CRM, and is actively used by more than one thousand businesses. Routing errors dropped from 35% to 4% after rollout.*

Keywords: *large language models, intent detection, message routing, tool use, middleware, omnichannel, CRM integration, FastAPI.*

Introduction

Messenger platforms have become the primary channel for business-to-client communication [1]. WhatsApp, Telegram, Viber and similar services handle everything from appointment reminders to support requests — and the volume keeps growing. The practical challenge is not sending messages, but handling the ones that come back.

Most current solutions rely on rigid scripts: numbered menus, keyword triggers, or flowcharts. These work for narrow, predictable inputs and fail the moment a customer phrases something slightly differently [2]. This paper examines whether a large language model can replace that classification layer with something more robust at production-grade latency and cost.

Objective. To increase the automation level of message routing in messenger channels by applying large language models for accurate intent classification, reducing manual operator workload and improving first-response time.

Main Material

The proposed architecture introduces a classification middleware between the messenger gateway and downstream business systems (fig. 1). The implementation uses Python / FastAPI for the API layer and asyncio for concurrent handling. Each incoming message is forwarded to a cloud-hosted LLM inference endpoint via HTTPS before any routing decision is made.

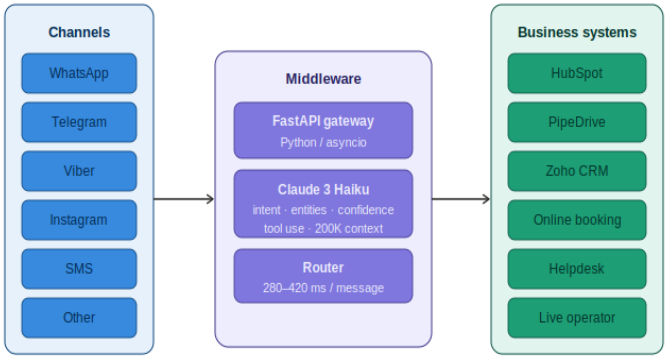

Figure 1 – System architecture

The system applies the tool use mechanism of Claude 3 Haiku (Anthropic) — a compact model optimised for production: 200 K-token context window, native tool use, multilingual input [3]. Instead of generating prose, the model returns a JSON object with three fields: *intent*, *entities*, *confidence*. This output is machine-readable with no post-processing. The routing layer matches the intent to a configuration table and triggers the corresponding action — CRM lookup, calendar check, ticket creation, or operator escalation. Intent classes are managed via system prompt; no retraining is required when business logic changes [4].

The processing sequence for a single message is shown in fig. 2.

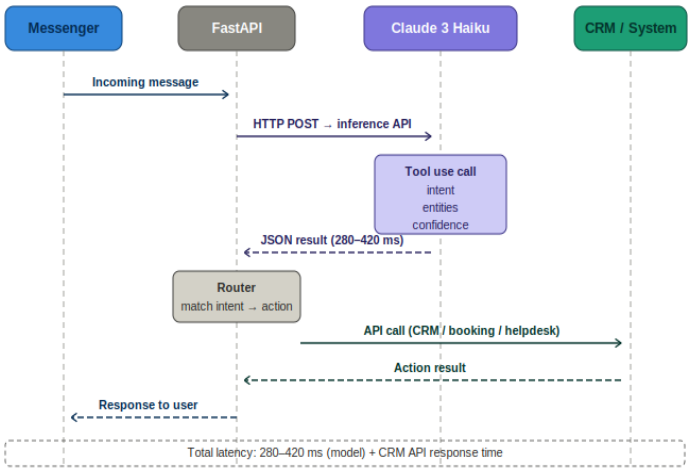

Figure 2 – Message processing sequence

Performance characteristics of the deployed system: average processing time 280–420 ms; latency growth under 50 concurrent requests — under 15%; up to 20 intent classes reconfigurable via prompt; cost per message under $0.001.

The system is integrated with more than five CRM platforms (PipeDrive, HubSpot, Zoho CRM, online booking and helpdesk tools). Connected businesses exceed one thousand. Table 1 shows the impact based on a modelled service-sector scenario.

Table 1

Performance before and after deployment

Metric	Before	After
Auto-resolved requests	20%	65%
First response time	5 min	< 1 sec
Routing errors	35%	4%
Cost per request	—	< $0.001

The architecture is horizontally scalable: new channels and intent classes are added without changes to core routing logic.

Conclusions

LLM-based intent detection via Claude 3 Haiku is a viable replacement for rule-based routing in commercial messenger systems. The tool use mechanism delivers structured, machine-readable output with 280–420 ms latency and under $0.001 per

message. Production deployment across more than one thousand businesses confirms scalability of the approach.

ЛІТЕРАТУРА / REFERENCE

1. Brown T. et al. Language Models are Few-Shot Learners. Advances in NeurIPS. 2020. Vol. 33. P. 1877–1901.
2. Liu P. et al. Pre-train, Prompt, and Predict. ACM Computing Surveys. 2023. Vol. 55. No. 9. P. 1–35.
3. Schick T. et al. Toolformer: Language Models Can Teach Themselves to Use Tools. NeurIPS. 2023. Vol. 36.
4. Wei J. et al. Chain-of-Thought Prompting Elicits Reasoning in LLMs. NeurIPS. 2022. Vol. 35. P. 24824–24837.

ЗАСТОСУВАННЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ ЯК ШАРУ МАРШРУТИЗАЦІЇ У БАГАТОКАНАЛЬНИХ МЕСЕНДЖЕР-СИСТЕМАХ

О.М. Поляков

Анотація. *Компанії, що ведуть комунікацію з клієнтами через кілька месенджер-каналів, стикаються зі спільною проблемою: автоматично спрямувати кожне вхідне повідомлення до правильного обробника. Підходи на основі правил — фільтри ключових слів, кнопкові меню — не справляються з довільним введенням У роботі описано middleware-архітектуру, де велика мовна модель Claude 3 Haiku розташована між потоком повідомлень і бізнес-логікою: через механізм tool use вона класифікує намір та повертає структурований JSON безпосередньо шару маршрутизації. Система впроваджена у понад тисячі компаній, інтегрована з PipeDrive, HubSpot, Zoho CRM та іншими платформами. Помилки маршрутизації скоротилися з 35% до 4%.*

Ключові слова: *великі мовні моделі, intent detection, маршрутизація повідомлень, tool use, middleware, омніканальна комунікація, CRM-інтеграція.*