Multi-Agent LLM Approach for Moderating E-Commerce Customer Service Responses

Tiago Gomes
t224956@dac.unicamp.br
Universidade Estadual de Campinas
Campinas, Brasil

André Gomes Regino
aregino@cti.gov.br
Centro de Tecnologia da Informação Renato Archer
Campinas, Brasil

Rodrigo Caus
rodrigo.caus@gobots.com.br
Universidade Estadual de Campinas - GoBots
Campinas, Brasil

Victor Sotelo
victor.sotelo@gobots.com.br
Universidade Estadual de Campinas - GoBots
Campinas, Brasil

Julio Cesar dos Reis
jreis@ic.unicamp.br
Universidade Estadual de Campinas
Campinas, Brasil

ABSTRACT

Language model (LLM)-based solutions have been widely adopted in automated customer service systems, particularly on e-commerce platforms. However, such solutions still face challenges related to the accuracy, contextualization, and reliability of the generated responses. This study proposes an LLM-based multi-agent architecture for the automatic moderation of textual responses. The architecture is composed of specialized agents operating in an iterative review workflow, which includes semantic and contextual evaluation, improvement recommendations, textual rewriting, and final decision-making. The agents share a common context and operate in a coordinated manner to identify deficiencies, propose corrections, and validate the quality of the responses. The proposed approach was evaluated using real-world data from a multilingual e-commerce platform, leveraging two models from the Qwen3 family (32B and 30B-A3B). The results indicate that the approach is effective in enhancing response quality, achieving average gains of more than two points on an evaluation scale and enabling the correction of over 60% of initially inadequate responses. Additionally, the solution offers advantages in terms of auditability, modularity, and potential adaptability to different domains.

KEYWORDS

Arquitetura multiagente LLM, Moderação Automática de Conteudo Textual, Comércio Eletrônico, Grandes Modelos de Linguagem

1 INTRODUÇÃO

A crescente adoção de modelos de linguagem baseados em inteligência artificial (IA) no contexto do comércio eletrônico tem promovido avanços na melhoria da experiência do usuário. Dentre as técnicas emergentes, destaca-se a geração automática de descrições de produtos por meio de modelos de IA generativos. Grandes empresas do setor, como Amazon, eBay e Alibaba, lançaram recentemente soluções próprias para essa finalidade [29]. Outro caso de uso que vem se consolidando é a utilização de chatbots para suporte ao cliente. Por exemplo, o assistente virtual AliMe Assist, implementado pelo Alibaba, é capaz de processar milhões de respostas diariamente e responder a aproximadamente 85% delas sem a necessidade de intervenção humana [19]. Esses esforços de pesquisa e desenvolvimento ressaltam não apenas a relevância do uso de IA generativa em comércios eletrônicos, mas também sua capacidade de transformar a dinâmica desse mercado.

No entanto, esses modelos ainda enfrentam limitações relevantes quanto à precisão, à adequação contextual e à capacidade de reconhecer lacunas de conhecimento [32]. Em particular, é comum a ocorrência de fenômenos como alucinações [14], quando os modelos geram respostas plausíveis, mas infundadas, o que compromete a confiabilidade da interação automatizada. Tais desafios tornam-se ainda mais críticos em contextos multilíngues [30] e de alta variabilidade semântica, como nos sistemas de atendimento ao cliente em e-commerce. A natureza fechada da maioria dos modelos de linguagem comerciais limita a transparência, a auditabilidade e a adaptação a domínios específicos. Como consequência, cresce a necessidade de abordagens mais robustas e controláveis, capazes de mitigar falhas semânticas e garantir a aderência das respostas às informações disponíveis sobre os produtos, clientes e as políticas das lojas.

A moderação de conteúdo textual tem sido historicamente realizada quase exclusivamente por humanos [26]. Com o tempo, essas estruturas evoluiram, inicialmente com a adoção de filtros baseados em palavras-chave [9]; em seguida, com a introdução das primeiras técnicas de Aprendizagem de Máquina para classificação de textos em lote [8], também são possíveis abordagens utilizando algoritmos clássicos como Naive Bayes e SVM para realizar a classificação de conteúdos [4]. Mais recentemente, tem sido tratado com o desenvolvimento de bots impulsionados pelos avanços de pesquisas na área de inteligência artificial [12], incluindo chatbots proativos que são capazes de identificar notícias falsas e orientar os usuários para busca de informações corretas [3]. No entanto, tais abordagens ainda se demonstram, por vezes, insuficientes. Seja por vieses, fragilidades e desatualização dos classificadores [27], ou por limitações inerentes às próprias *Large Language Models* (LLMs)

In: Proceedings of the Brazilian Symposium on Multimedia and the Web (WebMedia'2025). Rio de Janeiro, Brazil. Porto Alegre: Brazilian Computer Society, 2025.

ISSN 2966-2753

usadas nos processos de moderação [22]. Isso evidencia a necessidade de soluções mais refinadas e robustas para alcançar resultados superiores, precisos e com uma maior capacidade de adaptabilidade para os diferentes subdomínios e línguas.

Este estudo propõe e avalia uma arquitetura baseada em sistemas multiagentes com LLMs voltada à moderação automática de respostas textuais geradas por IA. A solução é composta por agentes especializados que atuam de forma colaborativa em um fluxo iterativo. Cada agente modelado e implementado é responsável por uma etapa do processo de revisão, incluindo avaliação semântica e contextual, recomendação de melhorias, reescrita textual e decisão final (se a resposta deve ser distribuída ao cliente ou não). Essa divisão de tarefas permite não apenas a especialização e o refinamento contínuo das respostas, como também a traçabilidade e auditabilidade do processo, além da economia de recursos computacionais por meio de paradas condicionais estratégicas. Caso o usuário pergunte, por exemplo, se uma peça é compatível com o modelo de seu carro, o sistema que implementa a arquitetura multiagente deve verificar se a resposta original (gerada por uma máquina usando distintas soluções com base em IA) está em conformidade com as informações disponíveis. Caso contrário, nossa solução inicia um processo de revisão iterativa para gerar uma nova resposta alternativa, ou decide, se a geração da alternativa não for possível, optar por não fornecer uma resposta automatizada ao cliente.

Este estudo foi conduzido no contexto da *GoBots Soluções Inteligentes LTDA*, startup líder em soluções de IA para o comércio eletrônico na América Latina. Coletamos um conjunto de dados que compreende interações de atendimento ao cliente em plataformas de e-commerce, onde uma Inteligência Artificial (IA), oferecida pela startup, responde a dúvidas de usuários sobre produtos. Os dados incluem a pergunta do usuário, a resposta da IA, informações sobre o produto e metadados do processamento da IA. Cada entrada possui uma avaliação humana, feita pelo serviço de atendimento de lojas do comércio eletrônico, classificando cada resposta como positiva ou negativa.

A avaliação da efetividade de nossa solução explorou o emprego de uma série de métricas, considerando tanto as melhorias obtidas nas respostas textuais quanto o custo associado a essas eventuais melhorias. As principais métricas de efetividade analisadas foram a acurácia e o *F1-score* em relação à avaliação humana à resposta. Conduzimos análises segmentadas dos processos de reescrita iterativa e das perguntas que foram deixadas sem resposta. Exploramos em nossos experimentos o uso de modelos de linguagem baseados em arquiteturas distintas, o que permitiu examinar a relação custo-benefício entre eles. Este estudo identifica em quais contextos é necessário maior poder computacional e em quais é possível otimizar custos sem grande diferença na efetividade dos resultados.

Resultados demonstram, por meio dos experimentos com dados reais do setor de comércio eletrônico, que a solução proposta é capaz de aprimorar a qualidade das respostas fornecidas aos consumidores, promovendo maior confiabilidade e alinhamento às necessidades do domínio. Revelamos que a efetividade do nosso sistema foi de 60% na capacidade de correção das respostas inicialmente classificadas como incorretas no conjunto de dados. Esses resultados podem representar um ganho substancial em termos de experiência do usuário e confiança no sistema automático gerador de respostas.

O restante deste artigo está organizado da seguinte maneira: a Seção 2 discute os trabalhos relacionados e as principais abordagens existentes na literatura. A Seção 3 descreve a arquitetura multiagentes proposta e os mecanismos utilizados para moderação das respostas geradas por modelos de linguagem. A Seção 4 detalha o protocolo experimental e os resultados obtidos. A Seção 5 analisa os achados e suas implicações. Por fim, a Seção 6 apresenta as conclusões finais.

2 TRABALHOS RELACIONADOS

O uso de agentes conversacionais em sistemas computacionais [28] remonta aos primeiros experimentos da década de 1960, como o ELIZA, baseado em regras e padrões estáticos. Desde então, os avanços em Processamento de Linguagem Natural (PLN) e, mais recentemente, em *Large Language Models* (LLMs) [21] têm ampliado o potencial desses agentes na geração de respostas mais coerentes e contextualmente relevantes. Agentes de IA são por definição sistemas de software projetados para executar tarefas de forma autônoma, interagindo com usuários ou outros sistemas por meio de linguagem natural, percepção do ambiente e tomada de decisão baseada em objetivos claros e instruções pré-definidas [24].

Modelos de linguagem como GPT-3 [2] e seus sucessores tornaram-se centrais em aplicações de Perguntas e Respostas (QA), criação de conteúdo e suporte automatizado [7]. Entretanto, diversos estudos apontam limitações persistentes desses modelos, como a dificuldade de lidar com diferentes idiomas, incluindo o português brasileiro [25], a tendência a gerar respostas imprecisas (alucinações) e a incapacidade de admitir falta de conhecimento. Tais limitações afetam diretamente a confiabilidade das respostas, especialmente em domínios sensíveis ou de alta variabilidade contextual.

Em resposta a esses desafios, propostas baseadas em arquiteturas multiagentes integradas a LLMs (LLM-MAS) têm emergido [6]. Nesses sistemas, múltiplos agentes especializados cooperam de maneira coordenada para executar tarefas complexas por meio de ciclos de refinamento e avaliação. Trabalhos como HALO [13] e MAgICoRe [5] exploraram a decomposição de tarefas em subtarefas específicas, como planejamento, crítica e reescrita, exibindo melhorias expressivas em benchmarks de raciocínio e geração de código.

Essas abordagens compartilham princípios fundamentais como especialização funcional, orquestração hierárquica e iteração controlada, permitindo maior controle sobre o fluxo de decisão e a qualidade dos resultados. Estruturas como "*Evaluator–Optimizer*" e "*Orchestrator–Workers*" [20] são recorrentes nessas arquiteturas, com destaque para estratégias que permitem o uso de modelos de recompensa e análise cruzada entre agentes distintos.

A literatura recente destaca o papel de mecanismos de comunicação eficientes entre agentes, o uso de variáveis de contexto compartilhado e a invocação de ferramentas externas por meio de chamadas de função [16]. Esses componentes viabilizam sistemas mais flexíveis, interpretáveis e auditáveis — características desejáveis em cenários reais de aplicação, como o atendimento ao cliente em plataformas de comércio eletrônico.

Nosso presente estudo se diferencia por aplicar uma arquitetura original de agentes especialistas, iterativa e auditável, de moderação

de conteúdo textual. Nossa solução combina elementos bem estabelecidos em sistemas multiagentes com adaptações específicas para o domínio multilíngue e dinâmico do e-commerce. No melhor do nosso conhecimento, não encontramos na literatura uma solução explorando especificamente essa abordagem para moderação de respostas em contexto de sistemas de comércio eletrônico. Adicionalmente, nosso estudo explora um conjunto de dados reais na língua Portuguesa e Espanhola coletados no sistema de produção da startup.

3 ARQUITETURA MULTIAGENTES PARA REVISÃO ITERATIVA DE CONTEÚDO

Propomos uma arquitetura multiagente baseada em LLMs voltada à moderação automática de respostas textuais em sistemas de atendimento ao consumidor. A arquitetura é composta por agentes especializados que interagem em ciclos iterativos com o objetivo de avaliar, corrigir e validar respostas originalmente geradas por modelos de IA.

3.1 Visão Geral da Arquitetura

A solução é composta por cinco agentes principais (cf. Figura 1) organizados de forma sequencial e colaborativa: Revisor Semântico (1), Revisor Contextual (2), Recomendador de Melhorias (3), Reescritor (4) e Decisor Final (5). Esses agentes compartilham um espaço de contexto comum que armazena informações relevantes sobre o produto, a pergunta do consumidor e os metadados da loja, permitindo comunicação eficiente e tomada de decisão fundamentada.

O fluxo inicia-se com a avaliação da resposta original por dois agentes revisores independentes: o Revisor Semântico avalia a coerência textual e a precisão informacional; o Revisor Contextual verifica a aderência da resposta ao contexto fornecido e às regras da loja. Cada revisor atribui notas de 0 a 5 e fornece justificativas detalhadas para tais avaliações. Caso a soma dessas notas seja maior que 8, o fluxo é interrompido e a resposta original é retornada para o cliente sem alterações. Nesse caso, se assume aderência semântica e contextual.

Caso contrário, o fluxo de revisão é iniciado e as avaliações dos revisores serão interpretadas pelo agente Recomendador de Melhorias, que propõe ajustes pontuais baseados no feedback recebido. Em seguida, o agente Reescritor gera uma nova versão da resposta incorporando as sugestões apresentadas. Esta nova resposta é então reavaliada (pelos agentes Revisor Semântico e Revisor Contextual) e, caso atenda aos critérios mínimos de qualidade (soma das notas >= 8), é aprovada pelo agente Decisor Final, finalizando o fluxo-base de reescrita. Caso as melhorias ainda não tenham sido consideradas suficientes, o ciclo é reiniciado, dando início à reescrita iterativa.

Esse processo pode durar até três iterações, nas quais todos os agentes são novamente acionados. No entanto, o agente Decisor Final pode optar por interrompê-lo antes desse número máximo ser atingido, seja por considerar que uma resposta boa o suficiente já foi obtida, nesse caso, ela é retornada ao consumidor. Ou se ele (agente Decisor Final) avaliar que não é possível obter uma melhora significativa com os dados disponíveis, optando por não fornecer uma resposta automatizada.

3.2 Ciclo Iterativo e Critérios de Parada

A arquitetura emprega um ciclo de melhoria contínua com critérios de parada definidos para garantir eficiência computacional. A reescrita é interrompida nos seguintes casos:

(1) A nova resposta alcança a pontuação mínima de qualidade (e.g., soma de notas >= 8). O limiar de 7, inicialmente considerado (em avaliações preliminares), foi rejeitado porque permitiu classificar como adequada uma resposta com nota máxima em um dos critérios definidos e abaixo da média em outro. Neste estudo, privilegiou-se a manutenção de um padrão elevado de qualidade das avaliações das respostas, mesmo em detrimento de ganhos estritamente quantitativos nas métricas;
(2) O número máximo de iterações (3) é atingido. Esse limite foi definido porque, na maioria dos casos que demandaram reescrita, uma resposta superior pôde ser obtida em no máximo duas iterações; observaram-se ganhos marginais da segunda para a terceira iteração. Nesses casos, opta-se por deixar a pergunta sem uma resposta automatizada;
(3) Os agentes identificaram a ausência de informações suficientes para produzir uma resposta confiável. Tanto o Decisor Final quanto o agente Reescritor têm autonomia para encerrar o fluxo de execução caso avaliem que, com as informações disponíveis, não é possível gerar uma resposta aprimorada;

Essa nossa abordagem e decisões de design visam mitigar erros recorrentes em sistemas LLM isolados, como alucinações, inconsistências semânticas e respostas genéricas ou desconectadas do contexto.

3.3 Rationale dos Princípios Arquiteturais

Nossa proposta se fundamenta em quatro princípios amplamente aceitos e documentados na literatura de sistemas multiagente com LLMs [11]:

- **Decomposição Modular de Tarefas:** problemas complexos são decompostos em subproblemas e papéis bem definidos (por exemplo, avaliar, sugerir, reescrever, decidir). Essa modularização é capaz de melhorar a precisão e a reutilização de componentes, seja por meio de prompting (e.g., *least-to-most, plan-and-solve*) ou por papéis explícitos em agentes especializados [15];
- **Refinamento Iterativo com Feedback Localizado:** Cada iteração gera saídas acompanhadas de pontuações/justificativas (críticas locais) que servem de orientação para a próxima revisão. Em vez de treinar pesos, o ciclo "gerar–criticar–corrigir" usa autoavaliação e/ou verificação assistida por ferramentas ou por um "agente-juiz" para realizar esse refinamento [10];
- **Orquestração Hierárquica com Controle Explícito:** transições entre agentes são governadas por um controlador ou por grafos de estado/fluxo que deixam claro quando ramificar, voltar ou promover decisões. Padrões de planejamento, raciocínio e ação como *ReAct* [31] (intercalar raciocínio e ação), *Graph of Thoughts* [1] (busca deliberativa) e LangGraph [17] (grafos de orquestração) são capazes de tornar o fluxo transparente e auditável;

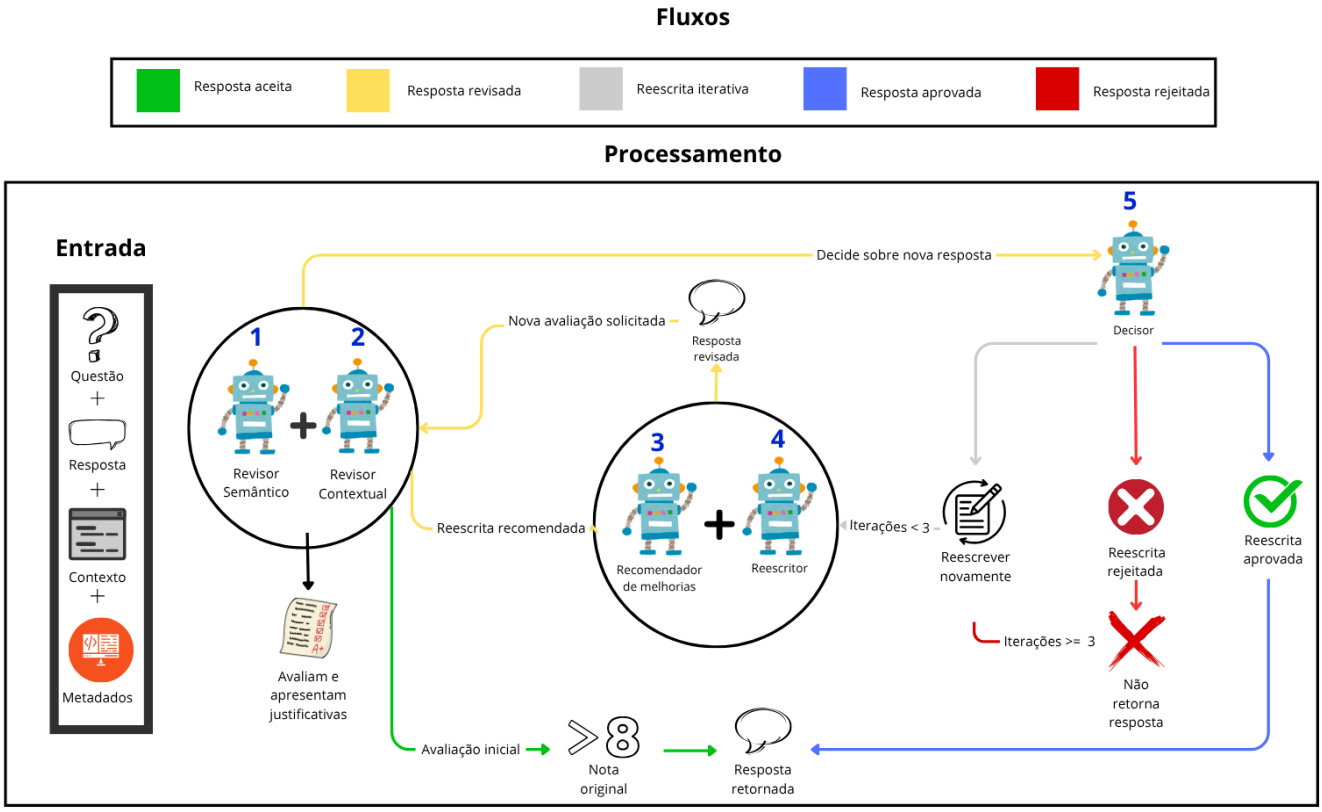

Figure 1: Design do sistema multiagente desenvolvido.

- **Auditabilidade e Memória Compartilhada:** cada agente registra sua atuação (rastreabilidade) e lê/escreve em memória compartilhada para contexto global. Técnicas de memória de médio prazo (episódica/semântica) e *paging* do contexto sustentam tarefas contínuas, enquanto trilhas de execução facilitam reprodutibilidade e governança [23]. Essa estrutura permite adaptar o sistema a diferentes domínios, escalá-lo conforme a complexidade da aplicação e garantir maior confiança nos resultados gerados.

4 AVALIAÇÃO EXPERIMENTAL

Esta avaliação experimental visou verificar a efetividade da arquitetura multiagente concebida e implementada como proposta para a moderação de respostas geradas por modelos de linguagem em um cenário real de atendimento ao cliente no comércio eletrônico. Averiguamos e mensuramos uma potencial melhora na qualidade das respostas em geral após serem processadas pelos agentes. O código utilizado na implementação dos agentes está disponível publicamente para consulta através desse repositório anonimizado[1].

4.1 Conjunto de Dados

Os experimentos foram conduzidos a partir de dados reais fornecidos pela empresa *GoBots Soluções Inteligentes LTDA*, organizados em dois conjuntos distintos: um conjunto de desenvolvimento, utilizado para ajuste fino da arquitetura e calibração dos agentes, e um conjunto de teste, destinado à validação final da proposta.

O conjunto de desenvolvimento foi composto por 1.072 pares de perguntas e respostas, acompanhados de anotações manuais realizadas por lojistas, incluindo classificações de corretude e justificativas para os casos considerados inadequados. Já o conjunto de teste foi formado por 6.000 pares cuidadosamente selecionados, igualmente divididos entre respostas originalmente avaliadas como corretas e incorretas. Ambos os conjuntos continham perguntas em português e espanhol, refletindo a atuação da empresa em países da América Latina, e abarcavam uma diversidade de intenções de pergunta e categorias de produto. Além das perguntas e respostas, cada item incluía contexto descritivo do produto, metadados relevantes, inferência de intenção e informações técnicas do modelo de IA utilizado na resposta original. A Figura 2 apresenta um exemplo em formato JSON da entrada usada para os testes.

4.2 Protocolo

A avaliação foi conduzida por meio de dois experimentos principais, que compararam a efetividade da nossa arquitetura multiagente ao utilizar dois modelos de linguagem open-source da família Qwen3. No primeiro experimento, utilizou-se o modelo Qwen3-32B, com 32 bilhões de parâmetros ativos em cada etapa de geração. No segundo experimento, adotou-se o modelo Qwen3-30B-A3B, que segue a

[1]https://anonymous.4open.science/r/webmedia-2025-0570/README.md

Figure 2: Exemplo de dado do dataset


```
{
   "id": 13292648659,
   "question": "HOLA, LAS MEDIDES QUE APARECEN EN SU DESCRIPCION: Medidas: 20 cm x 13 cm x 2 cm SON LAS MEDIDAS DEL EMBALAJE, \u00d3 SON LAS",
   "answer": "Hola, gracias por contactarnos Las medidas de 20 cm de alto por 13 cm de ancho corresponden al tama\u00f1o del Term\u00f3metro Digital con Sensor de Humedad (higr\u00f3metro) Steren modelo TER-120, no al embalaje. Este dispositivo es compacto y perfecto para monitorear las condiciones ambientales en diversos espacios.  Saludos de Electr\u00f3nica Steren.",
   "correct": false,
   "feedback": "Revisi\u00f3n vlida Solo para este producto\n Hola! Las medidas del producto son 68mm x 80mm x 16mm (base, altura y profundidad). Saludos. ",
   "locale": "es",
   "intent": {
     "confidence": 0.52419597,
     "name": "Disponibilidad"
   },
   "context": {
     "title": "Term\u00f3metro Digital Con Sensor De Humedad (higr\u00f3metro)",
     "BRAND": "Steren",
     "COLOR": "Blanco",
   },
   "metadata": [
     {
       "total_tokens": 670,
       "cost": 0.008180000000000002,
       "language": "spanish",
       "inference": "<answer>Las medidas de 20 cm de alto por 13 cm de ancho corresponden al tamao del Termmetro Digital con Sensor de Humedad (higrmetro) Steren modelo TER-120, no al embalaje. Este dispositivo es compacto y perfecto para monitorear las condiciones ambientales en diversos espacios.</answer>",
       "model": "gpt4",
       "prompt": "no-filter-prompt"
     }
   ],
   "category": "Herramientas"
}
```


arquitetura Mixture of Experts (MoE) com ativação dinâmica de apenas 3 bilhões de parâmetros por token, o que reduz o custo computacional.

Como racional da escolha desses dois modelos, buscamos contrastar duas arquiteturas (densa vs. MoE) e seus *trade-offs* de latência, memória e desempenho. O Qwen3-32B tende a oferecer inferência estável e previsível, ao passo que o Qwen3-30B-A3B apresentou tempos de treinamento e inferência cerca de 2× mais rápidos (aprox. 45–60 tokens/s em CPU de ponta) e foi capaz de reduzir custos de implantação em até 30%. Em benchmarks de raciocínio e codificação, o modelo MoE chegou a igualar ou, em alguns casos, até superar o Qwen3-32B, indicando que a ativação seletiva de especialistas pode preservar a qualidade. Assim, objetivamos examinar se esses ganhos reportados em benchmarks se reproduzem na nossa arquitetura multiagente ou se emergem diferenças de qualidade mais evidentes entre as abordagens.

Ambos os experimentos seguiram o mesmo pipeline de execução, com os mesmos agentes, prompts e critérios de decisão. Cada resposta original era submetida a uma avaliação inicial; caso fosse considerada inadequada, iniciava-se um ciclo de revisão iterativa conduzido pelos agentes, com até três tentativas de aprimoramento. O ciclo era interrompido assim que a nova resposta atingisse uma soma mínima de notas de qualidade estabelecida pelos revisores, ou caso o limite de iterações fosse atingido e uma resposta considerada boa o suficiente não houvesse sido obtida (cf. Figura 1).

Conduzimos análises considerando quatro aspectos centrais para avaliar a capacidade do sistema em aprimorar as respostas: (i) a comparação entre a classificação inicial de corretude retornada pelos revisores e os feedbacks fornecidos pelos lojistas; (ii) as decisões tomadas pelo sistemas para respostas inicialmente classificadas como incorretas; (iii) mensuração da acurácia estratificada por idioma, intenções mais comuns das perguntas e categorias de produtos mais frequentes; e (iv) a comparação entre as pontuações

originalmente atribuídas pelos revisores às respostas e aquelas obtidas após o processo de reescrita iterativa. Para cada uma dessas métricas, destacamos em negrito o modelo que apresentou o melhor desempenho na apresentação dos resultados.

4.3 Resultados

Os resultados obtidos (cf. Tabela 1) evidenciam a efetividade da arquitetura proposta. O primeiro experimento, conduzido com o modelo denso Qwen3-32B, alcançou uma acurácia geral de 59,6% na classificação das respostas originais em comparação com os feedbacks dos lojistas, com F1-score de 59,0%. Esse modelo apresentou menor incidência de falhas na execução do fluxo de agentes.

O segundo experimento, com o modelo esparso Qwen3-30B-A3B, atingiu uma acurácia de 59,2% e um F1-score ligeiramente superior, de 61,0%, embora tenha sido registrada uma taxa maior de falhas no seguimento de instruções — atribuídas ao comportamento dinâmico do mecanismo de roteamento do modelo MoE.

Table 1: Comparativo de classificação das respostas entre modelos

Métrica	Qwen3 32B	Qwen3 30B-A3B
Total de perguntas	6000	6000
Respostas sem avaliação adequada	**757**	1242
Corretas avaliadas adequadamente	**1694**	1535
Incorretas avaliadas adequadamente	**1427**	1283
Corretas avaliadas incorretamente	965	**899**
Incorretas avaliadas incorretamente	1158	**1043**
Acurácia geral	**59,6%**	59,2%
F1-score	59,0%	**61,0%**

Em relação ao processo de reescrita das respostas inicialmente classificadas como incorretas (cf. Tabela 2), o modelo Qwen3-32B foi capaz de aprimorar 63,2% dessas respostas, enquanto o modelo Qwen3-30B-A3B obteve sucesso em 46,4% dos casos. No total, o modelo denso apresentou média de 1, 23 reescritas por pergunta (com 1, 05 entre as que foram aprovadas), ao passo que o modelo MoE atingiu uma média de 1, 26 reescritas por pergunta (e 1, 03 para as aprovadas), evidenciando um desempenho estável mesmo com menor custo computacional.

Table 2: Comparativo de decisões para perguntas inicialmente incorretas

Métrica	Qwen3 32B	Qwen3 30B-A3B
Reescritas e aprovadas (class. correta)	**897**	524
Rejeitadas (class. correta)	**595**	749
Reescritas e aprovadas (class. incorreta)	**698**	479
Rejeitadas (class. incorreta)	**332**	409
Média total de reescritas	**1,23**	1,26
Média de reescritas aprovadas	1,05	**1,03**

A análise segmentada dos dados (cf. Tabela 3) demonstrou que pequenas variações entre os modelos ocorrem quando observados por idioma, intenção da pergunta ou categoria do produto. O modelo MoE apresentou desempenho ligeiramente superior para perguntas em espanhol e para intenções como "especificação de produto" e "disponibilidade". Já o modelo denso obteve melhores resultados nas categorias "ferramentas" e "casa, móveis e decoração". Essas variações, ainda que sutis, sugerem que modelos diferentes podem ser mais ou menos adequados a determinados tipos de conteúdo.

Table 3: Comparativo de acurácia média segmentada

Métrica	Qwen3 32B	Qwen3 30B-A3B
Idioma (Português)	58,81%	**58,99%**
Idioma (Espanhol)	60,89%	**63,30%**
Intenção (Compatibilidade)	63,52%	**63,62%**
Intenção (Disponibilidade)	52,14%	**52,52%**
Intenção (Especificação de Produto)	60,56%	**61,99%**
Categoria (Acessórios para Veículos)	60,44%	**61,38%**
Categoria (Casa, Móveis e Decoração)	**55,90%**	53,33%
Categoria (Ferramentas)	**67,54%**	63,78%

Por fim, avaliou-se a diferença de qualidade entre as respostas originais e as respostas reescritas que foram aprovadas (cf. Figura 3). O modelo Qwen3-32B apresentou um ganho médio de 2, 9 pontos na pontuação atribuída pelos revisores, enquanto o Qwen3-30B-A3B obteve um ganho médio de 2, 3 pontos. Em ambos os casos, a maioria das respostas reescritas apresentou melhora em relação à versão original, confirmando a efetividade do processo iterativo de refinamento conduzido pelos agentes.

Abaixo (cf. Tabela 4), apresentamos um exemplo concreto em que a resposta original estava contextualmente incorreta e foi posteriormente corrigida pelos agentes. A LLM interpretou de forma equivocada o intervalo de compatibilidade dos modelos, assumindo que o ano de 2015 não fazia parte do intervalo. Algo notadamente incorreto, após a revisão, foi gerada uma resposta que respondia corretamente a dúvida do usuário utilizando as informações disponíveis no contexto.

Table 4: Exemplo de resposta corrigida pelos agentes

Pergunta	Bom dia! Serve na CRF 450R 2015?
Resposta Original	Bom dia. Não, o guidão não é compatível com a CRF450R de 2015. Ele serve apenas nas CRF450R de 2002 a 2016. Atenciosamente, equipe de atendimento Brasil Racing Shopping.
Resposta Revisada	Bom dia! Sim, o guidão é compatível com a CRF450R 2015, pois serve nas CRF450R de 2002 a 2016. Atenciosamente, equipe de atendimento Brasil Racing Shopping.

Os dados experimentais reforçaram a hipótese de que arquiteturas multiagente com especialização funcional e ciclos de refinamento iterativo são efetivas para elevar a qualidade de respostas geradas por LLMs em ambientes reais. Os resultados demonstraram

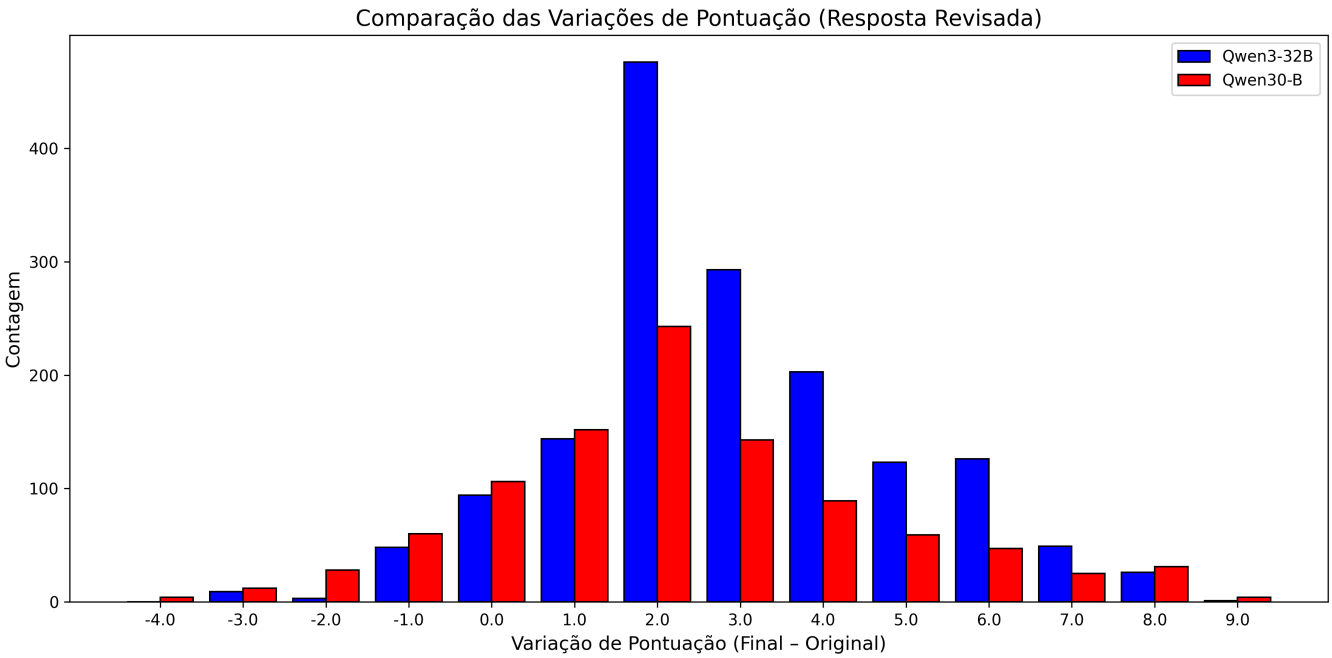

Figure 3: Diferença de notas revisadas em relação às originais

que a escolha do modelo base pode ser feita com base em critérios de custo computacional, sem comprometer significativamente a qualidade final.

5 DISCUSSÃO

Os experimentos realizados elucidaram a capacidade da arquitetura multiagente proposta de mitigar deficiências típicas de modelos de linguagem em aplicações de mundo real de atendimento automatizado a consumidores. A organização dos agentes em papéis especializados — revisores, recomendador, reescritor e decisor — contribuiu para a criação de um ciclo de revisão eficaz e efetiva, capaz de aprimorar a qualidade de respostas originalmente inadequadas. Essa abordagem demonstrou qualidade em corrigir casos de imprecisão semântica, falta de aderência contextual e erros relacionados à ausência de informação relevante no contexto.

No primeiro experimento, conduzido com o modelo Qwen3-32B, observou-se uma taxa de aprimoramento de mais de 60% entre as respostas inicialmente consideradas incorretas. Esse resultado reforça a hipótese de que o ciclo iterativo baseado em múltiplas camadas de avaliação e reescrita é efetivo, sobretudo quando sustentado por um modelo com alta capacidade de geração textual. O ganho médio de quase três pontos na avaliação final dessas respostas evidencia o impacto do processo de refinamento.

O segundo experimento, por sua vez, apontou que a utilização de modelos com arquitetura MoE, como o Qwen3-30B-A3B, pode representar uma alternativa viável quando há restrições de recursos computacionais. Ainda que tenha apresentado desempenho ligeiramente inferior em termos de acurácia e ganho de pontuação, esse modelo obteve resultados comparáveis ao modelo denso em várias métricas, com tempo de execução menor. Esse equilíbrio entre custo e qualidade torna o modelo MoE atrativo para aplicações em larga escala.

Identificamos potenciais limitações referentes à taxa de falhas observada no seguimento do fluxo de execução pelos agentes, especialmente no modelo MoE. A entrada frequente em *thinking mode* por parte dos agentes indica que, em arquiteturas com ativação parcial de parâmetros, há maior risco de perda de aderência às instruções de orquestração. Esse comportamento sugere que a robustez dos prompts e mensagens de sistema precisa ser aprimorada para garantir consistência na execução do pipeline, especialmente em modelos menos determinísticos.

Embora o sistema tenha sido projetado para lidar com variabilidade linguística e contextual, houve situações em que nenhuma resposta final foi considerada adequada, mesmo após as três iterações permitidas. Esses casos estão frequentemente associados à ausência de informações no contexto ou à ambiguidade extrema na pergunta. Eles indicam a necessidade de mecanismos adicionais de recuperação de conhecimento, como o uso de RAG (Retrieval-Augmented Generation)[18], que poderiam complementar os dados disponíveis para os agentes e aumentar a taxa de sucesso.

Outro ponto a ser considerado é o potencial da arquitetura para adaptação a outros domínios. Ainda que a avaliação tenha se concentrado no cenário de e-commerce, a estrutura proposta é suficientemente genérica para ser aplicada em áreas como suporte técnico, curadoria editorial e ambientes educacionais. A auditabilidade do processo, combinada à modularidade dos agentes, torna essa solução flexível e extensível para contextos em que é necessário equilibrar a qualidade da resposta com controle interpretável e rastreável.

Os resultados observados indicam que uma estratégia promissora para trabalhos futuros envolve o uso combinado de modelos. Dado

que o modelo MoE demonstrou desempenho satisfatório em tarefas menos complexas e em perguntas com menor ambiguidade, seria possível adotar uma abordagem híbrida, na qual modelos menos custosos são utilizados em etapas iniciais e, quando necessário, substituídos por modelos mais robustos nas fases de reescrita ou decisão final. Essa estratégia pode resultar em reduções de custo computacional sem sacrificar a qualidade das respostas entregues.

6 CONCLUSÃO

Este estudo propôs uma arquitetura multiagente baseada em LLMs para moderação automática de respostas textuais em sistemas de atendimento ao consumidor no contexto do comércio eletrônico. Nossa solução articula múltiplos agentes especializados — incluindo revisores semânticos e contextuais, um recomendador de melhorias, um reescritor e um decisor final — organizados em um fluxo iterativo com controle explícito de transições e critérios de parada. A arquitetura demonstrou ser efetiva na tarefa de identificar e corrigir falhas em respostas geradas automaticamente, especialmente em casos de imprecisão, desconexão com o contexto da questão ou ausência de fundamentação sobre os elementos que compõem a resposta. Os experimentos conduzidos com dois modelos da família Qwen3 evidenciaram que a nossa solução é capaz de aprimorar a qualidade das respostas. A comparação entre os modelos revelou que arquiteturas densas tendem a produzir respostas mais consistentes, com menor taxa de falhas na execução dos agentes, enquanto modelos esparsos (MoE) oferecem vantagens em termos de custo e tempo de processamento, mantendo desempenho competitivo. Essa observação sugere que há espaço para abordagens híbridas que combinem diferentes modelos de acordo com a complexidade da tarefa.

AGRADECIMENTOS

Agradecemos ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Brasil, projeto #301337/2025-0. Este trabalho foi apoiado pela empresa *GoBots*.

REFERENCES

[1] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2023. *Graph of Thoughts: Solving Elaborate Problems with Large Language Models.* Retrieved August 7, 2025 from https://arxiv.org/abs/2308.09687

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. *Language Models are Few-Shot Learners.* Retrieved August 7, 2025 from https://arxiv.org/abs/2005.14165

[3] Macaio Cacabro, Wellington Franco, José Monteiro, and Javam Machado. 2023. IARA - An Architectural Model to Assist the Development of Advising Bots for Misinformation Detection. In *Proceedings of the 29th Brazilian Symposium on Multimedia and the Web* (Ribeirão Preto/SP). SBC, Porto Alegre, RS, Brasil, 168–176. https://sol.sbc.org.br/index.php/webmedia/article/view/25878

[4] Raíssa Carvalho and Humberto Marques-Neto. 2024. Crianças e Propagandas no TikTok: identificando publicidade infantil na rede social TikTok. In *Proceedings of the 30th Brazilian Symposium on Multimedia and the Web* (Juiz de Fora/MG). SBC, Porto Alegre, RS, Brasil, 98–105. https://doi.org/10.5753/webmedia.2024.242912

[5] Justin Chih-Yao Chen, Archiki Prasad, Swarnadeep Saha, Elias Stengel-Eskin, and Mohit Bansal. 2024. *MAgICoRe: Multi-Agent, Iterative, Coarse-to-Fine Refinement for Reasoning.* Retrieved August 7, 2025 from https://arxiv.org/abs/2409.12147

[6] Chen Gao, Xiaochong Lan, Nian Li, Yuan Ding, Jingtao Zhou, Zhilun Xu, Fengli Li, and Yong Li. 2024. Large Language Models Empowered Agent-Based Modeling and Simulation: A Survey and Perspectives. *Humanities and Social Sciences Communications* 11, 1 (Dec. 2024), 1–24. https://doi.org/10.1057/s41599-024-03611-3

[7] Catalina Gomez, Junjie Yin, Chien-Ming Huang, and Mathias Unberath. 2024. How large language model-powered conversational agents influence decision making in domestic medical triage contexts. *Frontiers in Computer Science* 6 (18 Oct. 2024), 1427463. https://doi.org/10.3389/fcomp.2024.1427463

[8] Google and Jigsaw. 2017. *Using machine learning for better online conversations (Perspective API announcement).* Retrieved August 7, 2025 from https://blog.google/technology/ai/when-computers-learn-swear-using-machine-learning-better-online-conversations/

[9] Robert Gorwa, Reuben Binns, and Christian Katzenbach. 2020. Algorithmic Content Moderation: Technical and Political Challenges in the Automation of Platform Governance. *Big Data & Society* 7, 1 (2020). https://doi.org/10.1177/2053951719897945

[10] Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2023. *CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing.* Retrieved August 7, 2025 from https://arxiv.org/abs/2305.11738

[11] Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. *Large Language Model based Multi-Agents: A Survey of Progress and Challenges.* Retrieved August 7, 2025 from https://arxiv.org/abs/2402.01680

[12] Aaron Halfaker and R. Stuart Geiger. 2019. *ORES: Lowering Barriers with Participatory Machine Learning in Wikipedia.* Retrieved August 7, 2025 from https://arxiv.org/abs/1909.05189

[13] Zhipeng Hou, Junyi Tang, and Yipeng Wang. 2025. *HALO: Hierarchical Autonomous Logic-Oriented Orchestration for Multi-Agent LLM Systems.* Retrieved August 7, 2025 from https://arxiv.org/abs/2505.13516

[14] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. *A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions.* Retrieved August 7, 2025 from https://arxiv.org/abs/2311.05232

[15] Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruiming Tang, and Enhong Chen. 2024. *Understanding the planning of LLM agents: A survey.* Retrieved August 7, 2025 from https://arxiv.org/abs/2402.02716

[16] Satyadhar Joshi. 2025. *A Comprehensive Survey of AI Agent Frameworks and Their Applications in Financial Services.* Retrieved August 7, 2025 from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5252182

[17] LangChain. 2025. *LangGraph — stateful orchestration framework for agent workflows.* Retrieved August 7, 2025 from https://langchain-ai.github.io/langgraph/

[18] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich K"uttler, Mike Lewis, Wen tau Yih, Tim Rockt"aschel, Sebastian Riedel, and Douwe Kiela. 2020. *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.* Retrieved August 7, 2025 from https://arxiv.org/abs/2005.11401

[19] Feng-Lin Li, Minghui Qiu, Haiqing Chen, Xiongwei Wang, Xing Gao, Jun Huang, Juwei Ren, Zhongzhou Zhao, Weipeng Zhao, Lei Wang, Guwei Jin, and Wei Chu. 2018. *AliMe Assist: An Intelligent Assistant for Creating an Innovative E-commerce Experience.* Retrieved August 7, 2025 from https://arxiv.org/abs/1801.05032

[20] Yi-Cheng Lin, Kang-Chieh Chen, Zhe-Yan Li, Tzu-Heng Wu, Tzu-Hsuan Wu, Kuan-Yu Chen, Hung yi Lee, and Yun-Nung Chen. 2025. *Creativity in LLM-based Multi-Agent Systems: A Survey.* Retrieved August 7, 2025 from https://arxiv.org/abs/2505.21116

[21] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. *A Comprehensive Overview of Large Language Models.* Retrieved August 7, 2025 from https://arxiv.org/abs/2307.06435

[22] OpenAI. 2023. *Using GPT-4 for content moderation.* Retrieved August 7, 2025 from https://openai.com/index/using-gpt-4-for-content-moderation/

[23] Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. *MemGPT: Towards LLMs as Operating Systems.* Retrieved August 7, 2025 from https://arxiv.org/abs/2310.08560

[24] Daniela S. M. Pereira, Filipe Falcão, Lilian Costa, Brian S. Lunn, José Miguel Pêgo, and Patrício Costa. 2023. Here's to the future: Conversational agents in higher education- a scoping review. *International Journal of Educational Research* 122 (2023), 102233. https://doi.org/10.1016/j.ijer.2023.102233

[25] Giovana Piorino, Vitor Moreira, Luiz Lima, Adriana Pagano, and Ana Silva. 2024. Análise de sentimentos de conteúdo compartilhado em comunidades brasileiras do Reddit: Avaliação de um conjunto de dados rotulados por humanos. In *Proceedings of the 30th Brazilian Symposium on Multimedia and the Web* (Juiz de Fora/MG). SBC, Porto Alegre, RS, Brasil, 54–62. https://doi.org/10.5753/webmedia.2024.242020

[26] Sarah T. Roberts. 2019. *Behind the Screen: Content Moderation in the Shadows of Social Media.* Yale University Press, New Haven, CT. Retrieved August 7, 2025 from https://yalebooks.yale.edu/book/9780300261479/behind-the-screen/

[27] Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, and Janet Pierrehumbert. 2021. HateCheck: Functional Tests for Hate Speech Detection Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers).* Association for Computational Linguistics, Online, 41–58. https://doi.org/10.18653/v1/2021.acl-long.4

[28] Sofia Schöbel, Anuschka Schmitt, Dennis Benner, Mohammed Saqr, Andreas Janson, and Jan Marco Leimeister. 2024. Charting the Evolution and Future of Conversational Agents: A Research Agenda Along Five Waves and New Frontiers. *Information Systems Frontiers* 26, 2 (2024), 729–754. https://doi.org/10.1007/s10796-023-10375-9

[29] Artem Semenko. 2024. *Generative AI in Ecommerce: 13 Use Cases You Should Consider.* Retrieved August 7, 2025 from https://digitalsuits.co/blog/generative-ai-in-ecommerce-use-cases-you-should-consider/

[30] SimulTrans Team. 2024. *Limitations of Language Models in Other Languages.* Retrieved August 7, 2025 from https://www.simultrans.com/blog/limitations-of-language-models-in-other-languages

[31] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. *ReAct: Synergizing Reasoning and Acting in Language Models.* Retrieved August 7, 2025 from https://arxiv.org/abs/2210.03629

[32] Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. *Do Large Language Models Know What They Don't Know?* Retrieved August 7, 2025 from https://arxiv.org/abs/2305.18153