Arquitetura Multiagente para Análise de Risco em Contextos Complexos: Um Estudo de Caso em Violência Doméstica

Cairo Gabriel Castadini Cruz[1], **Paulo Henrique Ribeiro Gabriel**[1]

[1]Faculdade de Computação – Universidade Federal de Uberlândia (UFU)
Av. João Naves de Ávila, 2121 – 38.400-902 – Uberlândia – MG – Brasil

cairo.cruz@ufu.br, phrg@ufu.br

Abstract. *Risk analysis in sensitive domains, such as domestic violence, demands precision, traceability, and grounding in documented precedents. This work proposes a multi-agent architecture with specialization by socio-psychological dimension, hierarchical supervision, and self-correction, aimed at producing auditable assessments. Validation across three synthetic scenarios parameterized by Brazilian legal instruments demonstrated correct severity differentiation, recommendations adapted to urgency, and full traceability, with no false positives. The exclusive use of synthetic data delimits the scope of the prototype without invalidating the central contribution: a replicable paradigm for multidimensional risk analysis in contexts that demand broad analytical coverage and an auditable justification of the decision-making process.*

Resumo. *A análise de risco em domínios sensíveis, como a violência doméstica, exige precisão, rastreabilidade e ancoragem em precedentes documentados. Este trabalho propõe uma arquitetura multiagente com especialização por dimensão sociopsicológica, supervisão hierárquica e autocorreção, voltada à produção de avaliações auditáveis. A validação em três cenários sintéticos parametrizados por instrumentos jurídicos brasileiros demonstrou diferenciação correta de severidade, recomendações adaptadas à urgência e rastreabilidade integral, sem falsos positivos. O uso exclusivo de dados sintéticos delimita o escopo do protótipo sem invalidar a contribuição central: um paradigma replicável para análise de risco multidimensional em contextos que demandem cobertura analítica ampla e justificativa auditável do processo decisório.*

1. Introdução

A avaliação de risco em cenários multifacetados, como os contextos de violência doméstica, impõe desafios significativos aos sistemas de apoio à decisão. Abordagens tradicionais esbarram na racionalidade limitada do analista humano e na suscetibilidade a vieses de julgamento [Simon 1955], restringindo a capacidade de correlacionar simultaneamente grandes volumes de variáveis contextuais. Nesse cenário, a adoção de Modelos de Linguagem de Grande Escala (LLMs) em arquiteturas de Inteligência Artificial Distribuída [Wooldridge 2009] surge como alternativa viável para modelar riscos complexos em domínios de alta sensibilidade [Goldstick and Jay 2022].

Entretanto, a mera substituição do analista humano por um LLM monolítico reproduz, em escala computacional, dois problemas crônicos: a produção de informações factualmente incorretas (*alucinações*) e a opacidade do processo decisório. O primeiro é

mitigado pela técnica de *Retrieval-Augmented Generation* (RAG) [Lewis et al. 2020], que ancora as respostas do modelo em precedentes documentados. O segundo exige mecanismos de governança que tornem o raciocínio auditável, tais como supervisão hierárquica entre agentes [Stone and Veloso 2000, Bernath 2025].

Este artigo apresenta uma arquitetura de Sistemas Multiagente (MAS) que integra decomposição especializada, RAG e supervisão hierárquica com laço de autocorreção. A solução, orquestrada pelo framework CrewAI [CrewAI 2024], é validada em um estudo de caso de análise de risco de violência doméstica parametrizado pela Lei Maria da Penha [Brasil 2006] e pelo Formulário Nacional de Avaliação de Risco (FONAR) [Conselho Nacional de Justiça 2020]. A contribuição central não reside no sistema instanciado, mas no paradigma arquitetural: a combinação de (i) cinco agentes especialistas cobrindo dimensões sociopsicológicas distintas, (ii) ancoragem obrigatória em contexto histórico via RAG e (iii) supervisão hierárquica com retrabalho condicional é apresentada como um modelo aplicável a domínios nos quais a rastreabilidade do processo decisório é requisito não funcional inegociável. As seções subsequentes detalham a fundamentação teórica, a arquitetura proposta, a metodologia de validação e os resultados obtidos em três cenários sintéticos.

2. Referencial Teórico

Este trabalho apoia-se em três fundamentos interdependentes: (i) Sistemas Multiagente justificam a divisão do problema analítico; (ii) RAG ancora as inferências em precedentes documentados; e (iii) a orquestração hierárquica sustenta o controle de qualidade do processo decisório.

Sistemas Multiagente para Análise de Risco. Um agente é uma entidade computacional capaz de perceber seu ambiente e agir sobre ele de forma autônoma [Wooldridge 2009]. Em cenários complexos, a abordagem MAS permite a decomposição de um problema macro em subproblemas menores, simulando a colaboração de uma equipe multidisciplinar. O uso de MAS para avaliação de risco é consolidado em domínios financeiros e ambientais [Hassan et al. 2018], mas sua integração com LLMs constitui uma evolução recente. No contexto específico da violência, [Goldstick and Jay 2022] argumentam que a modelagem baseada em agentes é subutilizada. A arquitetura proposta neste trabalho adapta o paradigma mediante a criação de agentes com domínios de expertise distintos (rotina, tom emocional, redes de apoio, controle financeiro e bem-estar), reduzindo vieses oriundos de avaliações monolíticas [Stanley and Humphreys 2014].

***Retrieval-Augmented Generation*.** LLMs operam sobre o conhecimento adquirido em treinamento, o que favorece alucinações e desatualização. O RAG mitiga essas limitações ao recuperar documentos relevantes de uma base externa antes de gerar a resposta [Lewis et al. 2020]. O trabalho de [Nanua et al. 2025] evidencia a viabilidade do RAG na interpretação de regulamentações clínicas, sugerindo sua aptidão em domínios auditáveis. Neste trabalho, cada agente especialista consulta obrigatoriamente uma base vetorial (Supabase com `pgvector`) antes de produzir seu parecer, aplicando

busca por similaridade de cossenos sobre *embeddings* gerados pelo modelo *paraphrase-multilingual-MiniLM-L12-v2* [Reimers and Gurevych 2019]. Os precedentes recuperados são injetados no *prompt* como uma forma dinâmica de *few-shot learning* em tempo de execução.

Orquestração Hierárquica e Governança. A orquestração hierárquica organiza agentes em níveis distintos de autoridade, com entidades superiores supervisionando subordinadas [Stone and Veloso 2000]. Arquiteturas multiagente baseadas em LLMs sem supervisão centralizada tendem à inconsistência entre agentes [Bernath 2025]. Tal diagnóstico orientou a adoção de um Agente Supervisor de Qualidade, responsável por avaliar criticamente os relatórios dos especialistas e acionar, quando necessário, um *self-correction loop*. O mecanismo é essencial para reduzir erros em domínios críticos [Dhayakar 2025] e amplia a explicabilidade do sistema ao registrar, em logs auditáveis, cada iteração de retrabalho.

3. Arquitetura Proposta

A arquitetura proposta neste trabalho é organizada em torno de três decisões de projeto: (i) cada agente especialista recebe apenas a dimensão de risco que lhe compete, impedindo que aspectos salientes do relato dominem dimensões sutis; (ii) nenhum relatório avança para a síntese sem aprovação do supervisor; e (iii) todas as interações são registradas em log persistente, tornando o processo auditável externamente. O sistema é exposto por uma API RESTful construída em FastAPI, que delega as requisições ao orquestrador CrewAI e persiste metadados de execução no Supabase.

3.1. Módulo RAG e Base de Conhecimento

O módulo de recuperação de contexto é o pilar de ancoragem empírica das análises. Antes de um agente emitir seu parecer, a narrativa da usuária é vetorizada e comparada contra cinco *datasets* domínio-específicos (um por agente especialista, totalizando 75 casos históricos). Os *datasets* foram sintetizados com auxílio de modelo fundacional e parametrizados pelas categorias da Lei Maria da Penha [Brasil 2006] e pelos indicadores do FONAR [Conselho Nacional de Justiça 2020], conforme amostra na Tabela 1. O fluxo completo de vetorização, recuperação e geração contextualizada é ilustrado na Figura 1.

A recuperação é resiliente: opera em três níveis em cascata, cuja lógica é resumida no Código 1. O primeiro nível executa busca vetorial no Supabase com limiar de similaridade de 0,70; o segundo reduz o limiar para maximizar *recall*; o terceiro recorre a arquivos CSV locais caso o serviço remoto esteja indisponível. Essa estratégia preserva o funcionamento do sistema mesmo sob degradação parcial da infraestrutura, atendendo ao requisito de tolerância a falhas. O limiar de 0,70 opera como filtro de alta precisão; quando nenhum caso supera esse valor, o fallback retorna os casos disponíveis da base domínio-específica do agente, garantindo que o contexto injetado seja sempre semanticamente pertinente à dimensão analisada. Reranker foi descartado nesse momento dado o tamanho reduzido da base (75 casos); sua incorporação é direção de trabalho futuro.

3.2. Orquestração Hierárquica e *Self-Correction Loop*

O núcleo do sistema opera em três fases sequenciais, ilustradas na Figura 2. Na Fase 1 (Execução), cinco agentes especialistas analisam, cada um, uma dimensão distinta da

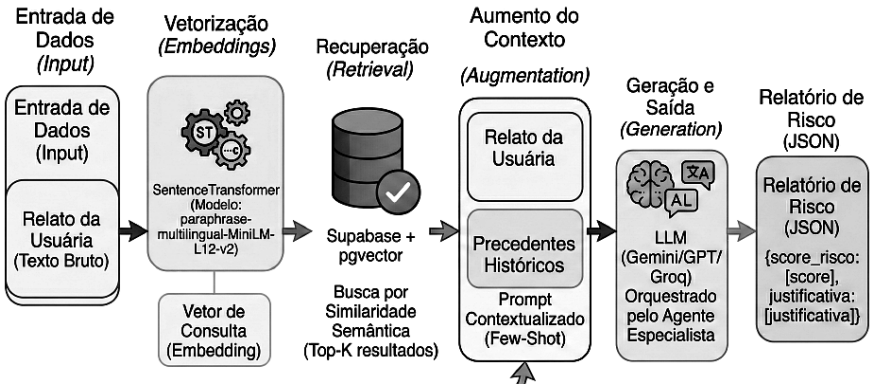

Figura 1. Arquitetura do fluxo RAG: a narrativa da usuária é vetorizada pelo SentenceTransformer, comparada por similaridade contra a base Supabase/`pgvector`, e os *top-K* precedentes recuperados são injetados no *prompt* do LLM como contexto *few-shot* antes da geração do relatório estruturado.

Tabela 1. Amostra da estruturação dos *datasets* sintéticos por dimensão especialista. Cada registro possui relato em primeira pessoa, fator de risco, nível e taxonomia.

Agente	Fator	Relato (amostra)	Nível	Taxonomia
Rotina	Sobrecarga	"Eu faço tudo em casa; ele só chega e senta."	Alto	Desequilíbrio
Emocional	Intimidação	"Ele grita quando algo não sai do jeito dele."	Alto	Agressão psicológica
Redes de apoio	Isolamento	"Ele não gosta que eu saia com minhas amigas."	Alto	Controle de vínculos
Financeiro	Controle	"Preciso pedir para comprar qualquer coisa."	Alto	Dependência econômica
Saúde mental	Esgotamento	"Tenho me sentido muito cansada e sem energia."	Médio	Fadiga crônica

narrativa da usuária. Cada agente é obrigado por *prompt* a invocar a ferramenta RAG antes de emitir parecer; essa restrição, expressa em linguagem imperativa no *backstory* do agente, foi necessária após testes preliminares em que os modelos ignoravam a ferramenta disponível e geravam análises exclusivamente paramétricas. Os cinco domínios e as respectivas perguntas-guia injetadas no *prompt* de cada especialista são apresentados na Tabela 2.

Na Fase 2 (Supervisão), o Agente Supervisor avalia criticamente os cinco relatórios, verificando coerência lógica, fundamentação técnica e uso adequado das ferramentas RAG. Relatórios reprovados são devolvidos ao agente de origem com *feedback* estruturado, respeitando um limite máximo de duas iterações; esse teto foi estabelecido para impedir que ciclos ilimitados comprometessem o tempo de resposta. Na Fase 3 (Síntese), após aprovação unânime, um Agente Sintetizador consolida os pareceres em um documento mestre em formato JSON contendo *risk_score* numérico, *risk_level* categórico, fatores consolidados e recomendações.

Listing 1. Lógica de recuperação RAG com *fallback* em cascata.

```
def _run(self, frase_usuario: str) -> str:
    embedding = self.encoder.encode([frase_usuario]).tolist()[0]
    historico = []
    if self.db.client is not None:
        historico = self.db.find_similar(embedding, self.agent_id,
                                         limit=5, threshold=0.70)
        if not historico:  # reduz exigencia se nada acima do limiar
            historico = self.db.find_similar(embedding, self.agent_id,
                                             limit=5, threshold=-1.0)
    if not historico:      # fallback local em CSV
        historico = _local_find_similar(self.encoder, frase_usuario,
                                        self.agent_id, limit=5)
    return self._format_results(historico)
```

Tabela 2. Mapeamento dos agentes especialistas e perguntas-guia injetadas como *prompt*. A restrição de escopo por agente força a cobertura analítica uniforme das dimensões de risco.

Agente	Dimensão	Pergunta-guia
Especialista 1	Rotina e tarefas	Como é a divisão de tarefas domésticas na sua casa? Você sente que há equilíbrio?
Especialista 2	Tom emocional	Como é a comunicação com seu parceiro? Ele costuma gritar, xingar ou intimidar?
Especialista 3	Redes de apoio	Você tem pessoas de confiança? Seu parceiro interfere nessas relações?
Especialista 4	Controle financeiro	Como funciona a questão financeira? Você tem autonomia sobre o dinheiro?
Especialista 5	Bem-estar e saúde	Como você tem se sentido? Tem sentido medo, ansiedade ou desconforto?

Um subsistema de *tracing* desacoplado intercepta os eventos dos agentes (prompts, saídas intermediárias, chamadas de ferramentas e consumo de tokens) e os persiste assincronamente no Supabase. As gravações são não bloqueantes: falhas de I/O são capturadas silenciosamente para não interromper a inferência principal. A camada adicional de arquitetura LLM-agnóstica (via LiteLLM) permite substituir o provedor de modelo (Gemini, OpenAI, Groq) por configuração de ambiente, evitando *vendor lock-in*.

4. Validação Experimental

A validação da arquitetura foi conduzida por meio de cenários simulados (*vignettes*) com geração de dados sintéticos assistida por IA. A opção por dados sintéticos decorre de duas restrições: a coleta de dados primários com vítimas reais exigiria aprovação ética e conformidade plena com a LGPD, inviável no escopo do estudo; e o foco do trabalho é a validação da arquitetura de software, não a validação clínica de um instrumento psicológico. A parametrização pela Lei Maria da Penha e pelos eixos do FONAR garante que os padrões representados possuam respaldo em instrumentos juridicamente reconhecidos, circunscrevendo o espaço de variação dos relatos a cenários com precedente documentado.

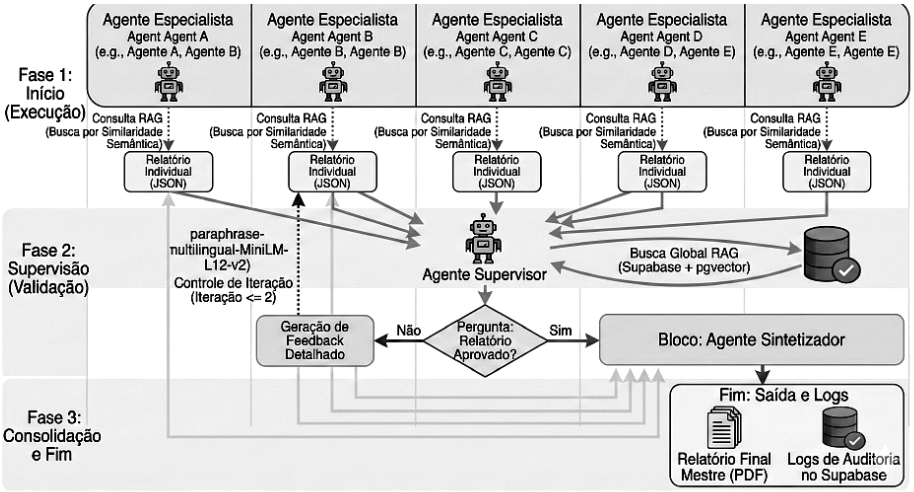

Figura 2. Fluxograma da orquestração hierárquica em três fases: execução paralela dos especialistas com RAG (Fase 1), validação pelo supervisor com *self-correction loop* limitado a duas iterações (Fase 2) e síntese consolidada (Fase 3). Todos os eventos são persistidos no módulo de *tracing*.

Foram desenhados três cenários com graus distintos de risco. O *Cenário 1* (Persona Ana) simula controle coercitivo oculto, com dependência econômica, isolamento social velado e *gaslighting*, visando testar a capacidade do sistema em detectar violências sutis. O *Cenário 2* (Persona Maria) representa risco crítico com violência física, ameaças e confisco de salário, validando a emissão de alertas máximos. O *Cenário 3* (Persona Carla) descreve um relacionamento igualitário e saudável, atuando como caso de controle para estressar o sistema contra falsos positivos, a emissão de alerta em um relacionamento sem indicadores de violência invalidaria os cenários anteriores. Cada cenário foi submetido ao sistema por meio de cinco respostas textuais correspondentes às perguntas-guia da Tabela 2, e o sistema produziu, de forma completamente autônoma, o veredito consolidado final.

5. Resultados e Discussão

A Tabela 3 sintetiza os resultados obtidos nos três cenários, destacando o escore numérico, o nível categórico, o comportamento do *self-correction loop* e a natureza das recomendações geradas pelo Agente Sintetizador.

A análise transversal evidencia três propriedades arquiteturais relevantes. A diferenciação de severidade (escores 90, 97 e 4) mostra que a arquitetura distingue coerentemente entre violência oculta, violência explícita e ausência de risco, evitando classificação indevida no caso de controle. A adaptação das recomendações refletiu a diferença entre urgência absoluta (Cenário 2: intervenção imediata) e necessidade de desconstrução gradual (Cenário 1: orientação jurídica). A estabilidade emergente observada, dado que nenhum cenário acionou o *self-correction loop*, sugere que a restrição de escopo por agente combinada à ancoragem obrigatória via RAG já produz relatórios consistentes na primeira passagem, mantendo o mecanismo de retrabalho como salvaguarda para entradas atípicas.

O módulo de *tracing* capturou integralmente o *chain of thought* dos agentes, ma-

Tabela 3. Comparativo dos três cenários de validação. O sistema diferencia corretamente riscos sutis, críticos e nulos, e ajusta o tom das recomendações à severidade detectada. O escore é calculado pelo Agente Sintetizador por raciocínio holístico sobre os escores individuais dos cinco especialistas.

Cenário (Persona)	Escore	Nível	Retrabalho	Natureza das recomendações
1 – Controle coercitivo oculto (Ana)	90	Crítico	0 iterações	Apoio, orientação jurídica, plano de segurança, medida protetiva possível
2 – Risco crítico explícito (Maria)	97	Crítico	0 iterações	Intervenção imediata, acompanhamento psiquiátrico, afastamento do agressor
3 – Relacionamento saudável (Carla)	4	Baixo	0 iterações	Manter comunicação, fortalecer rede de apoio, monitoramento contextual

terializando a explicabilidade exigida em sistemas de IA responsáveis [Dhayakar 2025]. No Cenário 1, o registro evidenciou que o escore 90 decorreu da coerência cruzada entre os cinco especialistas, nenhum produziria esse escore isoladamente, reforçando que a decomposição especializada captura riscos que avaliações monolíticas diluem [Stanley and Humphreys 2014].

Reconhecem-se limitações: a validação sobre *vignettes* sintéticas não substitui ensaios clínicos com dados reais aprovados por comitê de ética; a dependência de provedores externos de LLM introduz risco de conformidade com a LGPD; e a latência do *self-correction loop* pode ser proibitiva em cenários de tempo real. Essas restrições delimitam o escopo do protótipo, sem invalidar a contribuição arquitetural. O limiar de similaridade de 0,70 foi definido empiricamente, buscando recuperar apenas casos com similaridade semântica relevante acima de 0,5; estudos futuros podem avaliar sistematicamente o impacto de diferentes limiares nos resultados.

6. Conclusões

Este artigo apresentou uma arquitetura multiagente que combina decomposição especializada, RAG e supervisão hierárquica com *self-correction loop* para análise de risco em domínios críticos. A validação em três cenários sintéticos de violência doméstica evidenciou que a arquitetura produz vereditos diferenciados em função da severidade do risco, adapta o tom das recomendações à urgência detectada e mantém rastreabilidade integral do processo decisório. A contribuição central não é o sistema instanciado, mas o paradigma arquitetural: os três pilares, especialização restrita, ancoragem empírica obrigatória e governança hierárquica auditável, são aplicáveis a qualquer domínio que demande cobertura multidimensional e justificativa auditável.

Cada frente mapeada como trabalho futuro visa superar uma limitação estrutural identificada: o fine-tuning on-premise elimina o risco de conformidade com a LGPD; o GraphRAG enriquece a recuperação com relações semânticas entre entidades; a governança human-in-the-loop incorpora supervisão especializada ao processo decisório; a ingestão multimodal amplia a cobertura de evidências; e a validação com dados reais consolida a transição do paradigma arquitetural para instrumento clínico.

Declaração de uso de IA

Para a produção deste trabalho, foram utilizados Google Gemini (Google, 2026) para geração e refinamento visual dos diagramas e datasets sintéticos de validação, e Claude Sonnet 4.6 (Anthropic, 2026) para revisão gramatical e reorganização de trechos do texto. Todo conteúdo gerado pelas ferramentas foi revisado criticamente pelos autores, que assumem integral responsabilidade pelo trabalho final.

Referências

Bernath, E. (2025). Dual-mode AI orchestration: Intelligent coordination mode selection for multi-agent systems. *ArXiv Preprint*, pages 1–4.

Brasil (2006). Lei n. 11.340, de 7 de agosto de 2006 – lei maria da penha. Diário Oficial da União, n. 151.

Conselho Nacional de Justiça (2020). Formulário nacional de avaliação de risco – resolução conjunta cnj/cnmp n. 5/2020.

CrewAI (2024). CrewAI: Multi-agent orchestration framework. `https://crewai.com/docs`.

Dhayakar, D. (2025). Multi-agent architecture for enterprise AI orchestration. *Journal of Computational Analysis and Application*, 34(11):155–165.

Goldstick, J. E. and Jay, J. (2022). Agent-based modeling: an underutilized tool in community violence research. *Current Epidemiology Reports*, 9:135–141.

Hassan, M. H., Mostafa, S. A., Mustapha, A., Wahab, M. H. A., and Nor, D. M. (2018). A survey of multi-agent system approach in risk assessment. In *International Symposium on Agent, Multi-Agent Systems and Robotics*, pages 51–56. IEEE.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

Nanua, S., Steward, R., Neely, B., Datto, M., and Youens, K. (2025). Retrieval-augmented generation for interpreting clinical regulatory questions. *Journal of Pathology Informatics*, 19:100520.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pages 3982–3992.

Simon, H. A. (1955). A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1):99–118.

Stanley, N. and Humphreys, C. (2014). Multi-agency risk assessment and management for domestic violence. *Children and Youth Services Review*, 47:78–85.

Stone, P. and Veloso, M. (2000). Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3):345–383.

Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. John Wiley & Sons.