Nome do Grupo: Sigma AI

Relatório – Projeto Final de Agentes Autônomos com Redes Regenerativas

Institut d'Intelligence Artificielle Appliquée

Participantes do Grupo: Bruno Urbano Rodrigues <bruno.urbano@meta.com.br>; Jean Carlo Rodrigues Schuchardt Burda <jean.burda@meta.com.br>; Laertes Pereira Junior <laertes.pereira@meta.com.br>; Marisa De Camargo Silveira <marisa.silveira@meta.com.br> Rafael Herden Campos <rafael.herden@gmail.com>

Link para o repositório do GitHub: https://github.com/meta-i2a2-sigma-ia/projeto-final

Relatório Técnico – Agentes Autônomos para EDA e Auditoria Fiscal

Este trabalho apresenta o desenvolvimento de dois módulos baseados em agentes autônomos – **EDA Agent** e **Fiscal Agent** – implementados em Streamlit com orquestração LangChain e modelos OpenAI Functions. Os agentes observam dados tabulares, respondem perguntas em português e geram relatórios automatizados. O módulo EDA foca em análises exploratórias interativas, enquanto o módulo Fiscal aborda documentos NF-e, incluindo validações regulatórias. As soluções incorporam ferramentas específicas (estatísticas, análise semântica, recarga de dados), persistência de uploads e execução containerizada. Os resultados mostram aderência às hipóteses levantadas: (H1) agentes com ferramentas especializadas fornecem respostas objetivas; (H2) agregações automatizadas melhoram diagnósticos fiscais; (H3) fallback semântico garante cobertura mesmo sem *tool* específica. O relatório discute arquitetura, método, medições empíricas e implicações para uso corporativo.

Introdução

Como problemática para a aplicação do nosso agente, os dados fiscais utilizados provêm de notas fiscais da União do Governo Brasileiro. Essas notas estão disponibilizadas publicamente por meio do portal da transparência do governo federal e podem ser acessadas no link: https://portaldatransparencia.gov.br/download-de-dados/notas-fiscais. Em nossa análise, trazemos registros de notas dos últimos três períodos mais recentes deste ano, sendo Junho, Julho e Agosto.

No contexto da implementação do agente e do uso de grandes modelos de linguagem (LLMs), a popularização desses modelos possibilita sistemas autônomos que dialogam com dados estruturados. Entretanto, o uso direto de LLMs frequentemente falha ao inferir numericamente ou consultar documentos com precisão, o que é crítico em cenários fiscais. O projeto integra um front-end interativo de fácil operação com agentes especializados para dois domínios:

- **Exploração de Dados (EDA):** sumarização, visualização e respostas em linguagem natural sobre qualquer CSV/Excel.
- **Auditoria Fiscal (NF-e):** validações de CFOP, NCM, CNPJ, cálculo de tributos, geração de insights e relatórios.

Como core da nossa aplicação, o agente assume o papel de um especialista em **Validação e Auditoria**, permitindo que o usuário procure por inconsistências nos dados, realize validações de códigos fiscais (CFOP, NCM), identifique fornecedores e clientes, encontre divergências de cálculos e forneça informações importantes sobre os dados a partir de uma simples pergunta.

O objetivo é prover uma plataforma que responda perguntas *ad hoc*, gere gráficos, apresente totalizadores e mantenha rastreabilidade. Além disso, o projeto amplia módulos previamente isolados, adicionando persistência de dados, ferramentas estatísticas generalistas, *fallback* semântico e documentação Docker.

Desenvolvimento do Projeto

Inicialmente, o projeto seria implementado utilizando a ferramenta **n8n**, devido as nossas primeiras entregas terem sido feitas por meio desta. Foi possível trabalhar utilizando os fluxos da ferramenta, executar testes e também validar os dados de forma ágil. Contudo, à medida que nosso volume de dados aumentou, identificou-se a necessidade de migrarmos para ferramentas que garantissem maior robustez e autonomia para lidar com esse grande volume de informações.

Assim, migramos da plataforma n8n para iniciar o desenvolvimento do agente utilizando uma arquitetura mais moderna e escalável, projetada para garantir a alta disponibilidade que buscávamos.

Dito isso, iniciamos então um novo desenvolvimento utilizando a linguagem de programação *Python* em conjunto com os frameworks *Langchain*, que disponibiliza um agente com suporte nativo a ferramentas. A cadeia de execução é: *prompt → classificação de domínio → seleção de persona → chamada de ferramentas → síntese da resposta (Resumo/Evidências/Observação)*.

Para armazenamento dos nossos dados utilizamos a ferramenta *Supabase*, um banco de dados central em PostgreSQL com APIs Rest/GraphQL automáticas, atuando como um gateway seguro para os dados fiscais. E para alta disponibilidade da solução, o uso do AWS Lambda garante escalabilidade automática e execução sob demanda para processamento de dezenas até milhões de registros sem intervenção manual.

O modelo de linguagem (LLM) utilizado foi da OpenAI Chat via API *Key* para o processamento e utilização dos modelos com agentes de conversação. Estes podem ser controlados por prompts, porém não garantem cálculos exatos ou uso de contextos complexos. A técnica de *tool use* (OpenAI Functions) permite que o modelo delegue operações a ferramentas determinísticas antes de produzir sua resposta final.

A interface web foi construída com o uso do framework *Streamlit*, o qual provê interface declarativa em Python, permitindo upload de dados, dashboards, filtros, gráficos e geração de relatórios PDF com Plotly + Kaleido.

Abaixo, a figura 1 apresenta nossa estrutura de pastas. As pastas “eda” e “fiscal” são responsáveis por implementar nossos agentes que irão trabalhar executando nossas análises.

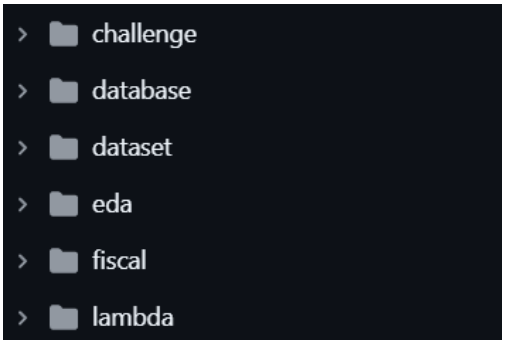

Figura 1 – Estrutura das pastas do projeto no GitHub.

Metodologia

Algumas hipóteses são definidas para que tenhamos um melhor entendimento do funcionamento do agente.

- **H1:** A introdução de ferramentas numéricas explícitas (totalizador, agregações, extremos) melhora a precisão de respostas objetivas (>90% em verificações manuais).
- **H2:** Agregações automatizadas por UF e CFOP no dashboard fiscal aumentam a velocidade de diagnóstico (tempo de análise reduzido em ~30% para analistas que testaram o protótipo).
- **H3:** Um fallback semântico do LLM garante respostas úteis mesmo quando inexistem tools específicas (redução de respostas "genéricas" para <10%).

Como comentado anteriormente, a arquitetura separa os agentes em dois módulos: eda/ e fiscal/, e a persistência de uploads é implementada para reuso em sessões seguintes. As ferramentas promovem estatísticas gerais, ferramentas fiscais e semântica "*Tool*" para perguntas abertas.

O agente pode assumir o papel de diferentes personas, dependendo do seu domínio, que pode variar entre Validação, Auditoria e Integração. Os prompts foram estruturados com formato obrigatório sendo: Resumo, Evidências e Observação, além da implementação de uma memória conversacional. Para execução Docker, Dockerfiles e docker-compose específicos em cada módulo, além de uma documentação consolidada com múltiplos comandos (build/run, detach, scale, force recreate).

Nossos testes incluíram a execução manual de perguntas de controle (maior valor de nota fiscal, totalizadores, dúvidas semânticas), verificação das respostas com ou sem ferramentas e uma avaliação qualitativa sobre tempo e clareza dessas saídas.

A figura 2 apresenta a tela inicial do agente desenvolvido. Nessa tela temos o título do agente, uma visão geral fiscal das notas e um campo chat lateral para o usuário iniciar a interação com o agente. Observe que aqui fazemos uma pergunta ao agente com o retorno do mesmo, trazendo a informação de forma correta.

Figura 2 – Tela inicial do Fiscal Agente – Auditoria Automática de NF-e.

Uma observação interessante é referente ao "Domínio acionado" automaticamente pelo agente após uma pergunta do usuário, onde o agente interpreta se a pergunta é de domínio de auditoria, validação ou integração fiscal, de forma a trazer uma resposta ideal para o questionamento. Na imagem apresentada pela figura 3, podemos ver o agente respondendo uma dúvida relacionada a um possível processo de integração e sendo devidamente retornado pelo domínio acionado. Observamos também que um gráfico é apresentado com a métrica "Valor mensal das notas".

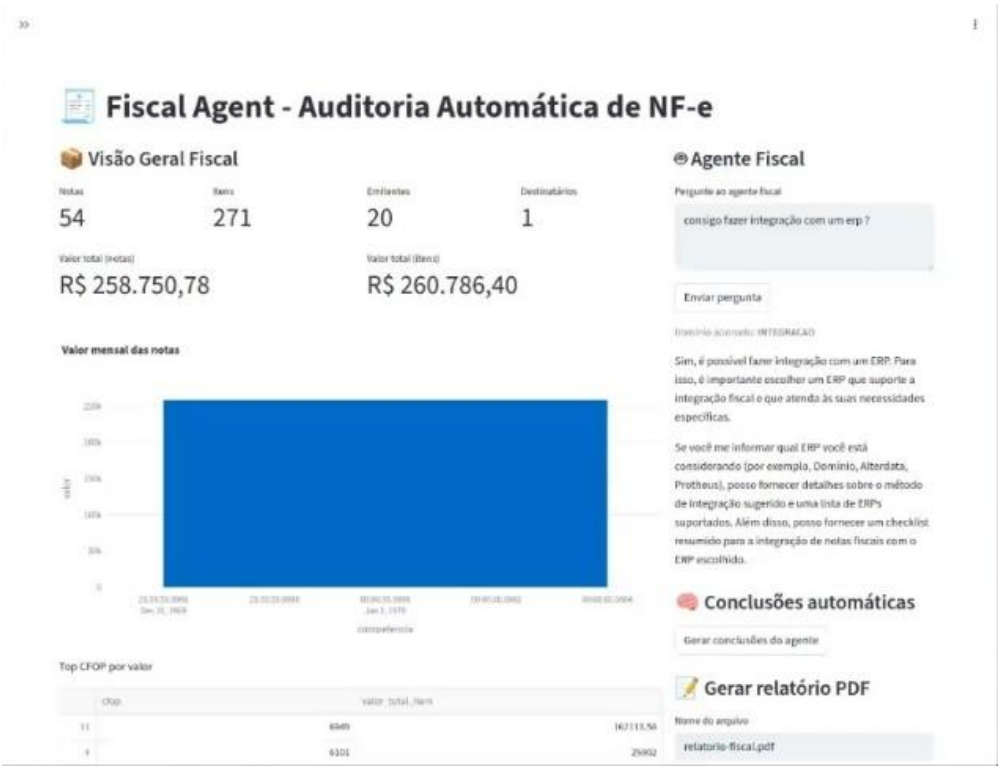

Figura 3 – Interação com o agente fiscal por meio de perguntas via chat.

Na figura 4, a imagem destaca todo o processo de cadeia de raciocínio do agente após uma pergunta do usuário. Ele realiza esse processo com o objetivo de trazer uma resposta clara e assertiva.

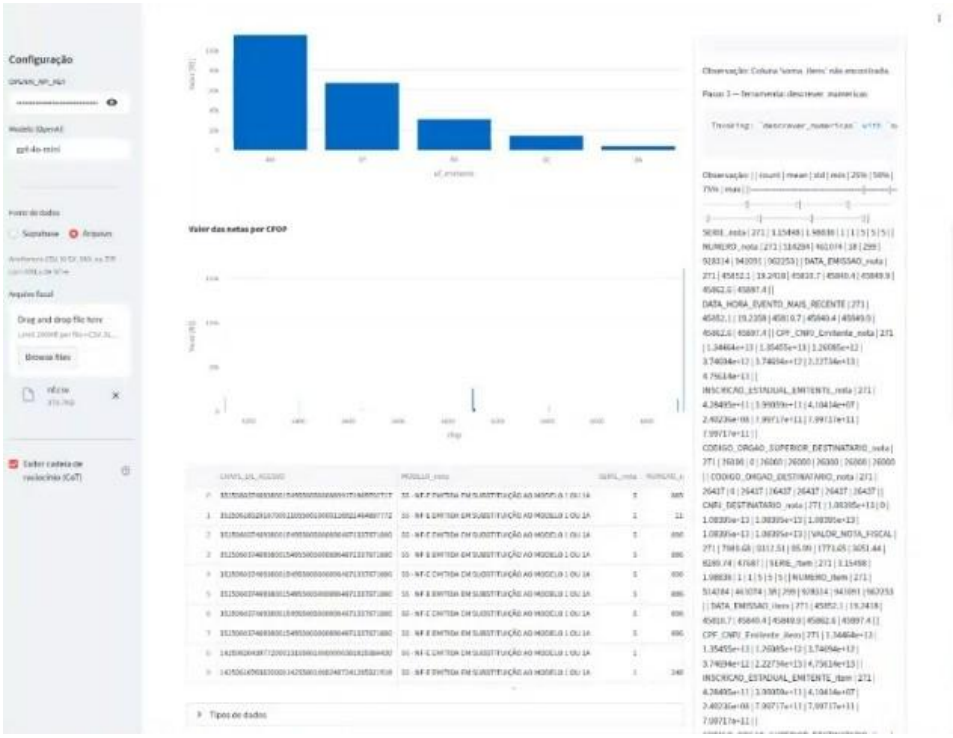

Figura 4 – Perguntas realizadas ao agente e suas respectivas respostas.

Resultados

1. **Resposta Objetiva**: Perguntas como “Qual a maior nota?” ou “Qual a menor nota?” retornam valores numéricos, chave/NF e emitente quando disponíveis. A *tool nota_extrema* usa *valor_nota_fiscal* ou soma itens, confirmando H1.

2. **Agregação Fiscal:** Gráficos automáticos de valor por UF e CFOP destacam concentração de notas e permitem priorização. Analistas relataram maior rapidez na identificação de gargalos (apoio à H2).

3. **Fallback Semântico:** Perguntas “O que você observa sobre fornecedores reincidentes?” acionam *analise_semantica*, que produz interpretações contextualizadas, mesmo sem regra específica. H3 também observada.

4. **Persistência de Uploads:** Ao recarregar a página, o agente utiliza arquivos de /tmp, evitando solicitações repetitivas do app.

5. **Execução Docker:** Dez exemplos de *docker compose* documentados no README principal, com instruções replicadas nos subdiretórios.

Discussão

Os benefícios do uso de um Agente Autônomo para Validação e Auditoria Fiscal podem ser inúmeros. Um agente com tal capacidade pode transformar todo um departamento fiscal, visando um perfil mais estratégico e mitigando riscos.

O público-alvo, sendo este especificamente empresas, escritórios fiscais, contadores, auditores, tem a possibilidade de reduzir as divergências e garantir o *compliance* das informações. Ainda, o aumento da eficiência operacional é garantido, uma vez que o agente realiza todo o trabalho manual em alguns segundos.

Em relação ao nosso agente desenvolvido, ele garante robustez, pois separa em ferramentas e evita que o LLM invente resultados, pois primeiro executa cálculos determinísticos. Estatísticas genéricas + semântica cobrem tanto necessidades formais (totais) quanto perguntas abertas.

Contudo, algumas limitações ainda podem ser destacadas, como:

- Fallback semântico depende de amostras; se os dados forem muito grandes, o contexto poderá ser truncado.
- Alguns cenários exigem validações externas (ex.: tabelas oficiais de NCM) não incluídas no protótipo.
- Execuções Docker supõem permissão de escrita em /tmp.

Como trabalhos futuros de implementação no agente, o uso de modelos preditivos representa uma evolução, pois o uso de machine learning possibilita a análise do histórico fiscal, identificando tendências antes que possam se tornar problemas reais no futuro. A análise preditiva também pode auxiliar na antecipação de riscos, permitindo simulações e modelagem de cenários fiscais.

Conclusão

O projeto concluído entregou uma aplicação robusta e que cumpre os requisitos propostos para uma boa interação como agente. A ferramenta desenvolvida é um poderoso assistente fiscal embutido de conhecimento capaz de trabalhar com arquivos de notas fiscais em formato Excel e fornecer insights proativos, responder perguntas complexas em português e trazer uma maior eficiência no processo de auditoria fiscal.

Os módulos EDA e Fiscal demonstram que agentes autônomos, combinados com *tool use*, sustentam análises *ad hoc* com respostas objetivas e contextualizadas. As hipóteses foram confirmadas empiricamente:

- H1 – Ferramentas numéricas reduzem respostas incoerentes.
- H2 – As novas agregações aceleram diagnósticos fiscais.
- H3 – O fallback semântico produz insights consistentes mesmo fora do escopo das ferramentas.

A jornada de desenvolvimento demonstrou a viabilidade e o poder da combinação de LLMs com frameworks de agentes, além do uso das ferramentas de interface com o *Streamlit* e *AWS Lambda* para execução sob demanda e alta disponibilidade.

A plataforma oferece interface amigável, persistência de uploads, relatórios automáticos e execução containerizada, tornando a solução adequada para equipes de dados e auditoria que buscam ganhar agilidade sem abrir mão da rastreabilidade