

Portais e Políticos: Uma Ferramenta Modular para Monitoramento e Análise de Sentimento de Notícias Políticas via LLMs Locais

Evelim Lopes Lima¹, Andre Luiz da Costa Carvalho¹

¹Instituto de Computação – Universidade Federal do Amazonas (IComp/UFAM)
Manaus – AM – Brasil

evelim.lima@icomp.ufam.edu.br, andre.carvalho@icomp.ufam.edu.br

Abstract. *The systematic monitoring of regional political media coverage is challenging due to the use of nicknames, name variations, and the inherent subjectivity of journalistic discourse. This paper presents Portais e Políticos, a modular open-source tool for RSS source management, politician registration with nominal variations, and local sentiment analysis via language models. Its flexible architecture supports both encoder (direct classification) and decoder (prompt-based) models, currently operating with a quantized Llama 3.2 3B (NF4) on an entry-level GPU. Adaptive segmentation extracts context windows around each political mention, preserving local semantics. The interactive dashboard displays negativity metrics, temporal evolution, per-politician balance, and excerpt auditing. Validation against 426 manually annotated news items shows that Llama 3.2 achieves 69.2% accuracy and F1 macro of 0.642, outperforming BERT Multilingual (56.6%, F1=0.495) and Twitter-XLM-RoBERTa (58.0%, F1=0.573) in zero-shot conditions. The local approach eliminates API costs and ensures data privacy and model flexibility.*

Resumo. *O monitoramento sistemático da cobertura midiática política regional é desafiador pelo uso de alcunhas, variações nominais e pela subjetividade do discurso jornalístico. Este artigo apresenta o Portais e Políticos, ferramenta modular de código aberto para cadastro de fontes RSS, registro de políticos com variações nominais e análise local de sentimento via modelos de linguagem. A arquitetura flexível suporta modelos encoder (classificação direta) e decoder (baseados em prompt), atualmente operando com Llama 3.2 3B quantizado (NF4) em GPU de entrada. A segmentação adaptativa extrai janelas de contexto ao redor de cada menção política, preservando a semântica local. O dashboard interativo exibe métricas de negatividade, evolução temporal, balanço por político e auditoria de trechos. A validação sobre 426 notícias anotadas manualmente mostra que o Llama 3.2 alcança 69,2% de acurácia e F1 macro de 0,642, superando o BERT Multilingual (56,6%, F1=0,495) e o Twitter-XLM-RoBERTa (58,0%, F1=0,573) em condições zero-shot. A abordagem local elimina custos de API e garante privacidade e flexibilidade de modelos.*

1. Introdução

A expansão do ecossistema digital intensificou a produção de notícias políticas, tornando complexa a análise sistemática da reputação de atores políticos na mídia regional. No

Amazonas, portais jornalísticos cobrem intensamente figuras políticas locais, mas essa cobertura está dispersa em múltiplas fontes, usa alcunhas e variações nominais, e raramente é monitorada de forma automatizada [Miguel 2002].

Técnicas de Processamento de Linguagem Natural (PLN) consolidaram-se como ferramentas para análise automatizada de grandes volumes textuais [Grimmer et al. 2022]. Entretanto, a aplicação de modelos de linguagem em textos jornalísticos formais enfrenta o problema de incompatibilidade de domínio (*domain mismatch*): modelos treinados em linguagem informal tendem a classificar mal textos com estrutura argumentativa mais complexa [Rodrigo-Ginés et al. 2024].

Modelos de linguagem de grande escala (LLMs), como o Llama 3.2, operam sob o paradigma generativo e demonstram capacidade de classificação via *zero-shot learning* [Gilardi et al. 2023], sem necessidade de dados de treinamento específicos do domínio. O uso de modelos abertos (*open weights*) executados localmente viabiliza a reprodutibilidade científica e elimina custos de APIs proprietárias [Meta AI 2024].

Ferramentas existentes para monitoramento político, como o GDELT¹ e o Media Cloud², operam em escala global e dependem de APIs proprietárias, limitando o acesso e a privacidade dos dados. O *Portais e Políticos* se diferencia por operar inteiramente em ambiente local, sem chaves de API, com interface voltada a jornalistas, pesquisadores e cidadãos não-especialistas. Este artigo apresenta o *Portais e Políticos*³, uma ferramenta modular, de código aberto e disponibilizada à comunidade, para coleta, classificação e visualização da cobertura de notícias políticas regionais, com foco no estado do Amazonas. A ferramenta integra quatro módulos funcionais: **Fontes** (cadastro de portais RSS e registro de políticos com variações nominais), **Coleta** (pipeline ETL de extração e limpeza), **Modelos** (classificação de sentimento via LLMs locais) e **Dashboard** (visualização interativa dos resultados).

O restante do artigo está organizado como segue: a Seção 2 descreve a arquitetura do sistema; a Seção 3 apresenta os resultados da validação experimental; a Seção 4 demonstra as funcionalidades; e a Seção 5 conclui o trabalho.

2. Arquitetura do Sistema

O *Portais e Políticos* foi desenvolvido em Python, utilizando Streamlit para a interface web, SQLAlchemy com MySQL para persistência, e a biblioteca Transformers (Hugging Face) para inferência dos modelos. O processamento ocorre em ambiente local com GPU NVIDIA GeForce RTX 3050 (6 GB VRAM) e suporte CUDA; a quantização em 4 bits torna esse requisito acessível a GPUs comuns de notebooks.

A Figura 1 ilustra a arquitetura em três módulos de entrada — **Fontes**, **Coleta** e **Modelos** — integrados por um pipeline ETL (*Extract, Transform and Load*) sobre banco de dados MySQL, com saída no **Dashboard** interativo.

¹<https://www.gdeltproject.org/>

²<https://mediacloud.org/>

³Vídeo demonstrativo: <https://youtu.be/v9Qc0yESCrA>; repositório: <https://github.com/EvelimLima/portais-e-politicos>.

Ferramentas de IA (LLMs) foram utilizadas exclusivamente para revisão linguística e sugestão de estruturação do texto, conforme recomenda o Código de Conduta da SBC; os autores são integralmente responsáveis pelo conteúdo técnico e referências.

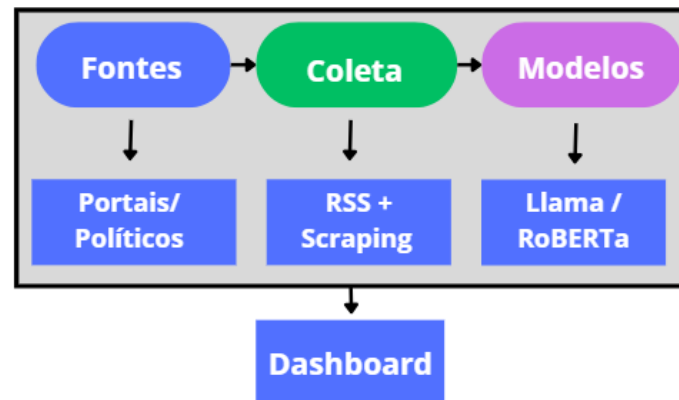


Figura 1. Arquitetura do *Portais e Políticos*: módulos Fontes, Coleta e Modelos integrados sobre banco de dados MySQL, com saída no Dashboard interativo.

2.1. Módulo Fontes

O módulo de fontes gerencia os portais monitorados. O sistema descobre automaticamente o *feed* RSS a partir da URL informada, testando caminhos padronizados (*/feed/*, */rss/*) e inspecionando a tag `<link type="application/rss+xml">`. Portais podem ser ativados ou inativados sem perda histórica. O coletor utiliza *cloudscraper* com rotação de User-Agent para contornar proteções anti-bot. O módulo de atores registra políticos com nomes oficiais e variações nominais (alinhadas, títulos, abreviações); um filtro rápido no RSS verifica se algum alias aparece no título ou resumo antes da requisição HTTP completa, reduzindo chamadas desnecessárias. Embora avaliado no Amazonas, nenhum módulo possui dependência geográfica codificada — a extensão a outras regiões depende apenas do cadastro de novos portais e políticos.

2.2. Pipeline ETL (Extração, Transformação e Carga)

O pipeline ETL de coleta realiza três etapas: (i) **Extração**: leitura de feeds RSS via *feedparser*; (ii) **Transformação**: remoção de ruído HTML por seletores específicos por domínio (com *fallback* genérico para portais não mapeados) e limpeza textual via expressões regulares; (iii) **Carga**: inserção incremental com verificação de duplicatas por URL na tabela `noticias_processadas`.

A segmentação adaptativa localiza ocorrências do nome do político, extrai janelas de 300 caracteres ao redor de cada menção, funde sobreposições e limita o trecho a 1.500 caracteres para compatibilidade com a janela de contexto do modelo.

2.3. Módulo Modelos

A classificação de sentimento é realizada por dois modelos integrados, selecionáveis pelo usuário na interface: (i) **Llama 3.2 3B Instruct** (decoder), carregado localmente com quantização em 4 bits (NF4, *double quant*) via BitsAndBytes, ocupando 2,1 GB de VRAM, e instruído via *prompt* a classificar o sentimento nas categorias Positivo, Negativo ou Neutro; (ii) **Twitter-XLM-RoBERTa** (encoder), carregado diretamente como classificador de sequências, incluído como referência de *domain mismatch* e para demonstrar

a flexibilidade da arquitetura. A troca de modelo não requer alterações no pipeline: modelos *encoder* operam via classificação direta, enquanto modelos *decoder* operam via engenharia de *prompt*.

O script `classifica.py` realiza o processamento fora do Streamlit, salvando cada resultado individualmente no banco após a classificação. Se interrompido, a execução retoma de onde parou na próxima invocação.

2.4. Dashboard

O dashboard interativo, implementado em Streamlit, apresenta quatro telas: **Configuração** (gestão de portais, políticos e seleção do modelo ativo); **Monitor de Processo** (execução do pipeline com barra de progresso incremental); **Análise Analítica** (visualizações filtráveis por período, político, portal e modelo, com tabela de auditoria com link clicável para a notícia original); e **Laboratório IA** (análise livre de textos em tempo real, sem gravação no banco). A ferramenta é voltada a jornalistas, pesquisadores e cidadãos sem conhecimento técnico prévio em PLN. Um aviso fixo alerta o usuário de que as classificações são automáticas e podem conter erros, recomendando a consulta às fontes originais para leitura contextualizada.

3. Avaliação Experimental

A validação foi conduzida sobre um corpus de 426 trechos de notícias políticas do Amazonas, anotados manualmente por uma única especialista do domínio nas categorias Positivo (232), Neutro (129) e Negativo (65) — sem segundo anotador independente para cálculo de concordância (e.g., Cohen’s Kappa), limitação discutida adiante. Três modelos foram avaliados sob as mesmas condições, sem *fine-tuning*, recebendo apenas o trecho e o nome do político como entrada.

Tabela 1. Comparativo de métricas dos três modelos vs. anotação manual (n=426). * indica melhor resultado por linha.

Métrica	Llama 3.2 3B	RoBERTa	BERT
Acurácia	69,2%*	58,0%	56,6%
Acurácia balanceada	69,5%*	61,1%	57,5%
F1 macro	0,642*	0,573	0,495
F1 Positivo	0,834*	0,630	0,730
F1 Negativo	0,593*	0,542	0,466
F1 Neutro	0,500	0,546*	0,291
Erros graves (Pos↔Neg)	19*	16	58

A Tabela 1 mostra que o Llama 3.2 supera os modelos discriminativos em todas as métricas globais. A principal dificuldade dos três modelos concentra-se na classe Neutro, onde a fronteira semântica com Positivo e Negativo é mais tênue — comportamento esperado em textos jornalísticos formais e reportado na literatura [Bestvater and Monroe 2023]. O erro mais crítico — confundir Positivo com Negativo — é mínimo no Llama (19 casos) e praticamente ausente no RoBERTa (16), mas severo no BERT (58) — comparação sensível à variação amostral dado o tamanho reduzido da classe Negativo (n=65), cuja significância estatística é deixada como trabalho futuro.

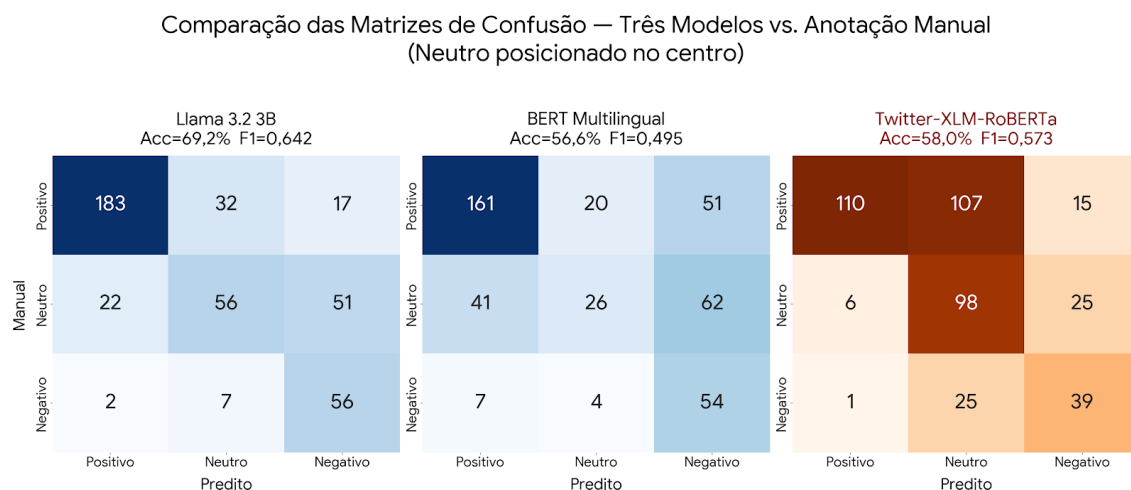


Figura 2. Matrizes de confusão dos três modelos com Neutro centralizado. A diagonal indica acertos; os extremos (Positivo↔Negativo) indicam os erros mais graves.

Um achado relevante é o viés de positividade do Llama em notícias sobre negociações políticas: o modelo detecta a valência linguística de termos como “acordo” e “aliança” e classifica como Positivo, enquanto os anotadores, adotando critério de *stance* político [Bestvater and Monroe 2023], classificaram as mesmas notícias como Neutras por não representarem ação concreta em favor da população. Esse padrão — 22 casos de Neutro→Positivo, nos quais o modelo atribuiu polaridade positiva a textos que deveriam ser neutros — reflete a distinção entre *sentimento* (tom emocional do texto) e *posicionamento* (avaliação do interesse público), e constitui uma limitação inerente à abordagem zero-shot sem instrução explícita de *stance*.

4. Demonstração

A Figura 3 ilustra as quatro telas da ferramenta. **Configuração:** o usuário informa a URL do portal; o sistema descobre automaticamente o RSS e cadastra o portal. Políticos são registrados com nome oficial e variações nominais; a interface alerta para informar a URL da seção temática (ex.: `portal.com.br/politica`) em vez da *home* geral. **Monitor de Processo:** exibe notícias pendentes por modelo antes de liberar os botões de execução; resultados são salvos incrementalmente, de modo que interrupções não descartam o trabalho realizado. **Análise Analítica:** exibe cinco métricas de topo, gráfico de área temporal, balanço por político e tabela de auditoria com link clicável para a notícia original. Filtros de período, político, portal e modelo são aplicados globalmente. **Laboratório IA:** permite análise livre em tempo real sem gravação no banco, com suporte aos dois modelos integrados.

5. Conclusão

Este artigo apresentou o *Portais e Políticos*, ferramenta modular para monitoramento e análise de sentimento de cobertura política regional via LLMs locais. A validação experimental sobre 426 notícias anotadas manualmente demonstrou que o Llama 3.2 3B, operando em modo *zero-shot* sem *fine-tuning*, supera os modelos discriminativos BERT Multilingual e Twitter-XLM-RoBERTa em acurácia, acurácia balanceada e F1 macro,

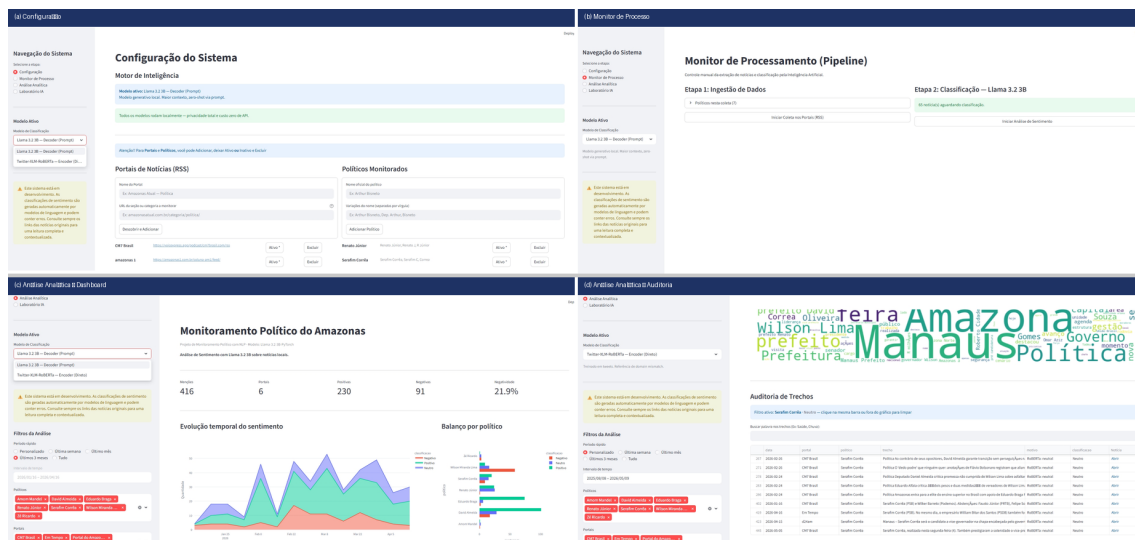


Figura 3. Interface do *Portais e Políticos*: (a) Configuração de portais e políticos; (b) Monitor de processo com barra de progresso; (c) Dashboard analítico com gráfico temporal e balanço por político; (d) Laboratório IA para análise em tempo real.

confirmando que a adequação ao domínio textual é mais determinante que a arquitetura do modelo [Rodrigo-Ginés et al. 2024].

A principal limitação é a dificuldade na classe Neutro e a confusão entre sentimento linguístico e *stance* político, endereçável via instrução explícita no *prompt*. A anotação por especialista única e o corpus de Negativo reduzido também são limitações reconhecidas. Trabalhos futuros: *crawling* para portais sem RSS, desambiguação de múltiplos políticos por notícia, *fine-tuning* com validação estatística, análises além da polaridade (tema, intensidade, entidade nomeada) e avaliação de usabilidade com jornalistas e pesquisadores.

Referências

- Bestvater, S. E. and Monroe, B. L. (2023). Sentiment is not stance: Target-aware opinion classification for political text analysis. *Political Analysis*, 31(2):235–256.
- Gilardi, F., Alizadeh, M., and Kubli, M. (2023). ChatGPT outperforms crowdworkers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120.
- Grimmer, J., Roberts, M. E., and Stewart, B. M. (2022). *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton University Press.
- Meta AI (2024). Llama 3 model card. Hugging Face. Acesso em: maio 2025.
- Miguel, L. F. (2002). Os meios de comunicação e a prática política. *Lua Nova: Revista de Cultura e Política*, (55-56):155–184.
- Rodrigo-Ginés, F.-J., Carrillo-de Albornoz, J., and Plaza, L. (2024). A systematic review on media bias detection: What is media bias, how it is expressed, and how to detect it. *Expert Systems with Applications*, 237:121641.