Desafios de Propriedade Intelectual no Desenvolvimento de Software Público: Uma Análise de Licenças no Ecossistema Python

Marjory S. S. Lima, Josiel M. Figueiredo

Instituto de Computação, Universidade Federal de Mato Grosso, Brazil

marjory.lima@sou.ufmt.br, josiel@ufmt.br

Abstract. *This paper addresses intellectual property and licensing challenges in NLP solutions for the public sector. A case study of a semantic ranking framework for court rulings shows that, despite the predominance of permissive licenses (MIT, Apache, BSD), GPL components introduce critical restrictions. The analysis highlights the need for compliance, adoption of standards such as SPDX, and verification tools to ensure legal sustainability in public software projects.*

Resumo. *Este artigo discute desafios de propriedade intelectual e licenciamento em soluções de PLN para o setor público. O estudo de caso de um framework de ranqueamento semântico de acórdãos mostra que, embora predominem licenças permissivas (MIT, Apache, BSD), a presença de componentes sob GPL gera restrições relevantes. A análise destaca a necessidade de compliance, uso de padrões como SPDX e ferramentas de verificação para garantir sustentabilidade jurídica em projetos de software público.*

1. Introdução

A construção de soluções de Processamento de Linguagem Natural (PLN) para o setor público brasileiro, incluindo aquelas direcionadas à análise de decisões judiciais, depende da utilização intensiva de bibliotecas *open source*, uma vez que o ecossistema Python, predominante nessas aplicações, é amplamente estruturado sobre componentes colaborativos e de código aberto [Silva *et al.* 2023]. Estudos mostram que esse ecossistema apresenta fragilidades importantes: mais de 10% dos pacotes disponíveis no Python Package Index (PyPI) não possuem licença explícita [Snyk 2018] e cerca de 7,3% das versões publicadas apresentam incompatibilidades de licenciamento [Xu *et al.* 2023]. Esse cenário traz implicações críticas de propriedade intelectual e sustentabilidade, uma vez que a incorporação de múltiplas dependências expõe projetos a riscos de incompatibilidade de licenças e à falta de padronização [Kapitsaki *et al.* 2023].

Nesse contexto, a análise de aspectos de governança e licenciamento torna-se importante para a construção de soluções robustas e juridicamente seguras. O presente artigo discute esses desafios no desenvolvimento de software público voltado ao Judiciário, com ênfase na necessidade de estratégias de *compliance*, uso de padrões como os definidos pela Software Package Data Exchange (SPDX), especificação mantida pela Linux Foundation que estabelece identificadores únicos e consistentes para licenças de *software*, e práticas de inovação aberta que favoreçam a sustentabilidade. Como estudo ilustrativo, utilizamos um *framework* de ranqueamento

semântico de acórdãos trabalhistas — detalhado na Seção 3 — que evidencia os dilemas e equilíbrios envolvidos na escolha e integração de bibliotecas externas.

Este artigo caracteriza-se como um trabalho em andamento, resultante da etapa inicial de análise de riscos de licenciamento no desenvolvimento de *software* público. O estudo encontra-se atualmente na fase de mapeamento de dependências e avaliação de alternativas de mitigação.

2. Fundamentação

A literatura sobre inovação aberta e *software* público destaca que a sustentabilidade de projetos digitais no setor público depende não apenas da qualidade técnica, mas da gestão das licenças de *software* e da governança das comunidades envolvidas [Linåker *et al*. 2025; Pereira *et al.* 2019].

No campo da propriedade intelectual, o Brasil adota a Lei nº 9.609/1998, alinhada ao Acordo TRIPS (*Trade-Related Aspects of Intellectual Property Rights*) da Organização Mundial do Comércio (OMC), que estabelece padrões mínimos de proteção internacional. Contudo, persistem tensões entre incentivar a inovação e garantir a difusão do conhecimento [Andrade *et al.* 2007; ENAP 2021], agravadas por lacunas regulatórias no campo da inteligência artificial [Santos 2022].

Outro eixo de debate envolve os ecossistemas de *software* aberto. Pesquisas revelam riscos relevantes de licenciamento, como a ausência de licenças explícitas em pacotes do PyPI [Snyk 2018], incompatibilidades em dependências transitivas [Xu *et al.* 2023; Kapitsaki *et al.* 2023] e fenômenos recentes de *license drift* em IA, que ocorre quando informações de licenciamento são gradualmente perdidas ou alteradas durante o reuso de *datasets*, modelos e derivados [Jewitt *et al*. 2025]. Nesse cenário, a governança transparente inclui não apenas comunidades ativas e políticas institucionais [Linåker *et al.* 2025], mas também uma gestão clara das licenças das dependências utilizadas.

No contexto brasileiro, a Lei nº 9.609/1998 reconhece o programa de computador como obra intelectual protegida, independentemente de registro, permitindo sua exploração mediante contratos de licença. Embora o modelo *open source* adote uma lógica distinta — baseada em concessões públicas de uso e modificação —, não há conflito jurídico direto: as licenças livres funcionam como instrumentos contratuais válidos dentro do regime da lei.

3. Produto Proposto e Estratégia de Inovação Aberta

Para exemplificar os desafios de propriedade intelectual em projetos de PLN jurídico, consideramos o desenvolvimento de um *framework* de ranqueamento semântico de acórdãos. Embora o objetivo técnico do sistema seja organizar decisões judiciais por similaridade semântica, sua principal contribuição para este estudo está no conjunto de bibliotecas empregadas. O protótipo depende de ferramentas como `NLTK` (Apache 2.0), `spaCy` (MIT), `sentence-transformers` (Apache 2.0), `HDBSCAN` (BSD), entre outras.

A análise das licenças dessas dependências revela a diversidade de modelos contratuais coexistentes e aponta riscos de incompatibilidade quando tais componentes são combinados em um produto único. Essa constatação reforça a necessidade de incorporar práticas de governança de licenciamento desde o início do desenvolvimento,

incluindo a verificação automatizada de licenças e a documentação padronizada dos termos de uso. Como discutem Kapitsaki *et al.* [2023], tais medidas mitigam riscos de incompatibilidade e promovem transparência, fatores essenciais para a sustentabilidade de projetos públicos de PLN.

4. Propriedade Intelectual e Licenciamento

A sustentabilidade do *framework* depende de duas frentes complementares: a escolha de uma licença permissiva, como Apache 2.0 ou MIT, e a implementação de rotinas permanentes de verificação e transparência. Estudos recentes demonstram que incompatibilidades de licenças em cadeias de dependências são frequentes no ecossistema Python, o que reforça a necessidade de verificar a compatibilidade entre bibliotecas externas e a licença escolhida para o produto [Xu *et al.* 2023; Kapitsaki *et al.* 2023].

Nesse cenário, ferramentas como `pip-licenses`, que permitem auditar automaticamente as licenças de dependências em ambientes Python, e o padrão SPDX (PEP 639), voltado à padronização da nomenclatura de licenças, têm sido amplamente reconhecidos como boas práticas de governança em projetos de *software* livre e institucional [Kapitsaki *et al.*, 2023]. Um ponto crítico emerge da presença da biblioteca `unidecode`, licenciada sob GPL-2.0. Essa exigência de *copyleft* colide diretamente com o objetivo de muitos projetos de *software* público, que buscam licenças permissivas (como Apache 2.0 ou MIT) para maximizar a reutilização e a integração com outras soluções governamentais, sem impor restrições de licenciamento a projetos derivados.

Como alternativa técnica à biblioteca `unidecode`, é possível o uso da `text-unidecode`, licenciada sob a Artistic License 1.0, que oferece funcionalidade equivalente de normalização de caracteres sem impor restrições de *copyleft*. Essa substituição reduz o risco de incompatibilidade e mantém a compatibilidade com licenças permissivas.

5. Resultados Parciais

Os resultados parciais deste estudo concentram-se na análise de licenciamento das bibliotecas empregadas no desenvolvimento do protótipo de ranqueamento semântico de acórdãos. A Tabela 1 apresenta um recorte das principais dependências identificadas, suas respectivas licenças e os riscos associados de incompatibilidade.

Tabela 1 - Dependências e licenças do protótipo

Biblioteca	Função Principal	Licença	Observações sobre riscos/incompatibilidade
`NLTK`	Pré-processamento de texto	Apache 2.0	Compatível com uso acadêmico e governamental
`spaCy`	Tokenização e análise linguística	MIT	Licença permissiva, baixo risco de conflito
`unidecode`	Normalização de caracteres	GPL v2	Pode gerar incompatibilidades em combinação com Apache 2.0
`sentence-transformers`	Embeddings semânticos (SBERT)	Apache 2.0	Compatível, mas exige citação adequada de modelos base

`torch (PyTorch)`	Backend para aprendizado profundo	BSD	Geralmente compatível; requer verificação de dependências adicionais
`hdbscan`	Clusterização	BSD	Compatível, sem restrições relevantes
`scikit-learn`	Aprendizado de máquina	BSD	Amplamente aceito; compatível
`umap-learn`	Redução de dimensionalidade	BSD	Opcional; compatível

A análise demonstra que, embora a maioria das bibliotecas utilize licenças permissivas (MIT, Apache, BSD), a presença de componentes sob GPL v2 pode gerar incompatibilidades em contextos de redistribuição como software público. Essa constatação reforça a necessidade de mecanismos de compliance baseados em padrões como SPDX e do uso de ferramentas como `pip-licenses` e LiDetector, capazes de identificar e mitigar conflitos de forma automatizada.

Esses resultados qualitativos mostram que a sustentabilidade de *frameworks* jurídicos em código aberto depende tanto da arquitetura técnica quanto da gestão adequada de licenças e direitos de propriedade intelectual.

6. Conclusão

Este artigo destacou que a principal barreira à sustentabilidade de soluções de PLN no Judiciário brasileiro não está apenas nos aspectos técnicos, mas nos desafios de propriedade intelectual e licenciamento das bibliotecas utilizadas. O estudo de caso do ranqueamento semântico de acórdãos mostrou que a diversidade de licenças, MIT, Apache, BSD, entre outras, exige estratégias ativas de *compliance* e governança.

Conclui-se que a consolidação de *frameworks* jurídicos como *software* público requer não apenas inovação tecnológica, mas também atenção à compatibilidade de licenças, à mitigação de riscos de *license drift* e à conformidade com padrões internacionais como o TRIPS-OMC. Assim, iniciativas baseadas em inovação aberta só alcançarão maturidade institucional se aliarem a colaboração técnica à segurança jurídica.

Como continuidade deste trabalho, prevê-se a substituição de bibliotecas incompatíveis, a aplicação de ferramentas de auditoria automatizada e a consolidação de diretrizes para licenciamento seguro em soluções de PLN. Esses desdobramentos buscam transformar os resultados parciais apresentados em um *framework* plenamente aderente às exigências legais do *software* público.

7. Referências

Andrade, E., Tigre, P. B., Silva, L. F., Silva, D. F., Moura, J. A. C., Oliveira, R. V., and Souza, A. (2007). Propriedade Intelectual em Software: o que podemos apreender da experiência internacional? *Revista Brasileira de Inovação*, 6(1), 31-53. Rio de Janeiro: FINEP. DOI: https://doi.org/10.20396/rbi.v6i1.8648940.

ENAP. (2021). *Instrumentos jurídicos para inovação aberta*. Brasília: Escola Nacional de Administração Pública. Disponível em: https://repositorio.enap.gov.br/handle/1/8075.

Jewitt, J., Capel, R., Harman, M., and Jia, Y. (2025). From Hugging Face to GitHub: Tracing License Drift in Machine Learning. *arXiv preprint*, arXiv:2509.09873. DOI: https://doi.org/10.48550/arXiv.2509.09873.

Kapitsaki, G., Paphitou, A., and Achilleos, A. (2023). Towards open source software licenses compatibility check. In *Proceedings of the 26th Pan-Hellenic Conference on Informatics* (PCI '22), (pp. 96-101). New York: Association for Computing Machinery. DOI: https://doi.org/10.1145/3575879.3575973.

Linåker, J., Lundell, B., Servant, F. et al. (2025). Public sector open source software projects - How is development organized?. *Empir Software Eng* 30, 80. https://doi.org/10.1007/s10664-025-10626-0.

Pereira, V. S., Araujo, R. M., & Santos, R. P. (2019). A Study on the Brazilian Public Software Portal Ecosystem Life Cycle and Collaboration. In *Anais do XV Simpósio Brasileiro de Sistemas de Informação* (pp. 1–8). Aracaju: ACM. DOI: https://doi.org/10.1145/3330204.3330261.

Santos, M. F. S. (2022). Regulação da inteligência artificial no Brasil: análise dos projetos de lei em tramitação na Câmara dos Deputados e no Senado Federal (Dissertação de Mestrado). Universidade Federal de Minas Gerais. DOI (ou identificador): https://hdl.handle.net/1843/52662.

Silva, L. C.; Santos, G. H.; Rodrigues, A. M. (2023). Investigando o Uso da Inteligência Artificial em Projetos Python Hospedados no GitHub. In: *Anais do Workshop de Visualização, Evolução e Manutenção de Software* (VEM 2023). SBC, 2023. DOI: https://sol.sbc.org.br/index.php/vem/article/view/30278.

Silveira, R., Ponte, C., Almeida, V., Pinheiro, V. and Furtado, V. (2023) "LegalBERT-pt: A Pretrained Language Model for the Brazilian Portuguese Legal Domain", In: *Proceedings of the 12th Brazilian Conference on Intelligent Systems (BRACIS)*, Belo Horizonte, Brazil. Sociedade Brasileira de Computação. Available at: https://sol.sbc.org.br/index.php/bracis/article/view/28420.

Souza, F., Nogueira, R., and Lotufo, R. (2020). Bertimbau: pretrained bert models for brazilian portuguese. In *Proceedings of the Brazilian Conference on Intelligent Systems*, pages 403–417. Springer. DOI: https://doi.org/10.1007/978-3-030-61377-8_28.

Snyk. (2018). Over 10% of Python packages on PyPI are distributed without a license. *Snyk Blog*. Available at: https://snyk.io/blog/over-10-of-python-packages-on-pypi-are-distributed-without-any-license/.

Xu, S., Gao, Y., Fan, L., Liu, Z., Liu, Y., and Ji, H. (2023). LiDetector: License Incompatibility Detection for Open Source Software. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 32(1), Article 22, 28 pages. ACM. DOI: https://doi.org/10.1145/3518994.