Costyl: Estilometria Aplicada na Identificação de Autoria de Código Fonte

Palavras-Chave: Estilometria, Autoria, Código

Autores(as):

Andrew Luigi Ferreira Lima, COTIL – UNICAMP

João Vitor Bernardis, COTIL – UNICAMP

Profª. Tânia Basso, COTIL – UNICAMP

INTRODUÇÃO:

No atual cenário da computação, a utilização de códigos externos - e de autoria de terceiros - é considerada comum. Porém, quando a autoria desses códigos é reivindicada, pode acabar desencadeando até disputas judiciais.

Paralelo ao progresso tecnológico e à propagação da internet, cresce também o índice de autores nos mais diversos tipos de mídia. Decorrente dessa relação, vários tipos de disputas de autoria vêm ganhando notabilidade, já que com a expansão da comunidade de autores e com certa anonimidade dada pela internet, se tornaram mais comuns os casos de plágio e disputa de *copyrights*. Assim, técnicas de análise de software para a atribuição de autoria se tornam cada vez mais importantes.

Portanto, o Costyl surgiu para suprir essa demanda. o Costyl é uma biblioteca desenvolvida em Python para auxiliar a determinar possíveis autores de códigos fontes de programas de computador. Ela utiliza técnicas de estilometria (ou seja, identificação de características e estilo de escrita) para identificar padrões de código e então comparar, por meio de modelos de aprendizado de máquina, os padrões de uma base de dados, identificando, assim, possíveis autores.

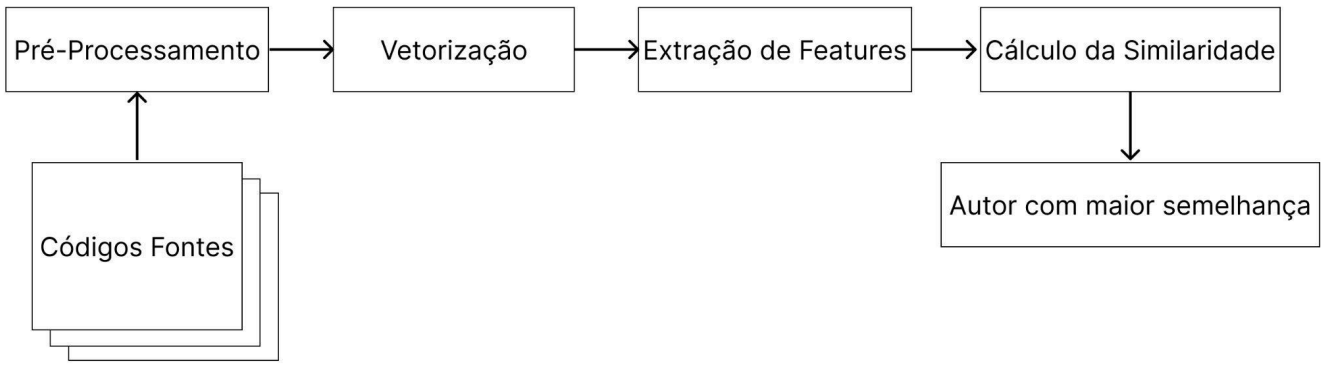

Figura 1. Módulos de funcionalidades da biblioteca Costyl

METODOLOGIA:

A biblioteca foi desenvolvida em Python (2024) por ter um grande ecossistema de algoritmos de machine learning, sendo o KNN (K-Nearest Neighbors (W3Schools, 2024)) o escolhido para fazer a predição do autor, pois ele foi mais bem sucedido em nossos casos de testes do que demais algoritmos, como por exemplo o Random Forest. O KNN é um classificador que funciona com base no quanto uma feature (característica do código) está mais próxima de outra, ou seja, o quanto esses códigos são semelhantes.

As features selecionadas para o nosso modelo podem ser divididas em duas partes: As de layout (características estruturais do código) e as léxicas (características textuais) (figura 2). Elas foram selecionadas com base no trabalho de Dong et al. (2020) e têm como objetivo identificar estilos de código de diferentes indivíduos para definição de perfis e realização de comparações.

Antes de passar pela etapa de extração de features, os códigos devem passar pelo processo de vetorização, que consiste em transformá-los em um texto para que seja possível identificar suas características. No entanto, os processos são transparentes para o usuário, ou seja, o Costyl funciona de maneira que o usuário só precisa incluir a base de teste e o arquivo cujo autor é desconhecido. Após, basta instanciar o modelo, testá-lo com a base e fazer a predição.

A build da biblioteca foi gerada através da ferramenta Poetry (2024) e disponibilizada no PyPI (2024), que é um repositório de software para a linguagem de programação Python.

Features de Layout
Uso de tabs ou espaços
Número de espaços
Número de linhas vazias
Número de comentários
Tamanho médio da linha

Features Léxicas
Número de palavras reservadas
Número de operadores
Número de literais

Figura 2. Features (características de código fonte) utilizadas na biblioteca Costyl

RESULTADOS E DISCUSSÃO:

Realizamos quatro testes diferentes utilizando códigos de competições de programação da Google (como por exemplo a Code Jam (2024)) para verificar a eficácia do algoritmo. Os testes se diferenciam na quantidade de autores e de códigos destes. Foi verificado que quanto maior a quantidade de código por autor, maior acurácia do classificador (em nossos testes, a acurácia ficou em torno de 70% para a maior quantidade de autores). Vale ressaltar que a base de dados foi selecionada de forma que os códigos de cada autor tinham as mesmas funcionalidades dos demais. Além disso, os códigos selecionados não utilizaram nenhuma biblioteca ou framework.

CONCLUSÕES:

Espera-se que a solução proposta possa ser utilizada para identificação de similaridade e atribuição de autoria de código fonte, contribuindo, assim, com o cenário atual de autoria de código fonte de sistemas, auxiliando a solucionar indicações de plágio e disputa de autoria em âmbito acadêmico e em demais casos.

BIBLIOGRAFIA

Code Jam. Competição de programação internacional hospedada e administrada pelo Google. Disponível em: https://codingcompetitionsonair.withgoogle.com/. Acesso em 26 de jun. de 2024;

Dong, W., Feng, Z., Wei, H., & Luo, H. (2020). *A Novel Code Stylometry-based Code Clone Detection Strategy*. 2020 IEEE International Wireless Communications and Mobile Computing (IWCMC), 2020. p. 1516-1521;

Poetry. *Python Packaging And Dependency Management Made Easy*. Disponível em: https://python-poetry.org/docs/. Acesso em 26 de jun. de 2024;

PyPI. *Python Package Index*. Disponível em em: https://pypi.org/. Acesso em 26 de jun. de 2024;

Python. *Python 3.12.4 documentation*. Disponível em: https://www.python.org/doc/. Acesso em 26 de jun. de 2024;

W3Schools. *Machine Learning - K-nearest neighbors (KNN)*. Disponível em: https://www.w3schools.com/python/python_ml_knn.asp. Acesso em 26 de jun. de 2024.