UNIVERSIDADE DA BEIRA INTERIOR
Engenharia

Métodos Eficientes de Deteção de Plágio em Grandes Corpora

Bruno Garcia Prata Graciano Felipe

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática
(2º ciclo de estudos)

Orientador: Prof. Doutor João Paulo da Costa Cordeiro

Covilhã, Outubro de 2016

Agradecimentos

Quero expressar o meu profundo agradecimento ao Professor Doutor João Paulo da Costa Cordeiro pela orientação facultada no âmbito desta dissertação de mestrado.
Assim como, quero carinhosamente expressar o meu agradecimento à minha Família, em especial aos meus Pais e aos meus Irmãos, pelo amor partilhado.
Também, quero agradecer à Quitolina.

Resumo

O crescente aumento da quantidade de informação publicada na *Web*, na forma de publicações literárias, científicas e académicas, implica uma constante verificação da integridade de novos documentos (suspeitos) em função dos documentos existentes (fonte). Surge, portanto, a necessidade de aumentar: a eficiência na redução do espaço de procura em grandes conjuntos de documentos fonte; a eficácia na deteção de plágios cada vez mais sofisticados. Nesta dissertação descreve-se uma metodologia baseada em dois atos: (i) indexação do *corpus* fonte, com um motor de pesquisa (código aberto), e extração de documentos fonte (candidatos), através de pesquisa por palavras relevantes e caraterísticas textuais; (ii) localização de excertos de plágio em documentos suspeitos, com uma métrica robusta, criada através da aplicação de programação genética sobre as caraterísticas de dados plagiados. Os resultados experimentais obtidos mostram uma redução significativa no tempo de processamento, devido à estratificação do *corpus*, assim como a capacidade de detetar eficientemente excertos de plágio literal, modificado e ofuscado. [FC15]

Palavras-chave

Deteção de Plágio Externo, Recuperação Fonte, Pesquisa de Informação, Análise Detalhada, Similaridade Textual, Mineração de Dados, Programação Genética.

Abstract

The increasing information volume published in the Web, either in terms of literary publications or scientific and academic papers, requires a constant surveillance to verify the integrity of daily entering new documents (suspicious), on the basis of the existing ones (sources). As a consequence arises the need to improve the efficiency in reducing the search space for large sets of documents source and the effectiveness in detecting increasingly sophisticated plagiarism events. In this dissertation it is described a methodology based on two actions: (I) indexing the source corpus, with a search engine (open-source), and the extraction of source documents (candidates) by searching for key relevant words and textual features; (II) locating plagiarized passages in suspicious documents with a hybrid metric created by applying genetic programming on the characteristics of plagiarized data. The results show a significant reduction in processing time due to the corpus stratification, as well as a high success rate in detecting plagiarism passages, having none, low, and high obfuscation. The experimental results show a significant reduction in processing time due to stratification of the corpus, as well as the ability to detect plagiarism extracts of diffrent kind: literal, modified and obfuscated. [FC15]

Keywords

External Plagiarism Detection, Source Retrieval, Information Retrieval, Detailed Comparison, Textual Similarity, Data Mining, Genetic Programming.

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Lista de Acrónimos

ARFF	*Attribute-Relation File Format*
FIT	Função de Importância de Termos
JSON	*JavaScript Object Notation*
MP	Motor de Plágio
PAN	*Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse*
TF-IDF	Frequência de Termos e Frequência Inversa Documental
TIC	Tecnologias de Informação e Comunicação
XML	*Extensible Markup Language*

Capítulo 1

Introdução

1.1 Motivação

No atual contexto da sociedade de informação, milhares de documentos emergem diariamente na Web, num espectro que começa com os simples comentários em *blogs* e redes sociais e termina com as elaboradas obras técnicas, científicas e literárias, passando necessariamente pelas inúmeras publicações jornalísticas, quer noticiosas, quer de comentário ou crónica. [FC15]
A variedade e riqueza textual disponível ao utilizador comum têm crescido a ritmos exponenciais e de forma ímpar na história da humanidade. Esta nova realidade implica necessariamente o aumento de problemáticas e irregularidades clássicas, no domínio da publicação e autoria, sendo atualmente o plágio a mais preocupante. Por exemplo, no domínio da publicação académica e científica, esta é uma preocupação que está na ordem do dia com muitas situações detetadas de trabalhos plagiados, total ou parcialmente, havendo até algumas situações mediáticas recentes. Várias universidades já adotaram formalmente códigos deontológicos relativamente a esta matéria, com o intuito de *bem* formar os seus alunos. Todavia, as prevaricações continuam a acontecer. [FC15]
Neste contínuo fluir de novos documentos, propelido pelas *tecnologias de informação e comunicação* (TIC), a tarefa de detetar plágio de forma manual converge rapidamente para a impossibilidade absoluta ou pelo menos para o demasiado improvável. Para combater este *modernizado* antigo crime, torna-se fulcral fazer uso das mesmas TICs. Deste modo, a deteção de plágio externo é focada na análise de documentos suspeitos, procurando excertos de texto copiados de documentos fonte, sem o devido reconhecimento e/ou autorização do autor. Esta análise, designada por *deteção de plágio externo*, recorre a semelhanças léxico-gramaticais (lexical, sintática, semântica e estrutural) entre excertos frásicos do documento suspeito e uma coleção de documentos fonte (*corpus* fonte) [ASA12]. Entretanto, o tamanho do *corpus* fonte pode variar de uma centena de documentos até à totalidade dos documentos contidos na *Web*. [FC15]

1.2 Objetivos

Surge, portanto, a necessidade de reduzir o tamanho do *corpus* fonte para um conjunto menor de documentos (candidatos) com o mesmo género de informação do documento suspeito. O passo seguinte reside na análise frásica da informação partilhada entre os pares (documento suspeito *versus* documentos candidatos). [FC15]
Esta análise, detalhada e exaustiva, depende de uma métrica capaz de reconhecer as cópias ilícitas de informação. Surge, portanto, a necessidade de uma métrica robusta para detetar quer o plágio literal de palavras, frases e parágrafos, quer o plágio modificado com a reordenação das palavras mais relevantes, quer ainda o chamado *plágio ofuscado* que além da reordenação substitui palavras relevantes por seus sinónimos. [FC15]

1.3 Contribuições

Assim, neste trabalho aborda-se a eficiência na redução do *corpus* fonte para um pequeno conjunto de documentos candidatos, e a eficácia na deteção dos excertos de plágio com uma métrica robusta. Considerando que os atuais motores de pesquisa são eficientes na devolução das páginas *Web* pesquisadas, através de pesquisa com palavras-chave, então o mesmo pode ser aplicado para a procura de documentos candidatos, através da extração de palavras informativas (*chave de pesquisa*) do documento suspeito e sua pesquisa no *corpus* fonte. Se os casos de plágio confirmado seguem padrões baseados no comportamento humano então será possível definir uma métrica, baseada nesses padrões, com o auxílio da inteligência computacional [Eng07]. [FC15]

1.4 Estrutura da Dissertação

Este documento está organizado da seguinte forma:

- **no Capítulo** 2 apresenta-se uma breve revisão das principais estratégias de *recuperação fonte* e de *análise detalhada* na deteção de plágio externo;
- **no Capítulo** 3 apresenta-se o método de recuperação fonte proposto para a redução do espaço de pesquisa no universo de documentos fonte;
- **no Capítulo** 4 apresenta-se o método de análise detalhada proposto para a deteção de plágio no documento suspeito;
- **no Capítulo** 5 apresentam-se as principais conclusões obtidas, assim como, apresentam-se as propostas de trabalho futuro.

Capítulo 2

Estado da Arte na Deteção de Plágio Externo

Potthast *et al.* [PSE+09, PBE+10, PEB+11, PGH+12, PGH+13, PHB+14] definem um caso de plágio $s = \langle s_{plg}, d_{plg}, s_{src}, d_{src} \rangle$ como sendo constituído por um excerto com plágio s_{plg}, no documento com plágio d_{plg}, obtido sem o devido reconhecimento de um excerto fonte plagiado s_{src}, pertencente a um documento fonte plagiado d_{src}; sendo d_{src} pertencente a um *corpus* fonte D com dimensão eventualmente elevada.
Stein *et al.* [SMP07] interrogaram-se sobre a existência de casos de plágio em um documento suspeito e como identifica-los a partir de excertos fonte em um *corpus* fonte. Como resposta, a essa problemática, foi apresentado um *algoritmo* que ditou um *processo de análise de plágio* de um *documento suspeito* em função de um *corpus* fonte [SMP07]. Esse processo, pela sua simplicidade e pela sua generalidade, tornou-se o processo genérico para *deteção de plágio externo*[1] (*i.e.*, "external plagiarism detection") [SMP07, PSE+09, ASA12, HPS15]. Esse processo a deteção de plágio externo é, normalmente, constituída por dois atos principais [SMP07]. Conforme apresentado na Figura 2.1, inspirada em [SMP07, PSE+09, ASA12, PGH+12, PGH+13, PHB+14, HPS15], esses dois atos principais são, nomeadamente, a *recuperação fonte* e a *análise detalhada* [SMP07, PGH+12].

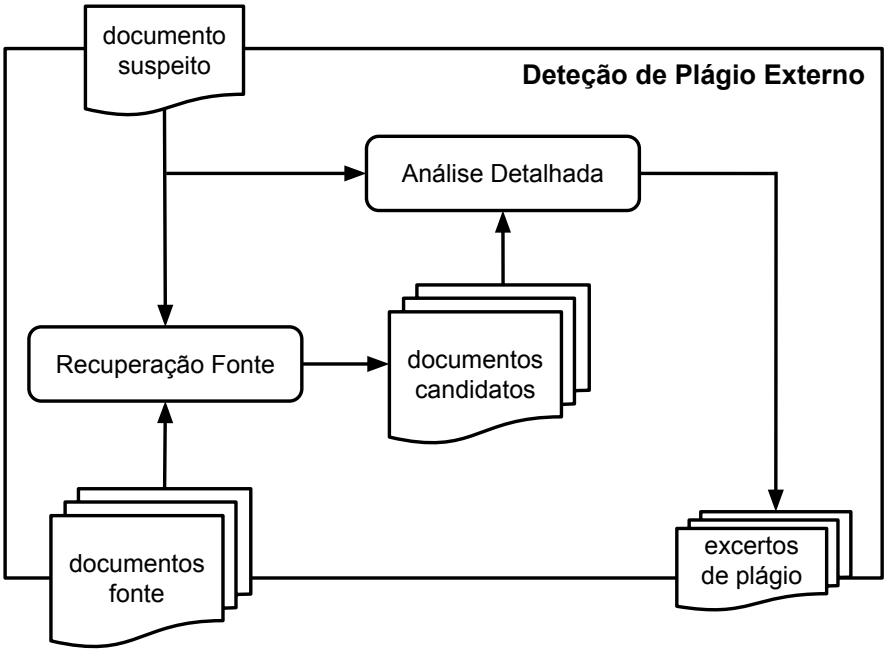

Figura 2.1: Diagrama do processo de deteção de plágio externo, inspirado em [SMP07, PSE+09, ASA12, PGH+12, PGH+13, PHB+14, HPS15].

Assim, Potthast *et al.* [PGH+12] descrevem a deteção de plágio externo como o processo no qual o plágio no d_{plg} é detetado através da pesquisa por excertos fonte, pertencentes ao D, que são extremamente similares aos excertos do d_{plg}. Assim como, esses Autores dividem esse processo

[1] Também conhecido como sistema de *deteção de plágio extrínseco* [ASA12].

em três atos básicos [PGH+12]; sendo dois, desses atos, os principais, nomeadamente, a *recuperação fonte* e a *análise detalhada*, apresentados nas Secções 2.1 e Secções 2.2, respetivamente [PGH+12].

2.1 Estratégias de Recuperação Fonte

Potthast *et al.* [PGH+13, PHB+14], Stamatatos *et al.* [SPR+15] e Hagen *et al.* [HPS15] intitulam o primeiro ato por *recuperação fonte* (*i.e.*, "source retrieval"); sendo, esse ato, também conhecido como: *recuperação heurística* (*i.e.*, "heuristic retrieval") [SMP07, PSE+09, ASA12]; *recuperação fonte candidata* (*i.e.*, candidate source retrieval) [PBE+10, PEB+11]; ou *recuperação candidata* (*i.e.*, "candidate retrieval") [PGH+12].
Potthast *et al.* [PGH+12, PGH+13, PHB+14] e Hagen *et al.* [HPS15] descrevem o primeiro ato como uma tarefa de identificação de um conjunto de documentos fonte D_{src}, candidatos a terem sido as fontes de plágio do d_{plg}, entre os documentos do D, de tal forma que $D_{src} \subseteq D$. Esses Autores, também, revelam uma série de métodos essenciais, que são estrategicamente utilizados, para concluir essa tarefa em conjunção com um motor de pesquisa. Esses métodos são, nomeadamente, o método de *segmentação* ("chunking"), o método de *extração de frases-chave* ("keyphrase extraction") e o método de *formulação de chave de pesquisa* ("query formulation") [PGH+12, PGH+13, PHB+14, HPS15]; apresentados a seguir.
No método de *segmentação* um documento suspeito é dividido em segmentos que, podem se sobrepor, representando as zonas suspeitas de análise. Cada segmento pode ter a dimensão de uma palavra, uma sequência de palavras, uma frase, uma sequência de frases, um linha, uma sequência de linhas, um parágrafos, uma sequência de parágrafos, ..., ou, até mesmo, ser um segmento com a dimensão do próprio documento [PGH+12, PGH+13, PHB+14, HPS15]. No método de *extração de frases-chave* um segmento é utilizado para selecionar e extrair um conjunto frases-chave que, ao ser utilizado para formular uma ou mais chaves de pesquisa, maximize as possibilidades de encontrar a maior quantidade possível de documentos fonte que correspondam ao documento suspeito [PGH+12, PGH+13, PHB+14, HPS15]. No método de *formulação de chave de pesquisa* um segmento é utilizado para extrair um conjunto de palavras-chave que, ao ser utilizado para formular uma ou mais chaves de pesquisa, complacentes com as *limitações*[2] de pesquisa, maximize as possibilidades de encontrar a maior quantidade possível de documentos fonte que estejam em correspondência de plágio com o documento suspeito [PGH+12, PGH+13, PHB+14, HPS15].
Em 2012, Suchomel *et al.* [SKB12] apresentam uma metodologia de *recuperação fonte* que, inicialmente no método de *segmentação*, divide o d_{plg} em segmentos com 45 palavras cada, e com uma sobreposição de 40 palavras entre segmentos adjacentes [PGH+12]. A seguir, cada segmento foi avaliado pela tamanho do seu vocabulário em função dos vocabulários dos segmentos adjacentes [PGH+12]. Por sua vez, os segmentos que se destacaram foram reavaliados pela quantidade de palavras invulgares na primeira frase de cada segmento, segundo um limiar mínimo de 8 palavras invulgares [SKB12, PGH+12]. Posteriormente, extraíram da primeira frase de cada segmento as 6 primeiras palavras invulgares para formar uma chave de pesquisa [SKB12, PGH+12]. Finalmente, as chaves de pesquisa formuladas foram submetidas a um motor de pesquisa, com D indexado, para a obtenção do D_{src} [SKB12, PGH+12]. Jayapal [Jay12], no mesmo ano, propôs uma metodologia de *recuperação fonte*, baseada na divisão do d_{plg} em segmentos de quatro frases [PGH+12]. Posteriormente, no método de *extração de frases-chave*,

[2] *E.g.*, o termo de termos aceite por um motor de pesquisa. [PGH+12]

foi extraído de cada segmento de d_{plg} um frase-chave com as 10 primeiras palavras, constituídas pelos primeiros substantivos, pronomes, verbos e/ou adjetivos do segmento, para ser utilizada como chave de pesquisa [Jay12, PGH$^+$12]. E por fim, as chaves de pesquisa foram submetidas ao motor de pesquisa para obtenção do D_{src} [Jay12, PGH$^+$12].

Williams *et al*. [WCCG13], em 2013, apresentam a sua estratégia de *recuperação fonte* [PGH$^+$13]. Na qual, primeiro, o d_{plg} foi segmentado em grupos de frases, com 5 frases por segmento [PGH$^+$13]. Segundo, esses Autores, para cada segmento utilizaram as três primeiras sequências disjuntas de 10 palavras, constituídas por substantivos, adjetivos e/ou verbos, para extrair as frases-chave que foram posteriormente utilizadas para formular as chaves de pesquisa [WCCG13, PGH$^+$13]. Por último, submeteram essas chaves de pesquisa ao motor de pesquisa para obtenção do D_{src} [WCCG13, PGH$^+$13]. Haggag e El-Beltagy [HEB13], no mesmo ano, propuseram uma metodologia semelhante para a *recuperação fonte*. Na qual, utilizou o *TextTiling* [Hea97] para dividir o d_{plg} em excertos com diferentes tópicos; cada excerto e respetivo tópico foram posteriormente divididos em segmentos com 4 frases [PGH$^+$13]. Estes Autores, a seguir, utilizaram o *KPMiner* [EBR09] para extrair uma frase-chave, por excerto, que foi combinada com a palavra menos frequente de cada segmento do excerto, até possuir 10 palavras; que foi utilizada para formular uma chave de pesquisa por excerto [HEB13, PGH$^+$13]. No fim, submeteram as chaves de pesquisa formuladas, a um motor de pesquisa com *D* indexado, para obtenção do D_{src} [HEB13, PGH$^+$13].

No ano seguinte, 2014, Williams *et al*. [WCG14] propuseram uma metodologia, semelhante à de 2013, para a *recuperação fonte*. Inicialmente, dividiram o d_{plg} em segmentos com cinco frases cada [PHB$^+$14]. Posteriormente, extraíram de cada segmento três frases-chave com, respetivamente, as três primeiras sequências disjuntas de palavra 10-gramas do segmento, sendo as palavras dessas sequências constituídas por apenas substantivos, adjetivos e/ou verbos; essas frases-chave foram utilizadas para formular três chaves de pesquisa por segmento [WCG14, PHB$^+$14]. E finalmente, foram submetidas as chaves de pesquisa do d_{plg} ao motor de pesquisa para a obtenção do D_{src} [WCG14, PHB$^+$14]. No mesmo ano, 2014, Zubarev e Sochenkov [ZS14] apresenta uma estratégia de *recuperação fonte* baseada na segmentação do d_{plg} em parágrafos, com no máximo 50 palavras em cada [PHB$^+$14]. A seguir, cada segmento foi avaliado pelas frequências de suas palavras, e pelo número ocorrências de substantivos, verbos e/ou adjetivos; utilizando apenas os melhores segmentos, segundo um limiar mínimo de frequências e um limite mínimo e máximo de ocorrências [ZS14, PHB$^+$14]. Mais adiante, para cada segmento dos melhores segmentos, removeram os artigos, pronomes e preposições, e as palavras repetidas [ZS14, PHB$^+$14]. E com as restantes palavras de cada segmento criaram uma chave de pesquisa [ZS14, PHB$^+$14]. Essas chaves de pesquisa foram posteriormente submetidas ao motor de pesquisa para obtenção do D_{src} [ZS14, PHB$^+$14].

Em 2015, Ravi N e Gupta [RG15] propuseram uma metodologia de *recuperação fonte* que, inicialmente, segmentou o d_{plg} em parágrafos [HPS15]. Seguidamente, para cada segmento extraiu um conjunto de palavras; constituído apenas por substantivos, adjetivos e verbos; ordenado com os melhores valores de *TF-IDF* de cada palavra [RG15, HPS15]. Posteriormente, cada conjunto de palavras foi dividido para a formulação de duas chaves de pesquisa do segmento [RG15, HPS15]. Finalmente, as chaves de pesquisa obtidas de cada segmento foram submetidas ao motor de pesquisa para a obtenção do D_{src} [RG15, HPS15]. No mesmo ano, Kong *et al*. [KLH$^+$15] propuseram uma metodologia de *recuperação fonte* que, primeiro, segmentou d_{plg} em frases, sendo um segmento correspondente a uma frase [HPS15]. Segundo, para cada segmento, esses Autores, extraíram um conjunto de palavras, constituído apenas por substantivos e verbos, ordenado com as 10 palavras mais frequentes [KLH$^+$15, HPS15]. Terceiro, foi formulada uma chave de

pesquisa com o conjunto de palavras de cada segmento[KLH+15, HPS15]. Por último, as chaves de pesquisa obtidas de cada segmento foram posteriormente submetidas ao motor de pesquisa para a obtenção do D_{src} [KLH+15, HPS15].

2.2 Estratégias de Análise Detalhada

Stein *et al.* [SMP07], Potthast *et al.* [PSE+09] e Alzahrani *et al.* [ASA12] intitulam o segundo ato por *análise detalhada* (*i.e.*, "detailed analysis"); sendo, esse ato também conhecido como: *comparação detalhada* (*i.e.*, "detailed comparison") [PBE+10, PEB+11, PGH+12]; ou *alinhamento textual* (*i.e.*, "text alignment") [PGH+13, PHB+14, SPR+15, HPS15].
Potthast *et al.* [PBE+10, PGH+12, PGH+13, PHB+14] descrevem o segundo ato como uma tarefa de identificação dos excertos similares entre o documento d_{plg} e cada documento fonte d_{src} do D_{src}. Assim como, esses Autores, revelam uma série de métodos essenciais que são estrategicamente utilizados para concluir essa tarefa, nomeadamente, o método de *análise de correspondências* ("seeding") e o método de *junção de correspondências* ("match merging") [PBE+10, PGH+12, PGH+13, PHB+14], apresentados a seguir.
No método de *análise de correspondência* (*i.e.*, "seeding" [PBE+10, PGH+12, PGH+13, PHB+14]) um par de documentos, constituído pelo documento suspeito d_{plg} e pelo documento fonte d_{src}, é dividido em segmentos, normalmente iguais para os dois documentos, que podem sobrepor-se em cada documento, representantes de zonas de comparação entre o par. Cada segmento pode ter a dimensão de uma palavra, uma sequência de palavras, uma frase, uma sequência de frases, uma sequência de caracteres (*i.e.*, um linha ou uma sequência de linhas), ..., ou, até mesmo, ser um segmento com a dimensão do próprio documento. Seguidamente, cada par de segmentos, um do documento suspeito e outro do documento fonte, é utilizado para encontrar uma ou mais correspondências que, são formadas por sub-segmentos que podem variar em tamanho (desde um caracter até ao tamanho do próprio segmento); ou para criar uma ou mais correspondências que, através da *manipulação*[3] dos sub-segmentos; sirvam para identificar similaridades entre cada par de segmentos, para todos os pares de segmentos, do par de documentos. No método de *junção de correspondências* (*i.e.*, "Match Merging" [PBE+10, PGH+12, PGH+13, PHB+14]) os "sub-segmentos correspondentes", de "um par de segmentos" (um do documento suspeito e outro do documento fonte), são unidos como "os sub-segmentos correspondentes" do par de segmentos, anteriormente e/ou posteriormente adjacente, caso existam segmentos adjacentes, e se nesses existam "sub-segmentos correspondentes", que possam ser unidos de modo a criar um alinhamento entre "sub-segmentos correspondentes" entre vários segmentos, que possa ser caso de plágio. Assim, a impossibilidade de união entre os "sub-segmentos correspondentes" de "segmentos adjacentes", implica que os "sub-segmentos correspondentes entre um par de segmentos" são objeto do acaso e não do plágio em si. [PBE+10, PGH+12, PGH+13, PHB+14]
Em 2012, Kong *et al.* [KQW+12] propuseram uma metodologia de *análise detalhada* que, inicialmente, segmentou o par de documentos, d_{plg} e d_{src}, nas suas repetitivas frases [PGH+12]. Seguidamente, analisa as correspondências entre os pares de segmentos (*i.e.*, um de d_{plg} e outro de d_{src}) utilizando um limiar mínimo de similaridade dado pela métrica do *coeficiente do cosseno* e um limiar de similaridade dado por uma métrica baseada no *coeficiente de Jaccard* [KQW+12, PGH+12]. Finalmente, identifica os excertos similares através da junção de correspondências dos pares de segmentos com um algoritmo, proprietário, de ordenação bilateral alternada (*i.e.*, "Bilateral Alternating Sorting Algorithm") [KQW+12, PGH+12]. Suchomel *et*

[3] *E.g.*, adição, remoção, reordenação, transformação, *etc*.

al. [SKB12], no mesmo ano, propuseram uma metodologia que inicialmente, divide o d_{plg} e o d_{src} em segmentos ordenados de palavra 5-gramas e em segmentos de palavra vulgar 8-gramas (*i.e.*, "stop word 8-grams") [PGH$^+$12, Sta11]. Seguidamente, identifica as correspondências entre os pares de segmentos (*i.e.*, um de d_{plg} e outro de d_{src}) através dos pares de segmentos iguais [SKB12, PGH$^+$12]. Identificando os excertos similares através de duas junções de correspondências, entre as correspondências adjacentes, no par de documentos [SKB12, PGH$^+$12]. Consequentemente, na primeira junção, cada correspondência (*i.e.*, segmento igual no par) foi unida com uma correspondência adjacente, se a distância entre as duas for menor ou igual que 4000 caracteres; criando assim uma "nova correspondência" (um segmento maior) constituída por correspondências (segmentos iguais) [SKB12, PGH$^+$12]. Finalmente, na última junção, cada correspondência foi unida com uma correspondência adjacente, se a distância entre elas for menor que 30000 caracteres, se o número de caracteres nelas for maior que duas vezes distância em caracteres entre elas, e se rácio do número de correspondências originais (*i.e.*, "segmentos originalmente iguais") pelo número de caracteres, nelas, permanecer semelhantes entre elas; identificando assim um par de excertos similar [SKB12, PGH$^+$12].

Torrejón e Ramos [RM13], em 2013, apresentaram uma estratégia para a *análise detalhada* que, inicialmente, divide o d_{plg} e o d_{src} em segmentos ordenados baseados em palavras 3-gramas [PGH$^+$13]. Seguidamente, identificam as correspondências com base nos pares de segmentos iguais entre o d_{plg} e o d_{src} [RM13, PGH$^+$13]. Posteriormente, esses autores, identificam os excertos similares, entre d_{plg} e d_{src}, através da junção das correspondências mais frequentes com as respetivas correspondências adjacentes [RM13, PGH$^+$13]. No mesmo ano, Kong *et al.* [KQD$^+$13] propuseram uma metodologia de *análise detalhada* que, inicialmente, divide o d_{plg} e o d_{src} em segmentos frásicos, sendo os segmentos correspondentes das frases dos documentos [PGH$^+$13]. Posteriormente, identificam as correspondências, entre segmentos, utilizando uma métrica baseada no *coeficiente de Jaccard* e um limiar mínimo de similaridade aplicado aos resultados obtidos com essa métrica [KQD$^+$13, PGH$^+$13]. Por fim, fizeram a junção de correspondências adjacentes, e a partir dessas identificaram os excertos similares entre o par de documentos [KQD$^+$13, PGH$^+$13].

Em 2014, Sanchez-Perez *et al.* [SPSG14] apresentaram uma abordagem de *análise detalhada* que, inicialmente, divide o d_{plg} e o d_{src} em segmentos frásicos, correspondentes às frases dos documentos; sendo os segmentos menores (com menos de quatro palavras) unidos com os segmentos seguintes [PHB$^+$14]. Seguidamente, identifica as correspondências frásicas, entre os pares de segmentos, utilizando uma métrica baseada no *coeficiente do cosseno* e no *coeficiente de Dice*, e um limiar mínimo de similaridade frásica aplicado aos valores obtidos com essas métricas [SPSG14, PHB$^+$14]. Finalmente, identifica os excertos similares, entre o par de documentos, utilizando a união de correspondências frásicas adjacentes e a junção, a essas, de correspondências frásicas próximas segundo um limiar mínimo e máximo de segmentos (frases) não correspondentes [SPSG14, PHB$^+$14]. Também em 2014, Palkovskii e Belov [PB14] propuseram uma metodologia que, primeiro, divide o d_{plg} e o d_{src} em múltiplos segmentos baseados em variações de palavra n-gramas (*e.g.*, palavra ordenada ("*alphasorting*") n-gramas, palavra frequente ("*stop-word*") n-gramas, radical ("*stemming*") n-grama, *etc* [PB14]); não sendo especificado o número de palavras *n* das sequências [PHB$^+$14]. Segundo, identifica as correspondências entre os pares de segmentos através da análise dos pares de segmentos iguais [PB14, PHB$^+$14]. Por último, identifica os excertos similares entre d_{plg} e no d_{src} através de duas junções de correspondências com um algoritmo de agregação gráfica de elipse angular (*i.e.*, "*angled ellipse based graphical clustering algorithm*") [PB14, PHB$^+$14].

2.3 Sumário

Nesse capítulo foram descritas as principais estratégias de deteção de plágio externo apresentadas, por investigadores e profissionais oriundos da comunidade científica e industrial, e avaliadas no âmbito da quarta, quinta, sexta e sétima *Competição Internacional de Deteção de Plágio* [PGH+12, PGH+13, PHB+14, HPS15] organizadas pela PAN[4].
Da "viagem bibliográfica" empreendida pode constatar-se que a grande maioria das abordagens, quer de geração de chave de pesquisa quer de análise detalhada, estão dependentes de certas partições dos documentos, fixadas previamente e de forma arbitrária. Na abordagem proposta nos capítulos seguintes evitou-se ao máximo a predefinição de tamanhos de segmentos, deixando que o sistema se ajuste dinamicamente às particularidades e dimensões específicas do texto a tratar.

[4]Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse: http://pan.webis.de/

Capítulo 3

Método de Recuperação Fonte Proposto

Este capítulo aborda o método de recuperação fonte proposto para a redução do espaço de pesquisa no universo de documentos fonte. Primeiro, introduz-se os principais conceitos associados e a problemática abordada, a seguir, define-se a abordagem adotada para a resolução do problema em causa e a notação utilizada. Segundo, apresenta-se o método proposto, para a resolução do problema, dividido nos três principais "momentos", intitulados por: *indexação fonte*; *formulação da chave de pesquisa*; e *pesquisa e filtragem da fonte*. Finalmente, descrevem-se as experiências realizadas e os resultados obtidos. Para tal, foi dado especial ênfase aos conjuntos de dados utilizados, às ferramentas adotadas e aos parâmetros utilizados na implementação, assim como, aos métodos de avaliação adotados e aos, respetivos, resultados experimentais obtidos do método proposto.

A recuperação fonte, abordada neste capítulo, constitui o primeiro ato na deteção de plágio externo. Este facto revela um elevado grau de importância no processo de deteção de plágio. Isso porque o seu objetivo consiste em identificar um subconjunto de documentos fonte similares ao documento suspeito, ou seja, possíveis fontes de plágio para o documento suspeito. Este subconjunto de documentos fonte designa-se por **documentos candidatos**. Uma vez que o conjunto de documentos candidatos são identificados inicia-se o segundo ato da deteção de plágio externo, nomeadamente, a análise detalhada entre os pares (documento suspeito *versus* documentos candidatos), descrito em detalhe no Capítulo 4, conforme apresentado no diagrama da Figura 2.1.

A identificação do conjunto de documentos candidatos introduz a problemática da redução do espaço de procura. Uma vez que o número de documentos fonte pode variar de uma dezena de documentos até grandes *corpora*, ou, até mesmo, ao universo de documentos contido na *Web*. Depara-se portanto com um problema do género do de *Pesquisa de Informação*[1] [CC04, Sin01], em que a partir de um excerto de texto, sobre um determinado assunto, é necessário identificar quais os documentos relacionados com o mesmo assunto. Consequentemente, conjetura-se que se os atuais motores de pesquisa são eficientes na devolução das páginas *Web* pesquisadas, através de pesquisa com palavras-chave, então o mesmo pode ser aplicado na procura por documentos candidatos, através da extração de palavras informativas (*chave de pesquisa*) do documento suspeito e, posterior, pesquisa nos documentos fonte (*corpus fonte*) previamente indexados, procurando assim por documentos semelhantes (*documentos candidatos*).

A Figura 3.1 apresenta o diagrama do processo de recuperação fonte adotado para a resolução do problema em causa, descrito anteriormente, através da utilização de um motor de pesquisa. Primeiro, utiliza-se a funcionalidade de indexação de um motor de pesquisa para indexar o *corpus* fonte, reduzindo assim o espaço de procura por documentos fonte, ver Secção 3.1. Segundo, formula-se a chave de pesquisa a partir das palavras mais informativas do documento suspeito, ver Secção 3.2. Finalmente, utiliza-se a funcionalidade de pesquisa de um motor de pesquisa para encontrar os melhores (seleção/filtragem) documentos candidatos, devolvidos, ordenadamente, pelo motor de pesquisa, ver Secção 3.3.

[1] Information retrieval.

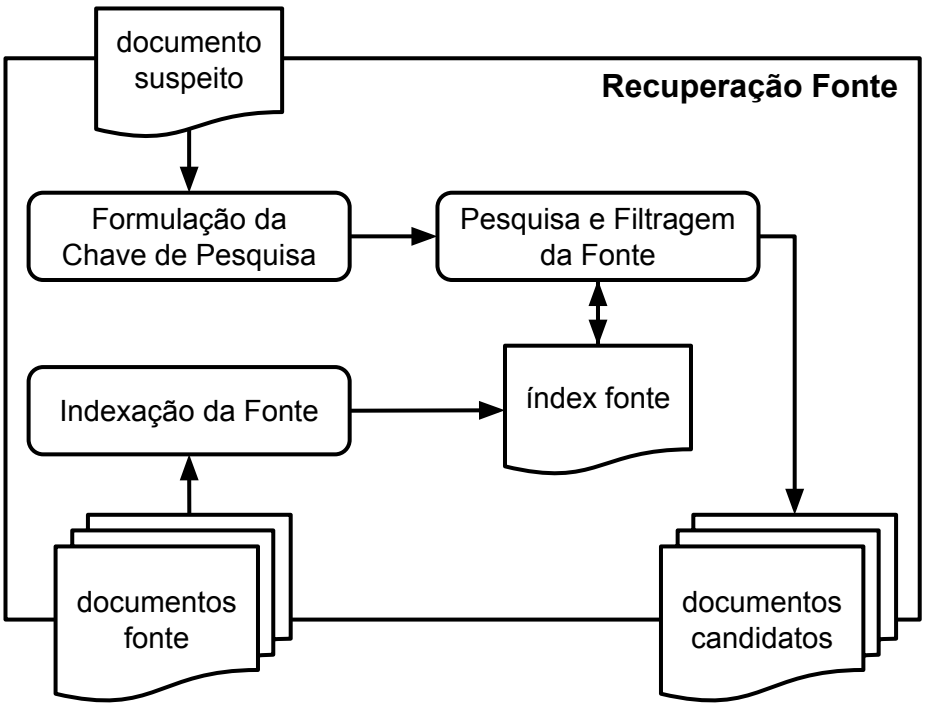

Figura 3.1: Diagrama do processo de recuperação fonte.

Para o método de recuperação fonte proposto, designa-se um documento "genérico" d como sendo constituído por uma sequência de m frases, $d = \{s_1, s_2, ..., s_m\}$, sendo, por sua vez, cada frase s composta por uma sequência de n palavras $s = \{w_1, w_2, ..., w_n\}$; assim considerou-se a palavra como sendo a unidade linguística mínima de análise. Representa-se um *documento suspeito* de plágio pela letra delta (δ) e um *documento fonte* por d, isto é, um documento a partir do qual poderá ter sido alvo de plágio. Tal como introduzido na Secção 2.1, dado um documento suspeito δ, pode-se designar $\Sigma_{(\delta)} = \{d_1, d_2, ..., d_n\}$ o conjunto de *documentos candidatos*, do universo de documentos fonte (*corpus* fonte D_{font}), relativos ao δ, de onde o infrator possivelmente obteve as passagens plagiadas. Assim, um caso de plágio (ou referência r_a) é um alinhamento entre dois excertos de texto, um de δ e outro de um $d \in \Sigma_{(\delta)}$, representado por $r_a = \langle \delta^{a:A}, d^{b:B} \rangle$, sendo a:A o início (a) e o fim (A) de um excerto de texto de δ, assim como, b:B o início (b) e o fim (B) de um excerto de texto de d. Para simplificar a notação e abstrair dos detalhes dos limites dos excertos, representa-se, sempre que se torne conveniente, o par de excertos de texto por $r_a = \langle s_a, s_b \rangle$, com $s_a \in \delta$ e $s_b \in d$. Designa-se um *corpus* de casos de plágio como sendo composto por um conjunto de m documentos suspeitos $D_{sus} = \{\delta_1, \delta_2, ..., \delta_m\}$, contendo documentos suspeitos com plágio confirmado (documentos *black*) e documentos originais sem nenhum conteúdo com plágio (documentos *white*). Portanto, tem-se $f(\delta_i) \in \{black, white\}$, com $f(.)$ a função de *avaliação* de um documento. Para além de D_{sus}, designa-se o conjunto de m documentos fonte por $D_{font} = \{d_1, d_2, ..., d_m\}$. Consequentemente um documento suspeito δ com plágio confirmado $f(\delta_i) = black$ terá uma ou mais referências de plágio: $refs_{(\delta)} = \{r_1, r_2, ..., r_p\}$, sendo $r_a = \langle \delta^{a:A}, d^{b:B} \rangle$, ou na forma simplificada $r_a = \langle s_a, s_b \rangle$.

3.1 Indexação da Fonte

Conforme descrito anteriormente, propõe-se a utilização das funcionalidades nativas de um motor de pesquisa para a resolução da problemática da redução do espaço de pesquisa na deteção de plágio externo. Para tal, o primeiro "momento" deste processo consiste na indexação do *corpus* fonte (D_{font}), possibilitando uma utilização eficiente do espaço de memória do computador, assim como, permitindo pesquisas rápidas da informação dos documentos [MHG10].
O método de indexação utilizado, inerente às bibliotecas de indexação do motor de pesquisa adotado [MHG10], baseia-se em uma estrutura de dados com o *índice invertido*[2] para mapear um conjunto de dados[3], no(s), respetivo(s), endereço(s) de ocorrência[4] [CP90]. Ao aplicar a indexação ao *corpus* fonte, obtém-se uma estrutura de dados composta pelo vocabulário de palavras do D_{font} em função do(s) nome(s) do(s) documento(s) fonte, em que cada palavra do vocabulário ocorre [Mit14]. Assim, para o método proposto, a aplicação do método de indexação do D_{font} segue, genericamente, os trâmites descrito no Algoritmo 3.1.

Algoritmo 3.1: Algoritmo de indexação da fonte.

```
1: index_fonte ← ∅
2: for i ← 1 to |D_font| do
3:     if d_i ∈ D_font then
4:         index_fonte ← index_fonte ∪ {indexacao(d_i)}
5: return index_fonte
```

Neste, os parâmetros de entrada (requisitos) são todos documentos fonte e respetivos dados textuais, todavia, o parâmetro de saída (objetivo) consiste em um ficheiro com a representação compacta desses dados, ou seja um índice entre os dados (vocabulário global) e sua origem (nomes dos documentos em que os dados ocorrem).
Ao fim da indexação da fonte é possível passar ao próximo "momento" do processo de recuperação fonte, Secção 3.2. Todavia, designe-se aqui a indexação do D_{font}, sob a forma do *índice fonte*, e a sua elegibilidade, em ser alvo de pesquisas, por *motor de plágio* (MP).

3.2 Formulação da Chave de Pesquisa

A formulação da chave de pesquisa é o segundo "momento" do método de recuperação fonte proposto, ver Figura 3.1. A seguir à indexação da fonte, ver Secção 3.1, foi possível utilizar as funcionalidades nativas de pesquisa, do motor de pesquisa, através da submissão de um conjunto de palavras extraídas do *documento candidato* δ, formando assim uma *chave de pesquisa*, para o *motor de plágio* MP e, assim, obter um conjunto de *documentos candidatos* $\Sigma_{(\delta)}$ relevantes, relacionados com a pesquisa, de entre o universo de *documentos fonte* D_{font} previamente indexados.
Contudo, este processo não é propriamente linear, havendo alguns desafios que aqui se colocam e que estão relacionados com o volume e a natureza dos dados textuais de δ. Primeiro, é preciso ter em conta a volatilidade de δ que pode variar de *uma dezena de palavras* até mesmo às *centenas de milhares de palavras*, como por exemplo se o δ contiver, respetivamente, uma notícia ou uma enciclopédia. Por último, é preciso ter em conta a heterogeneidade de δ, uma

[2] Inverted index.
[3] Por exemplo: palavras, números, datas, etc.
[4] Por exemplo: nome do(s) documento(s).

vez que a representação do conteúdo textual pode variar segundo *diferentes tipos de assuntos*[5] abordados nas *diferentes partes*[6] do documento. Surge assim a necessidade de formular uma chave de pesquisa abrangente e compacta sem perder a representatividade da informação contida no documento δ.

De modo a ultrapassar os desafios, supracitados, conjeturou-se que apenas as palavras mais relevantes do vocabulário de δ serviriam de base para a *chave de pesquisa*, reduzindo assim as palavras relevantes no vocabulário do documento. Uma primeira abordagem ao cálculo da relevância das palavras pode ser implementada com base no número de vezes que cada palavra do vocabulário ocorre no documento δ em função do número de ocorrências no D_{font}. Estas frequências, local no δ e global no D_{font}, são as bases da métrica de *Frequência de Termos e Frequência Inversa Documental* (TF-IDF)[7], introduzida e abordada por Salton [SYY75, SB88], para calcular a relevância das palavras num determinado documento, em função da distribuição das palavras num *corpus* geral *D* [SYY75, SB88, Ven14, Fer14]. Nesta métrica, apresentada na Equação (3.1):

$$TF\text{-}IDF(w) = \frac{F(w \mid \delta)}{|\delta|} \times \log\left(\frac{|D|}{F(w \mid D)}\right) \quad (3.1)$$

o w representa a palavra (ou termo, no caso de ser uma data ou um número), o $F(w \mid \delta)$ representa a frequência relativa de w no δ, o $|\delta|$ representa o número de palavras no δ, o $|D|$ representa o número de documentos no *corpus* D e o $F(w \mid D)$ representa o número de documentos onde w ocorre. Com a métrica da Equação (3.1) é possível filtrar os termos de um documento, penalizando as palavras homogeneamente frequentes (*i.e.*, os *termos vulgares*[8]), como por exemplo "de", "um", "para", "que" , etc; assim como, permite beneficiar as *palavras incomuns e menos frequentes* (*i.e.*, os termos invulgares), mas que são relevantes na coleção de documentos. Não o bastante, resta ultrapassar o desafio da heterogeneidade, para tal, conjeturou-se que a noção de relevância de uma palavra w, do vocabulário do δ, pode ser relacionada com a informação que esta veicula, na medida em que as palavras com morfologia mais complexas possuem uma maior quantidade de informação, devido ao facto de possuírem um maior número de caracteres, originários de *múltiplos radicais*[9] que veiculam significados mais complexos. Para identificar e beneficiar tais palavras, propõe-se uma nova utilização para a métrica, intitulada, *Função de Importância de Termos* (FIT) [Men13]. Esta métrica, inicialmente, inspirada no TF-IDF de Salton [SYY75, SB88], e, posteriormente, utilizada para o calculo da similaridade documental [Men13], permite identificar as palavras mais relevantes do vocabulário do δ, tendo em conta o volume e a natureza dos dados textuais, para, a seguir, formular a *chave de pesquisa* que será submetida ao motor de plágio.

Assim, na métrica da Equação (3.2):

$$FIT(w) = |w| \times \log_2\left(\frac{P(w \mid \delta)}{P(w)}\right) \quad (3.2)$$

o $|w|$ representa o número de caracteres da palavra (termo) w; $P(w \mid \delta)$ representa a frequência

[5]Por exemplo: uma revista com notícias, receitas de culinária e anúncios de produtos.

[6]Por exemplo: um relatório de projeto com introdução, trabalho relacionado, métodos e resultados, e conclusão.

[7]TF-IDF: Term Frequency - Inverse Document Frequency.

[8]Stopwords. [Uni15]

[9]Por exemplo: a palavra pneumoultramicroscopicossilicovulcanoconiose designa uma doença pulmonar inflamatória causada pela inalação de partículas de pó de sílica contidas nas cinzas vulcânicas.

relativa de w em δ; e, finalmente, $P(w)$ representa a frequência relativa do mesmo termo w, mas estimado no D_{font}. De uma maneira geral, quanto maior for o número de carateres de uma palavra, maior é a informação que veicula, logo maior será a sua importância. Assim, a métrica *FIT* permite selecionar um conjunto de palavras relevantes e representativas de um documento ou excerto deste. Esta noção de relevância é uma combinação do comprimento da palavra com o logaritmo de um rácio de ocorrências, no documento (δ) e num universo mais lato de documentos (D_{font}).
Assim, a formulação da chave de pesquisa é constituída, inicialmente, pela identificação do vocabulário V do documento δ ($V_{(\delta)}$), seguido do calculo do valor de $FIT(.)$ para cada palavra do $V_{(\delta)}$, e, por fim, identificação das palavras com os melhores valores de $FIT(.)$, tal que $FIT(w_i) > 0$. O método proposto segue os trâmites apresentados no Algoritmo 3.2. No pseudo-código o parâmetro de entrada (requisito) é definido pelo documento suspeito δ e respetivos dados textuais, todavia, o parâmetro de saída (objetivo) consiste em uma estrutura, ordenada, com as melhores palavras do $V_{(\delta)}$.

Algoritmo 3.2: Algoritmo de formulação da chave de pesquisa.

```
1: m ← |δ|
2: V ← ∅
3: for i ← 1 to m do
4:    if w_i ∈ δ ∧ w_i ∉ V then
5:       value ← FIT(w_i)
6:       if value > 0 then
7:          V ← V ∪ {w_i}
8: chave_pesquisa ← V.sort
9: return chave_pesquisa
```

A seguir, na Secção 3.3, é a apresentado o próximo "momento" do método de recuperação fonte proposto. Neste são apresentados a utilização e otimização das funcionalidades de *submissão da pesquisa* e *obtenção de resultados* do motor de pesquisa.

3.3 Pesquisa e Filtragem da Fonte

Esta secção aborda o terceiro "momento" do método de recuperação fonte proposto, conforme descrito no Diagrama 3.1. Este "momento" preocupa-se com a utilização e otimização das funcionalidades de *submissão da pesquisa* e de *obtenção de resultados*, ambas nativas, do motor de pesquisa. Uma vez que o *corpus* de *documentos fonte* (D_{font}) encontra-se *indexado* (*índice fonte*) e *elegível em receber inquirições* (*motor de plágio*), ver Secção 3.1; e dado que o *documento suspeito* (δ) encontra-se analisado, estratificado segundo o seu *vocabulário* $V_{(\delta)}$, e esmiuçado em palavras relevantes (*chave de pesquisa*) segundo os valores obtidos com a *métrica de importância* (*FIT*), ver Secção 3.2; então há condições de avançar para a *pesquisa e filtragem da fonte*.

3.3.1 Pesquisa da Fonte

A pesquisa carateriza-se pelo processo de submissão de *um conjunto de palavras* (chave de pesquisa) às bibliotecas de pesquisa do motor de pesquisa. Estas bibliotecas, por sua vez, interrogam as *páginas* (documentos fonte), *previamente indexadas* (índice fonte) pelas bibliotecas

de indexação. O motor de pesquisa devolve *um conjunto de "resultados"* (documentos candidatos $\Sigma_{(\delta)}$) relevantes segundo as maiores partilhas de palavras [MHG10]. Porém, quanto menor for a chave de pesquisa submetida, menor será a representatividade textual do δ, assim como, menor será a probabilidade de se encontrar o $\Sigma_{(\delta)}$ que fora alvo de plágio. Por outro lado, quanto maior for a chave de pesquisa, submetida, maior será a representatividade do δ, assim como, maior será o volume do $\Sigma_{(\delta)}$ com documentos que *não* foram alvo de plágio, mas partilham parte da chave de pesquisa. Além de, o tamanho máximo da chave de pesquisa está *limitado*[10] a um valor máximo pré-definido pelo motor de pesquisa.
Por estes motivos, conjeturou-se que haveria, idealmente, um rácio entre a extensão da chave de pesquisa utilizada e o volume do $\Sigma_{(\delta)}$ relevante devolvido. Por esta razão *limitou-se a extensão máxima da chave de pesquisa* aos μ termos mais relevantes da chave de pesquisa; salienta-se que a chave de pesquisa fora formulada com as palavras mais relevantes do $V_{(\delta)}$, e que a noção de relevância está, diretamente, relacionada com os melhores valores obtidos da métrica *FIT*, tal que $FIT(w_i) > 0$ ver Secção3.2. Portanto, o limite máximo dos μ termos mais relevantes depende de um limiar mínimo entre uma percentagem fixa dos melhores termos da chave de pesquisa e um número fixo dos melhores termos da chave de pesquisa, conforme definido na Equação (3.3):

$$\mu = \min(a \times |chave\_pesqusia|, \beta) \quad (3.3)$$

onde $a \in [0, 1]$ representa uma constante definida à *priori*, $|chave\_pesqusia|$ representa o número de termos na chave de pesquisa, e β representa uma constante tal que $0 < \beta \leq |chave\_pesqusia|$.
Por exemplo, se um documento suspeito δ possui 4302 termos, seu vocabulário $V_{(\delta)}$ for composto por 1405 termos, e sua chave de pesquisa *chave_pesquisa* composta por 1262 termos relevantes, tal que $FIT(w_i) > 0$ e $w_i \in V_{(\delta)}$; admitindo também que $a = 0.05$ e $\beta = 100$; então o tamanho final da chave de pesquisa será composto por $\mu = 63$ termos, dado que $\mu = \min(0.05 \times 1262, 100)$.
Estes critérios, de seleção dos melhores e mais relevantes termos da chave de pesquisa, quando aplicados (Algoritmo 3.3, da primeira à nona linha), seguem uma "política" que permite ultrapassar as limitações descritas anteriormente, satisfazendo assim o rácio descrito. Com efeito, permite pesquisas (submissão de chaves de pesquisa) consistentes, segundo os padrões de relevâncias adotados, e adaptáveis para documentos extremamente grandes.

3.3.2 Filtragem da Fonte

A filtragem da fonte, como intitulado, carateriza-se pelo processo de filtragem *do conjunto de "resultados"*, devolvido pelas bibliotecas de pesquisa, ditado pelo motor de pesquisa como o conjunto de documentos fonte relacionados com a chave de pesquisa submetida anteriormente (*chave_pesquisa vs. indice_fonte*). Nesta filtragem, conjeturou-se que haveria, idealmente, um subconjunto de documentos fonte, de entre o conjunto de resultados devolvidos, que, segundo a disposição ditada pelo motor de pesquisa, possuem um elevado grau de partilha de termos relevantes entre a *chave_pesquisa* $\subseteq V_{(\delta)}$ submetida e os $d_i \in$ *indice_fonte* devolvidos. Com efeito, este subconjunto de documentos filtrados formaram o conjunto de documentos candidatos $\Sigma_{(\delta)}$ que, muito provavelmente, foram alvo de plágio pelo δ. Porém, conforme susodito,

[10] Para o motor de pesquisa adotado o limite máximo admitido para chave de pesquisa é de 1024 termos [MHG10].

existem documentos fonte, no conjunto inicial de resultados, que não foram plagiados e partilham apenas uma pequena parte, apesar de significativa, dos termos relevantes da chave de pesquisa do δ. Por esta razão *delineou-se um filtro* que segue uma "política" de seleção dos ω documentos candidatos mais relevantes, segundo a disposição de resultados ditada pelo motor de pesquisa.

Algoritmo 3.3: Algoritmo de pesquisa e filtragem da fonte.

```
1: dummy ← chave_pesquisa
2: m ← |dummy|
3: μ ← min(a × m, B)
4: chave_pesquisa ← ∅
5: dummy ← dummy.bestFirst
6: for i ← 1 to μ do
7:     if w_i ∈ dummy then
8:         chave_pesquisa ← chave_pesquisa ∪ {w_i}
9: results ← indice_fonte.pesquisar(chave_pesquisa)
10: results ← results.bestFirst
11: Σ_(δ) ← ∅
12: if ω > |results| then
13:     ω ← |results|
14: for i ← 1 to ω do
15:     if d_i ∈ results then
16:         Σ_(δ) ← Σ_(δ) ∪ {d_i}
17: return Σ_(δ)
```

O Algoritmo 3.3 apresenta, minimalistamente, o pseudo-código para a utilização da Equação (3.3), limitando assim a extensão da chave de pesquisa submetida às bibliotecas de pesquisa do motor de pesquisa. Assim como, apresenta a "política" de filtragem dos documentos fonte, devolvidos pelo motor de pesquisa, para a seleção dos ω documentos candidatos. Neste pseudo-código os parâmetros de entrada (requisitos) são constituídos: primeiro, pela chave de pesquisa (*chave_pesquisa*) previamente definida no Algoritmo 3.2; e, finalmente, pelas constantes a, B, e ω que representam, respetivamente, a percentagem máxima e o número máximo de termos que serão utilizados na pesquisa, e o número máximo de documentos candidatos $\Sigma_{(\delta)}$. Todavia, o parâmetro de saída (objetivo) consiste em um conjunto de documentos candidatos $\Sigma_{(\delta)}$ a terem sido alvo de plágio por δ.

Com a aplicação do algoritmo supracitado e respetiva identificação do subconjunto de documentos candidatos relacionados com o documento suspeito, conclui-se a descrição do último "momento" do método de recuperação fonte proposto (ver Diagrama 3.1), assim como o primeiro ato da deteção de plágio externo (ver Diagrama 2.1). A seguir, na Secção 3.4, descrevem-se as experiências realizadas e os resultados obtidos para o método de recuperação fonte proposto. Mais adiante, no Capítulo 4, aborda-se o segundo ato da deteção de plágio externo, nomeadamente, a análise detalhada entre os pares (documento suspeito *versus* documentos candidatos).

3.4 Experiências e Discussão

Nesta secção abordam-se as experiências realizadas e discutem-se os resultados obtidos no primeiro ato da deteção de plágio externo (ver Diagrama 2.1), nomeadamente para o método de recuperação fonte proposto e os seus momentos (ver Diagrama 3.1). Primeiro, descreve-se o conjunto de dados utilizado, nomeadamente os *corpora de plágio* empregues no treino, no teste,

e na validação do método proposto. Segundo, nomeiam-se as ferramentas adotadas para auxiliar a implementação e validação do método proposto, assim como, definem-se os valores impostos aos parâmetros de configuração utilizados nas diferentes fases de implementação e validação. Seguidamente, apresentam-se os métodos de avaliação [KBLP55, Rij79, PHVS13] utilizados para qualificar e avaliar as experiências realizadas. Por fim, apresentam-se e discutem-se os resultados de avaliação obtidos para o método de recuperação fonte proposto.

3.4.1 Conjunto de Dados

Nesta subsecção aborda-se o *conjunto de dados* utilizados para desenvolver, implementar, avaliar e validar o método de recuperação fonte proposto e os respetivos resultados obtidos nas experiências realizadas. Primeiramente, descreve-se o *conjunto de dados*, sua estruturação em *corpora*, sua divisão em dados de treino, teste e validação, em como os dados são apresentados e descritos. Seguidamente, descrevem-se os elementos constituintes dos corpora. Simultaneamente, descreve-se e quantifica-se a estrutura aglomerativa de documentos, os diferentes géneros de documentos, os diferentes tipos de plágio, e o modo como os documentos são representados e utilizados nos diferentes momentos do método proposto.
A fim de desenvolver, implementar e avaliar o método de recuperação fonte proposto utilizou-se um conjunto de dados composto por dois *corpora* de documentos de texto escritos na língua inglesa. Estes *corpora* foram criados para treinar e testar algoritmos de deteção de plágio externo em duas competições internacionais muito conceituadas, na área da deteção de plágio [PEB+11, PGH+13]. Todavia, no âmbito deste trabalho, dividiu-se estes *corpora* em dois subconjuntos de dados para finalidades distintas. O primeiro subconjunto de dados utilizou-se como dados de treino e teste para desenvolver, implementar e avaliar o método. Este subconjunto de dados é designado por *corpus PAN-PC-11*[11]. O segundo subconjunto de dados utilizou-se como dados de validação para avaliar os resultados do método com uma amostra desconhecida de dados. Este subconjunto de dados designa-se por *corpus PAN13-SourceRetrieval*[12]. Em ambos os *corpora* os dados são apresentados em ficheiros de texto com a extensão *txt*[13]. Conjuntamente, para o primeiro subconjunto de dados, as referências de plágio são descritas em ficheiros com extensão *xml*[14] que elucidam os limites (*i.e.*, início e fim) e os atores (*i.e.*, origem e destino) intervenientes no plágio. Todavia, para o segundo subconjunto de dados, esses dados são apresentados em ficheiros com a extensão *json*[15].
O *corpus PAN-PC-11* foi criado e atualizado para avaliar algoritmos de deteção de plágio intrínseco e externo, sendo utilizado como dados de treino e de teste no âmbito da terceira competição internacional de deteção de plágio da PAN[16] [PSBR10, PEB+11]. Este *corpus* é composto por dois *corpora* menores, um para a deteção de plágio intrínseco (*i.e.*, não recorre a documentos externos para encontrar o plágio) e outra para a deteção de plágio externo (*i.e.*, recorre a documentos externos para encontrar o plágio, ou seja, o foco do método proposto).
O *corpus* de plágio externo é composto por dois conjuntos de documentos de texto: um de *documentos fonte* sem qualquer tipo de plágio, que não se sabe, *a priori*, se foram alvo de plágio; e outro de documentos suspeitos com o conteúdo desconhecido que se suspeita existir plágio.

[11] https://www.uni-weimar.de/en/media/chairs/webis/corpora/pan-pc-11
[12] https://web.archive.org/web/20160901172225/http://www.uni-weimar.de/medien/webis/corpora/corpus-pan-labs-09-today/pan-13/pan13-data/pan13-source-retrieval-test-corpus2-2014-12-01.zip
[13] Plain text file.
[14] Extensible markup language.
[15] JavaScript object notation.
[16] Uncovering plagiarism, authorship and social software misuse.

O conjunto de documentos fonte possui 11053 documentos originais extraídos de livros provenientes do *Projeto Gutenberg*[17][PSBR10]. No método proposto representam-se esses documentos por D_{font} e um único documento por d, sendo esses utilizados na indexação da fonte, ver Secção 3.1. O conjunto de documentos suspeitos possui 10538 documentos suspeitos de possuírem plágio dos documentos fonte. Este conjunto é composto por dois subconjuntos de documentos de texto: um com 5546 documentos suspeitos sem plágio (*i.e.*, apesar de serem suspeitos, não possuem qualquer tipo de plágio); e outro com 4992 documentos suspeitos com plágio simulado. No método proposto representa-se um documento suspeito por δ, assim como, representa-se o conjunto de documentos suspeitos por D_{sus}, o subconjunto de documentos suspeitos sem plágio por D_{white} e o subconjunto de documentos suspeitos com plágio simulado por D_{black}, e utilizam-se na formulação da chave de pesquisa e na pesquisa e filtragem fonte, ver Secção 3.2 e Secção 3.3. Todavia, no subconjunto de documentos suspeitos com plágio simulado existem quatro tipos de simulação de plágio: o primeiro com 106 documentos com plágio simulado manualmente por humanos; o segundo com 114 documentos com plágio simulado artificialmente sem nenhum tipo ofuscação, *i.e.*, os excertos de texto com plágio são iguais aos excertos de texto plagiados; o terceiro com 2370 documentos com plágio simulado artificialmente com pouca ofuscação, *i.e.*, os excertos de texto com plágio são semelhantes aos excertos de texto plagiados; o quarto e último tipo com 2405 documentos suspeitos com plágio simulado artificialmente com elevada ofuscação, *i.e.*, os excertos de texto com plágio são distintos dos excertos de texto plagiados. Nos documentos com plágio artificial as simulações de plágio foram criadas por algoritmos computacionais utilizando técnicas de ofuscação automáticas e aleatórias [PSBR10, PEB$^+$11]. Dois exemplos destas técnicas de ofuscação são: as operações aleatórias de texto, em que um excerto de texto com plágio é criado através da mistura, remoção, inserção ou substituição aleatória das palavras ou pequenas frases de um excerto de texto original [PSBR10]; e as variações semânticas de palavras, em que um excerto de texto com plágio é criado através da substituição das palavras de um excerto de texto original por sinónimos (*i.e.*, palavras com o mesmo significado), antónimos (*i.e.*, palavras com significados opostos), hipónimos (*e.g.*, cereja é hipónimo de fruta) ou hiperónimos (*e.g.*, fruta é um hiperónimo de cereja) escolhidos aleatoriamente [PSBR10]. No método proposto, representa-se um referência de plágio (*i.e.*, o início e o fim de um excerto de texto com plágio do documentos suspeito, e o início e o fim de um excerto de texto plagiado do documento fonte) por $r_a = \langle \delta^{a:A}, d^{b:B} \rangle$, sendo a:A o início (a) e o fim (A) do excerto de texto com plágio em δ, e b:B o início (b) e o fim (B) de um excerto de texto plagiado em d.

O *corpus PAN13-SourceRetrieval* foi criado para avaliar algoritmos de recuperação fonte na deteção de plágio externo [PHVS13], sendo utilizado como *dados de teste*[18] no âmbito da quinta competição internacional de deteção de plágio da PAN [PGH$^+$13]. Este *corpus* é composto por um conjunto de documentos suspeitos com 58 documentos de texto com plágio simulado por humanos. Este conjunto de documentos é uma amostra de dados de um *corpus* maior, designado aqui por *corpus Pan12131415-SRTC*, composto por mais outros três conjuntos de documentos com um total de 297 documentos de texto baseados no *corpus* de textos reutilizados Webis 2012 (Webis-TRC-2012)[19] [PHVS13, PGH$^+$12, PGH$^+$13, PHB$^+$14, HPS15]. Estes conjuntos foram disponibilizados em diferentes anos, nas competições anuais de deteção de plágio externo da PAN [PGH$^+$12, PGH$^+$13, PHB$^+$14, HPS15]. Salienta-se que os dois últimos conjuntos de documen-

[17] https://www.gutenberg.org

[18] https://www.uni-weimar.de/medien/webis/corpora/corpus-pan-labs-09-today/pan-13/pan13-data/pan13-source-retrieval-test-corpus2-2014-12-01.zip

[19] https://www.uni-weimar.de/en/media/chairs/webis/corpora/webis-trc-12

tos (*i.e.*, conjuntos de dados de teste) referentes a competição de 2014 e de 2015 não estão, atualmente[20], disponíveis para consulta pública.
Os documentos do *corpus Pan12131415-SRTC* foram criados por escritores contratados, para cada documento foi atribuído um assunto e um escritor [PGH+13, PHVS13]. A tarefa do escritor consistiu em utilizar o assunto como chave de pesquisa, no motor de pesquisa *ChatNoir*[21], para encontrar documentos fonte relacionados com o assunto em questão, para assim escrever o documento reutilizando excertos de texto dos documentos fonte encontrados [PGH+13, PHVS13]. Neste motor de pesquisa está indexado o subconjunto de páginas *web*, escritas na língua inglesa, do *corpus ClueWeb09*[22] [PHVS13, PGH+12, PGH+13, PHB+14, HPS15]. Os escritores foram instruídos para modificar o máximo possível os excertos de texto utilizados de forma à evitar a deteção de plágio [PGH+13, PHVS13]. Para tal, utilizaram duas estratégias principais para ofuscar a reutilização dos excertos de texto. A primeira consiste na paráfrase dos excertos de texto plagiados e a segunda consiste na intercalação de dois ou mais excertos de textos de diferentes documentos fonte [PGH+13, PHVS13]. Alguns escritores fizeram poucas modificações, enquanto outros fizeram muitas, assim a escala de casos de plágio pode variar de zero a muitas paráfrases e intercalações [PGH+13, PHVS13]. No método proposto representa-se o conjunto de 58 documentos de texto com plágio simulado por D_{sus}, e utiliza-se assim para avaliar os resultados com um conjunto de dados novo, ver Subsecção 3.4.4.
Com ambos, os *corpora*, *PAN-PC-11* e *PAN13-SourceRetrieval*, conseguiu-se desenvolver, implementar e avaliar o método de recuperação fonte proposto.

3.4.2 Implementação

Nesta subsecção aborda-se a implementação do método de recuperação fonte proposto. Primeiro, apresentam-se as ferramentas adotadas para auxiliar na implementação dos algoritmos dos diferentes "momentos" do primeiro ato da deteção de plágio externo, ver Figuras 2.1 e 3.1. A seguir, definem-se os valores impostos aos parâmetros de configuração utilizados na implementação pelas ferramentas adotadas. Por fim, apresenta-se a implementação desenvolvida e a ferramenta adotada para validar, com o conjunto de dados de validação, o método proposto.
A fim de suprir as necessidades funcionais de implementação dos algoritmos de *indexação da fonte* (*i.e.*, o Algoritmo 3.1), de *formulação da chave de pesquisa* (*i.e.*, o Algoritmo 3.2) e de *pesquisa e filtragem da fonte* (*i.e.*, o Algoritmo 3.3) do método de recuperação fonte proposto, adotaram-se os recursos disponibilizados por um conjunto de ferramentas preexistentes. As ferramentas adotadas foram selecionadas especificamente pelo facto de possuírem um atrativo leque de vantagens inerentes às suas utilizações, sendo, assim, consideradas úteis para auxiliar a implementação do método proposto. As principais vantagens associadas ao uso destas ferramentas são nomeadamente: primeiro, ter o código implementado na linguagem de programação *Java* [GJS+15]; segundo, ter o *código aberto*[23] [Ini07]; terceiro, ter a totalidade do código disponível na *Internet*; também se destaca a boa documentação, além de outras importantes características perante ferramentas da mesma estirpe (*e.g.*, robustez, escalabilidade, popularidade perante utilizadores, *etc*).
Inicialmente, adotou-se a biblioteca *HultigLib*[24] [Cor06] como a ferramenta base, para pro-

[20] Última consulta em 22 Março de 2016.
[21] http://webis15.medien.uni-weimar.de
[22] http://lemurproject.org/clueweb09
[23] Open-source.
[24] http://www.di.ubi.pt/ jpaulo/hultiglib/

cessar o conjunto de dados (i.e., os *corpora*). Esta biblioteca fora concebida com intuito de proporcionar eficiência e escalabilidade no processamento de grandes volumes de dados textuais [Cor06], fazendo uso de um vasto conjunto de recursos disponibilizado por ferramentas externas (*e.g.*, a biblioteca *Apache OpenNLP*[25] [Com14]), para além dos recursos implementados [Cor06]. Utilizou-se e aperfeiçoou-se, sempre que possível e necessário, os algoritmos implementados nos métodos de leitura de ficheiros de texto, nos métodos de análise sintática de fases e palavras, nos métodos de representação textual com estruturas de dados (*i.e.*, representar textos como objetos, frases como listas ligadas, palavras como listas ligadas e letras como vetores de caracteres), e nos métodos de manipulação das estruturas de dados textuais (*i.e.*, criação, eliminação, adição, remoção e modificação de dados). Com o auxílio desta biblioteca, primeiro, leu-se os dados textuais de cada documento como um conjunto ordenado de caracteres alfanuméricos. Segundo, normalizou-se o conjunto de caracteres através da conversão das letras maiúsculas para letras minúsculas. A seguir, segmentou-se o conjunto de caracteres em frases e palavras, através identificação dos caracteres de pontuação (*e.g.*, o ponto final"." e o espaço " "), recorrendo aos *dicionários normativos de construção frásica da língua inglesa*[26] da biblioteca *Apache OpenNLP*. Por último, a partir dos excertos frásicos detetados, criou-se uma representação do texto com estruturas de dados [BT08]. Consequentemente, um documento "genérico" d_i foi representado por uma *lista ligada* com *n nodos*, sendo o *n-ésimo nodo* correspondente a *n-ésima* frase do documento, tal que $d_i = [s_1, s_2, ..., s_n]$. Por sua vez, uma frase "genérica" s_i foi representada por uma *lista ligada* com *n nodos*, sendo o *n-ésimo nodo* correspondente a *n-ésima* palavra da frase, tal que $s_i = [w_1, w_2, ..., w_n]$.

Seguidamente, adotaram-se as funcionalidades nativas do motor de pesquisa *Apache Lucene*[27] para reduzir do espaço de pesquisa, através da implementação do algoritmo de *indexação da fonte* (*i.e.*, o Algoritmo 3.1). Utilizou-se o *Lucene* pelo facto de este ser robusto e flexível, assim como por ter as bibliotecas de indexação bem definidas e com o código aberto, possibilitando eventuais modificações e ajustes ao nível do código para as finalidades específicas de indexação [MHG10]. Recorreu-se às bibliotecas de indexação do *Lucene* para indexar o *corpus* fonte D_{font}. Para tal, utilizaram-se essas bibliotecas para reduzir os 11093 documentos fonte em uma única estrutura de dados (*i.e*, o *index_fonte*) compacta e representativa do D_{font}.

Na implementação do algoritmo de *formulação da chave de pesquisa* (Algoritmo 3.2) utilizaram-se, e aprimoraram-se para as necessidades em questão, os algoritmos de processamento de *corpora* da biblioteca *HultigLib* [Cor06]. Adotaram-se esses algoritmos para implementar parte da métrica *FIT* (Equação (3.2)) no que diz respeito ao cálculo da frequência de um termo *w* no *corpus* fonte D_{font}. Simultaneamente, implementou-se a extração do $V_{(\delta)}$ e o cálculo da frequência de cada termo *w* de $V_{(\delta)}$. Posteriormente, estabeleceu-se que, como parâmetro de configuração do Algoritmo 3.2, um termo *w* só pertence à chave de pesquisa se, e só se, o seu valor, dado pela métrica *FIT*, for maior que zero, permitindo assim selecionar apenas os termos relevantes e representativos de um documento.

Para implementar o algoritmo de *pesquisa e filtragem da fonte* (Algoritmo 3.3) adotaram-se as funcionalidades nativas do motor de pesquisa *Apache Lucene*. Recorreram-se as bibliotecas de pesquisa, deste motor, para encontrar o conjunto de documentos fonte candidatos $\Sigma_{(\delta)}$ relacionados com o documento suspeito δ. Utilizaram-se essas bibliotecas para, com a chave de pesquisa, pesquisar no *índice fonte* pelos documentos fonte candidatos $\Sigma_{(\delta)}$ que, muito provavelmente, foram alvo de plágio pelo δ. Para cada pesquisa, definiram-se, como parâmetros

[25] https://opennlp.apache.org/

[26] http://opennlp.sourceforge.net/models-1.5/

[27] https://lucene.apache.org/core/

de configuração, os valores de $a = 0.05$ e $\beta = 100$, assim, para a Equação (3.3), o valor de μ foi igual ao mínimo entre 5% dos melhores termos da chave de pesquisa e os 100 melhores termos da, mesma, chave, tal que $\mu = \min(a \times |chave\_pesquisa|, \beta)$. Consequentemente, um termo $w \in chave\_pesquisa$ só é utilizado na pesquisa se, e só se, o seu valor, dado pela métrica *FIT*, estiver entre os μ melhores valores contidos na chave de pesquisa. Para cada filtragem da fonte (*i.e.*, seleção dos resultados da pesquisa), definiu-se, como parâmetro de configuração, o valor de $\omega = 100$ documentos, tal que $|\Sigma_{(\delta)}| = 100$. Consequentemente, um documento fonte $d \in D_{font}$ só é considerado como candidato relevante a ter sido alvo de plágio se, e só se, a sua disposição estiver entre os ω melhores resultados ditados pelo Lucene.
Implementou-se uma segunda versão do algoritmo de *pesquisa e filtragem da fonte*, definido no Algoritmo 3.3, para validar o método proposto com o conjunto de dados de validação (*i.e.*, *com o corpus PAN13-SourceRetrieval*) descrito na Subsecção 3.4.1. Esta versão diverge da anterior por utilizar o motor de pesquisa *ChatNoir*, através da *Internet*. A utilização deste motor deve-se ao facto deste possuir o *índice fonte* (*i.e.*, o index_fonte) dos 500 milhões de documentos fonte, escritos na língua inglesa, do *corpus ClueWeb09* [PHVS13], a partir do qual o *corpus* suspeito D_{sus} (*i.e.*, *o corpus PAN13-SourceRetrieval*) fora criado, fazendo uso de plágio.
A confluência das implementações dos algoritmos, supracitados, e as ferramentas externas adotadas mais os valores impostos aos parâmetros de configuração, formam a implementação do método de recuperação fonte proposto. Com a implementação do método proposto torna-se possível identificar os documentos fonte com provável plágio por um documento suspeito. Apresenta-se a seguir, na Subsecção 3.4.3, os métodos de avaliação utilizados para avaliar o método proposto para o conjunto de dados de treino e teste.

3.4.3 Métodos de Avaliação

Nesta subsecção aborda-se o conjunto de métodos de avaliação adotado para avaliar o método de recuperação fonte proposto. Primeiro, apresenta-se o conjunto de métodos de avaliação, a sua divisão em dois conjuntos distintos de métricas de avaliação, assim como o racional desta divisão. A seguir, descreve-se a utilização destes conjuntos na análise dos resultados obtidos nas diferentes fases de criação do método proposto, e com diferentes conjuntos de dados. Por último, apresentam-se e descrevem-se todas as métricas de avaliação dos dois conjuntos adotados.
A fim de avaliar o primeiro ato da deteção de plágio externo, ver Diagrama 3.1, adotou-se um conjunto de métodos de avaliação [KBLP55, Rij79, PGH+13] para avaliar o método de recuperação fonte proposto, com um conjunto de dados de treino e teste, e outro de validação. Este conjunto de métodos de avaliação divide-se em dois conjuntos distintos de métricas de avaliação. O primeiro conjunto [KBLP55, Rij79] foi utilizado para avaliar os resultados obtidos nas fases de desenvolvimento, implementação e teste do método proposto, a partir da experimentação do método proposto com o conjunto de dados de treino e teste (*i.e.*, o *corpus PAN-PC-11*, descrito na Subsecção 3.4.1). O segundo conjunto de métricas de avaliação [PGH+13] foi utilizado para validar os resultados obtidos com uma amostra desconhecida de dados de validação (*i.e.*, o *corpus PAN13-SourceRetrieval*, descrito na Subsecção 3.4.1). A utilização de um segundo conjunto de métricas de avaliação permite corroborar que o método proposto não se encontra "superajustado"[28] [GCF+15] ao conjunto de dados de treino e teste, assim como permite corroborar que este não se encontra "subajustado"[29] [GCF+15] ao conjunto de dados de validação,

[28] Overfitting.
[29] Underfitting.

e por fim, também, permite viabilizar a comparação entre o método proposto e os demais métodos de recuperação fonte [PGH+13], pelo facto de ambos utilizarem o mesmo conjunto de métricas de avaliação e o mesmo conjunto de dados [PGH+13].

O primeiro conjunto de métricas de avaliação é constituído por quatro métricas: o *Precision*; o *Recall*; o *F-Measure*; e a *Accuracy*. Utilizaram-se essas métricas pelo facto de serem as *normas de facto*[30] na avaliação de *sistemas de pesquisa de informação*[31] [KBLP55, Rij79, AAA11] e em *sistemas de aprendizagem automática*[32] [WF05]. Apresentam-se nas Equações (3.4a), (3.4b), (3.4c) e (3.4d) as métricas de avaliação do primeiro conjunto, nomeadamente, o *Precision*, o *Recall*, o *F-Measure* e a *Accuracy*, respetivamente.

$$Precision = \frac{TP}{TP + FP} \quad (3.4a)$$

$$Recall = \frac{TP}{TP + FN} \quad (3.4b)$$

$$F\text{-}Measure = \frac{(1 + \beta^2) \times Precision \times Recall}{\beta \times Precision + Recall} \quad (3.4c)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.4d)$$

Onde TP, FP, TN e FN, representam, respetivamente, os verdadeiros positivos, os falsos positivos, os verdadeiros negativos e os falsos negativos; com $\beta = 1$ para a equidade de pesos entre o *Precision* e o *Recall*. Para visualizar-se melhor os resultados experimentais do método proposto, utilizou-se a *matriz confusão*. Assim, a Tabela 3.1, inspirada em [WF05], apresenta uma *matriz confusão "genérica"* para as predições da classe realizadas por um "sistema genérico" em função do valor real da classe.

Sendo: a soma dos TP com os FN igual aos reais elementos constituintes da classe 1, e a soma dos FP com os TN igual aos reais elementos constituintes da classe 0; a soma dos TP com os FP igual aos elementos preditos como da classe 1, com os TP como acertos e com os FP como erros, respetivamente, para essa classe 1; e, finalmente, a soma dos FN com os TN igual aos elementos preditos como da classe 0, tendo os TN como acertos e os FN como erros, respetivamente, para essa classe 0.

Tabela 3.1: *Matriz confusão* para as predições da classe em função do valor real da classe. Inspirada em [WF05].

		Predição da classe		
		1	0	Total:
Classe	1	TP	FN	-
	0	FP	TN	-
Total de predições:		-	-	

O segundo conjunto de métricas de avaliação, adotado para validar os resultados obtidos nas fases de teste e validação do método proposto com o conjunto de dados de validação, é constituído por três métricas de avaliação, nomeadamente, o *precision*, o *recall* e o *F1*. Estas métricas, originalmente definidas em [PGH+13], são "especializações" do ***P**recision*, do ***R**ecall* e do ***F-Measure***, criadas para medir a performance de algoritmos de recuperação fonte, tendo

[30] De facto standard – https://en.wikipedia.org/wiki/De_facto_standard

[31] Information retrieval systems.

[32] Machine learning systems.

em consideração as deteções de documentos fonte *extremamente semelhantes*[33], que apesar de não terem sido alvo de plágio pelo δ, são *extremamente semelhantes* com os documentos fonte que foram alvo de plágio por δ; sendo ambos pertencentes a um *corpus* de documentos fonte de elevadas dimensões [PGH+13, PHB+14, HPS15].

A utilização do segundo conjunto de métricas de avaliação viabilizou a comparação entre os resultados obtidos com o método proposto e os resultados de outros métodos de recuperação fonte [PGH+13], uma vez que o conjunto de dados é igual [PGH+13]. Nas Equações (3.5a), (3.5b) e (3.5c) são apresentadas as métricas de avaliação do segundo conjunto:

$$precision = \frac{|D_{ret} \cap D_{dup}|}{|D_{ret}|} \quad (3.5a)$$

$$recall = \frac{|D'_{ret} \cap D_{src}|}{|D'_{ret}|} \quad (3.5b)$$

$$F_1 = \frac{2 \times precision \times recall}{precision + recall} \quad (3.5c)$$

onde D_{ret} representa o conjunto de documentos candidatos recuperados, pelo algoritmo de recuperação fonte, para um determinado documento suspeito δ [PGH+13], no contexto do método proposto representou-se D_{ret} como $\Sigma_{(\delta)}$, isto é $D_{ret} = \Sigma_{(\delta)}$; o D_{dup} representa o conjunto de documentos candidatos que não foi alvo de plágio, em que cada documento deste conjunto é *extremamente semelhante*[34] a um ou mais documentos do conjunto de documentos fonte que foi alvo de plágio, isto é $D_{dup} \approx^{34} D_{src}$, $D_{dup} \notin D_{src}$, $D_{dup} \subseteq D_{ret}$ e $D_{dup} \cap D'_{ret} = \emptyset$, assim no contexto do método de recuperação fonte proposto $D_{dup} \approx refs_{(\delta)}$, $D_{dup} \notin refs_{(\delta)}$, $D_{dup} \subseteq \Sigma_{(\delta)}$ e $\left(D_{dup} \cap (refs_{(\delta)} \cap \Sigma_{(\delta)})\right) = \emptyset$; o D'_{ret} representa o conjunto de documentos candidatos que foi corretamente identificado, pelo algoritmo de recuperação fonte, como o conjunto de documentos fonte que foi alvo de plágio [PGH+13], isto é $D'_{ret} = \Sigma_{(\delta)} \cap refs_{(\delta)}$; o D_{src} representa o conjunto de documentos fonte que foi alvo de plágio por um δ [PGH+13], no contexto do método proposto representou-se D_{src} como $\forall d \in ref(\delta)$, tal que $D_{src} = ref(\delta) \cap D_{font}$; e o $F1$ representa a média harmónica entre o ***precision*** e ***recall***. Além dessas três métricas de avaliação, foram utilizados mais seis parâmetros de "custo-eficiência" (*i.e.*, "cost-effectiveness") [PGH+13]. Esses parâmetros são: o valor médio de pesquisas submetidas ao motor de pesquisa; o valor médio de documentos candidatos recuperados; o valor médio de pesquisas submetidas, necessárias para encontrar o primeiro documento fonte que foi alvo de plágio; o valor médio de documentos candidatos recuperados, necessários para se obter o primeiro documento fonte que foi alvo de plágio; o número total de documentos fonte recuperados que foram alvo de plágio; e o tempo total de execução do algoritmo de recuperação fonte.

Com a utilização dos métodos de avaliação, supracitados, e suas respetivas métricas de avaliação foi possível avaliar o método de recuperação fonte proposto segundo as normas padrões de avaliação [KBLP55, Rij79, WF05], e segundo as normas de avaliação de uma competição in-

[33] Near-duplicate.

[34] Em [PGH+13], dois documentos fonte (*e.g.*, d_A e d_B) são considerados como *extremamente semelhantes* (*i.e.*, near-duplicate) se, e apenas se, pelo menos uma das seguintes condições se verificar: *igualdade*, o d_A é extremamente semelhaste ao d_B, se $d_A = d_B$; *similaridade*, o d_A é extremamente semelhaste ao d_B, se o *coeficiente de Jaccard* das suas *palavras-3-gramas* > 0.8, *palavras-5-gramas* > 0.5 e *palavras-8-gramas* > 0; *contenção*, o d_A é extremamente semelhaste ao d_B, se os excertos de texto de δ, com plágio confirmado de d_B, estão contidos em d_A, tal que o *coeficiente de Jaccard* das suas *palavras-3-gramas* > 0.8, *palavras-5-gramas* > 0.5 e *palavras-8-gramas* > 0. [PGH+13]

ternacional de deteção de plágio [PGH+13]. A seguir, ver Subsecção 3.4.4, são apresentados os resultados experimentais obtidos.

3.4.4 Resultados Experimentais

Nesta subsecção aborda-se os resultados experimentais obtidos com o método de recuperação fonte proposto. Primeiro, descrevem-se as experiências realizadas com o método proposto e o conjunto de dados de treino e teste. A seguir, apresentam-se e discutem-se os valores obtidos. Segundo, descrevem-se as experiências realizadas com o método proposto e o conjunto de dados de validação. A seguir, apresentam-se e discutem-se os valores obtidos e comparam-se com os resultados de outros métodos de recuperação fonte.

A fim de avaliar o desempenho do método de recuperação fonte proposto com o conjunto de dados de treino e teste (*i.e.*, o *corpus PAN-PC-11*), realizaram-se duas experiências com o conjunto de documentos suspeitos $D_{sus} \subset$ *PAN-PC-11*: a primeira para o conjunto de documentos suspeitos com plágio confirmado $D_{black} \subset D_{sus}$; e a segunda para o conjunto de documentos suspeitos sem plágio $D_{white} \subset D_{sus}$. Previamente, indexou-se a fonte através do uso das bibliotecas de indexação do *Lucene* para criar o índice fonte (*i.e.*, *index_fonte*) do conjunto de documentos fonte $D_{fonte} \subset$ *PAN-PC-11*, à elegibilidade deste índice em ser alvo de pesquisas chamou-se de *motor de plágio (MP)*.

Na primeira experiência formularam-se as chaves de pesquisa para o conjunto de documentos suspeitos $D_{black} = \{\delta_1, \delta_2, ..., \delta_{4992}\}$ com plágio confirmado, uma chave por documento $chave\_pesquisa_i \in \delta_i$. Seguidamente, pesquisaram-se e filtraram-se a fonte, através da submissão das chaves de pesquisas às bibliotecas de pesquisa do *Lucene* e, posterior, recolha dos respetivos conjuntos de documentos candidatos devolvidos pelo *MP* (*i.e.*, *MP:* $chave\_pesquisa_i \mapsto \Sigma_{(\delta_i)}$)[35]; para todas as pesquisas e filtragens seguiu-se uma "política" de submissão e recolha dirigida por $a = 0.05$, $\beta = 100$ e $\omega = 100$ [FC15]. Finalmente, contabilizaram-se os sucessos e os insucessos do método proposto em formular as chaves de pesquisa que, ao serem submetidas ao *MP*, permitiram a recuperação de pelo menos uma fonte de plágio, comum, entre cada um dos respetivos conjuntos de documentos de candidatos e dos respetivos conjuntos de documentos fonte plagiados; ou seja, para cada um dos documentos suspeitos com plágio ($\forall \delta \in D_{black}$): contabilizou-se como um sucesso se, e apenas se, a $chave\_pesquisa_i \in \delta_i$, submetida ao *MP*, permitiu recuperar pelo menos um documento candidato igual a um documento fonte que fora alvo de plágio (*i.e.*, $d \in refs_{(\delta_i)}$) por um δ_i (*i.e.*, $|\Sigma_{(\delta_i)} \cap refs_{(\delta_i)}| \geq 1$); e para o caso contrário (*i.e.*, $\Sigma_{(\delta_i)} \cap refs_{(\delta_i)} = \emptyset$) contabilizou-se como um insucesso.

Na segunda experiência com o *corpus PAN-PC-11* fez-se o teste de despistagem de erros para o método proposto. O objetivo deste teste consistiu em determinar o fator de aleatoriedade dos "resultados" devolvidos pelo *MP*, ou seja, determinar se as fontes de plágio que foram corretamente identificadas (*i.e.*, $\Sigma_{(\delta)} \cap refs_{(\delta)}$) derivaram do método proposto ou de um fator aleatório desconhecido (*e.g.*, proveniente do *Lucene*). Para tal, utilizou-se o conjunto de documentos suspeitos $D_{white} = \{\delta_1, \delta_2, ..., \delta_{5546}\}$ sem plágio para formular as chaves de pesquisa, uma chave de pesquisa por documento suspeito (*i.e.*, $chave\_pesquisa_j \in \delta_j : \forall \delta \in D_{white}$). A seguir, pesquisaram-se, através da submissão das chaves de pesquisa às bibliotecas de pesquisa do *Lucene*, e filtraram-se a fonte, através da recolha dos respetivos conjuntos de documentos candidatos devolvidos pelo *MP* (*i.e.*, *MP:* $chave\_pesquisa_j \mapsto \Sigma_{(\delta_j)}$); com as pesquisas e repetitivas filtragens definidas para $a = 0.05$, $\beta = 100$ e $\omega = 100$ [FC15]. Mais adiante, para cada documentos suspeito δ_j do conjunto de documentos suspeitos sem plágio D_{white} (*i.e.*, $\forall \delta \in D_{white}$)

[35] *E.g.*, *g:* $x \mapsto y$ significa que *g* mapeia o elemento *x* para o elemento *y*.

foi aleatoriamente[36] atribuído um número inteiro $k : 1 \leq k \leq |D_{black}|$ correspondente ao *i-ésimo* elemento (*i.e.*, ao δ_i) do conjunto de documentos suspeitos com plágio confirmado D_{black}. Assim, a partir destas atribuições e respetivas correspondências, simularam-se, para cada $\delta_j \in D_{white}$, "pseudo-referências" de "pseudo-plágio" $\widetilde{refs}_{(\delta_j)} = \{\tilde{r}_1, \tilde{r}_2, ..., \tilde{r}_n\}$ entre o documento suspeito δ_j e as fontes de plágio (*i.e.*, $\forall d \in refs_{(\delta_i)}$) do documento suspeito $\delta_i \in D_{black} : i = k$, de tal forma que $\widetilde{refs}_{(\delta_j)} = \{\langle \delta_j^{a_1:A_1}, d_l^{b_1:B_1} \rangle; \langle \delta_j^{a_2:A_2}, d_l^{b_2:B_2} \rangle; ...; \langle \delta_j^{a_n:A_n}, d_l^{b_n:B_n} \rangle\} : \forall d \in refs_{(\delta_{i=k})}, 1 \leq l \leq |refs_{(\delta_{i=k})}|$ e $\{a_1{:}A_1; a_2{:}A_2; ...; a_n{:}A_n\} \in refs_{(\delta_{i=k})}$. Por fim, contabilizaram-se os sucessos e os insucessos do método proposto em formular chaves de pesquisa "*únicas*" que, ao serem submetidas ao *MP*, *não* permitiram a recuperação *nenhuma* fonte de plágio, comum, entre cada um dos respetivos conjuntos de documentos candidatos e os respetivos conjuntos de documentos fonte de "pseudo-plagiado"; ou seja, para cada um dos documentos suspeitos *sem* plágio (*i.e.*, $\forall \delta \in D_{white}$): contabilizou-se como um sucesso se, e apenas se, a $chave\_pesquisa_j \in \delta_j$, submetida ao *MP*, *não* permitiu recuperar *nenhum* documento candidato (*i.e.*, $\nexists d \in \Sigma_{(\delta_j)}$) igual a um documento fonte de "*pseudo-plágio*" (*i.e.*, $d \in \widetilde{refs}_{(\delta_j)}$), tal que $\Sigma_{(\delta_j)} \cap \widetilde{refs}_{(\delta_j)} = \emptyset$; e para o caso contrário, contabilizou-se como um insucesso quando houve pelo menos um documento candidato (*i.e.*, $\exists d \in \Sigma_{(\delta_j)}$) igual a um documento fonte de "*pseudo-plágio*" (*i.e.*, $d \in \widetilde{refs}_{(\delta_j)}$), tal que $|\Sigma_{(\delta_j)} \cap \widetilde{refs}_{(\delta_j)}| \geq 1$.

Tabela 3.2: *Matriz confusão* da recuperação fonte com *chaves de pesquisa* em função dos *documentos suspeitos*.

		Recuperação fonte com chaves de pesquisa		
		$\forall \delta \in D_{black} : \Sigma_{(\delta_i)} \cap ref(\delta_i) \neq \emptyset$	$\forall \delta \in D_{white} : \Sigma_{(\delta_j)} \cap \widetilde{refs}(\delta_j) = \emptyset$	Total de documentos:
Documentos suspeitos	$\forall \delta \in D_{black} : ref(\delta_i) \neq \emptyset$	4304	688	4992
	$\forall \delta \in D_{white} : ref(\delta_j) = \emptyset$	160	5386	5546
	Total de recuperações:	4464	6074	

Na Tabela 3.2 apresentam-se os valores da *matriz confusão*, obtidos com as duas experiências, descritas anteriormente, do método de recuperação fonte proposto com o conjunto de dados de treino e teste (*i.e.*, o *corpus PAN-PC-11*). Na *matriz confusão*, os sucessos e os insucessos da primeira experiência são apresentados como os verdadeiros positivos (TP) e os falsos negativos (FN), respetivamente; e, semelhantemente, são apresentados os sucessos e os insucessos da segunda experiência como os verdadeiros negativos (TN) e os falsos positivos (FP), respetivamente. Assim, evidencia-se que, para os 4992 documentos suspeitos com plágio confirmado e as suas respetivas 4992 chaves de pesquisa que foram formuladas e submetidas ao *MP*, houve 4304 recuperações de um ou mais documentos fonte de plágio; contrariamente, houve 688 chaves de pesquisa que não conseguiram recuperar fontes de plágio. Salienta-se também que, para os 5546 documentos suspeitos sem plágio e as suas respetivas 5546 chaves de pesquisas que foram formuladas e submetidas ao *MP*, houve 160 recuperações de um ou mais documentos fonte de "pseudo-plágio"; de modo contrário, houve 5386 chaves de pesquisa que representaram corretamente os seus respetivos documentos e as ausências de plágio destes, dado que não houve deteções de documentos fonte de "pseudo-plágio".

Na Tabela 3.3 apresentam-se os resultados de avaliação do primeiro conjunto de métricas de avaliação, ver Subsecção 4.4.3, obtidos das experiências realizadas com o método proposto e o *corpus PAN-PC-11* (*i.e.*, o conjunto de dados de treino e teste, ver Subsecção 3.4.1). Na

[36] ***int*** *random* = ***new*** *Random().nextInt(*$|D_{black}|$*);*

Tabela 3.3: Resultados de avaliação da recuperação fonte com *chaves de pesquisa* para os *documentos suspeitos*.

Precision	Recall	F-Measure	Accuracy	Documentos suspeitos
0.964	0.862	0.910	0.919	D_{black}
0.887	0.971	0.927		D_{white}

primeira coluna, desta tabela, apresentam-se os dois resultados de *Precision*, ver Equação (3.4a) um com D_{black} e o outro com D_{white}. O valor de *Precision* do método proposto na primeira experiência com D_{black} mostrou-se superior (0.964) ao valor da segunda experiência (0.887) com D_{white}. Este facto deve-se à precisão das chaves de pesquisa em representarem os dados dos documentos suspeitos e consequentemente os dados com plágio desses, possibilitando um maior número de, corretas, recuperações de uma ou mais fontes de plágio para os documentos suspeitos analisados (*i.e.*, $\forall \delta \in D_{black} : \Sigma_{(\delta_i)} \cap ref(\delta_i) \neq \emptyset$), em função do fator de aleatoriedade dos "resultados" devolvidos pelo *MP* (*i.e.*, $\forall \delta \in D_{white} : \Sigma_{(\delta_j)} \cap \widetilde{refs}(\delta_j) = \emptyset$). Na segunda coluna, da mesma tabela, são apresentados os resultados de *Recall*, para as experiências do método proposto com D_{black} e D_{white}. Nesta métrica, definida na Equação (3.4b), avalia-se a informação relevante extraída em função da informação relevante existente [Cor03, WF05], e esta avaliou com uma abrangência de (0.862), as chaves de pesquisa que permitiram corretas deteções da(s) fonte(s) de plágio em função de todos os documentos suspeitos com plágio D_{black} (*i.e.*, $\forall \delta \in D_{black} : ref(\delta_i) \neq \emptyset$); e com uma abrangência de (0.971), as chaves de pesquisa que representaram corretamente as ausências de plágio em D_{white} em função da(s) "pseudo-referência(s)" de "pseudo-plágio" simuladas (*i.e.*, $\forall \delta \in D_{white} : ref(\delta_j) = \emptyset$). Na quarta coluna da Tabela 3.3, apresenta-se o resultado da "*taxa de acerto*"[37] das experiências do método proposto com o *corpus PAN-PC-11*. Este resultado, superior a (0.9), demonstra a avaliação, "final e global"[38], do desempenho do método de recuperação fonte proposto com o conjunto de dados de treino e teste.

Tabela 3.4: Resultados de avaliação da recuperação fonte para pan2013, inspirada na **Tabela 1** de [PGH+13].

Métodos	Total de Documentos Candidatos			Total de Carga Computacional		Carga Computacional da 1ª Recuperação		Total de Recuperações	Tempo Total
	F_1	*precision*	*recall*	Pesquisas	Documentos Candidatos	Pesquisas	Documentos Candidatos		
Método Proposto	0.21	0.14	0.39	**5.55**	57.07	7.00	43.00	**671**	1396.2 m
[Eli13]	0.17	0.12	0.44	44.50	107.22	16.85	15.28	5	241.7 m
[Gil13]	0.04	0.02	0.10	16.10	33.02	18.80	21.70	38	**15.1 m**
[HEB13]	0.44	**0.63**	0.38	32.04	**5.93**	8.92	**1.47**	9	152.7 m
[HEB13]	0.01	0.01	**0.65**	48.50	5691.47	2.46	285.66	3	4098.0 m
[KQD+13]	0.35	0.50	0.33	44.04	11.16	7.74	1.72	15	310.5 m
[LCPJ13]	0.06	0.04	0.23	12.38	261.95	**2.44**	74.79	10	1637.9 m
[SKB13]	0.15	0.11	0.35	161.21	81.03	184.00	5.07	16	655.3 m
[VFR13]	**0.47**	0.55	0.50	116.40	14.05	17.59	2.45	5	1163.0 m

[37] Accuracy: métrica de avaliação definida na Equação (3.4d).

[38] *I.e.*, por usar todos os valores da *matriz confusão*, ver Tabela 3.2 e Equação (3.4d).

A fim de validar o desempenho do método de recuperação fonte proposto com o conjunto de dados de validação (*i.e.*, o *corpus PAN13-SourceRetrieval*), fez-se uma experiência com o conjunto de documentos suspeitos $D_{sus} \subseteq$ *PAN13-SourceRetrieval*. Nessa experiência, inicialmente, formularam-se as chaves de pesquisa para o conjunto de documentos suspeitos $D_{sus} = \{\delta_1, \delta_2, ..., \delta_{58}\}$ com plágio confirmado $D_{sus} = D_{black}$ e $refs_{(\delta_i)} \neq \emptyset : \delta_i \in D_{sus}$, uma chave por documento $chave\_pesquisa_i \in \delta_i$. Uma vez o *corpus PAN13-SourceRetrieval* não possui um conjunto de documentos fonte (*i.e.*, necessário para estimar $P(w)$ da função *FIT* (Equação (3.2)), então estimou-se $P(w)$ no *corpus PAN-PC-11*. Seguidamente, pesquisaram-se e filtraram-se a fonte, através da submissão das chaves de pesquisas ao motor de pesquisa *ChatNoir*[39] e, posterior, recolha dos respetivos conjuntos de documentos candidatos devolvidos pelo *ChatNoir* ($chave\_pesquisa_i \mapsto \Sigma_{(\delta_i)}$); para todas as pesquisas e filtragens seguiu-se uma "política" de submissão e recolha ditada por $a = 0.05$, $\beta = 100$ e $\omega = 100$ [FC15]. Uma vez que as pesquisas no *ChatNoir* estão limitadas a um máximo de 10 palavras por pesquisa, e dado que o número de palavras nas chaves de pesquisa, formuladas pelo método proposto, varia e nem sempre é menor ou igual a 10 palavras, então para cada δ_i dividiu-se a sua $chave\_pesquisa_i = \{w_1, w_2, ..., w_\mu\}$ em $\lceil \frac{\mu}{10} \rceil$[40] *sub-chaves de pesquisa distintas* $chave\_pesquisa_i = sub\_chave_{i:1} \cup sub\_chave_{i:2} \cup ... \cup sub\_chave_{i:n} : 1 \leq n \leq \lceil \frac{\mu}{10} \rceil$ e $1 \leq |sub\_chave_{i:j}| \leq 10$. Consequentemente, filtraram-se os $\frac{\omega}{10}$ documentos candidatos mais relevantes, segundo a disposição ditada pelo *ChatNoir*, para cada $sub\_chave_{i:j}$ submetida. Finalmente, para cada $\delta_i \in$ *PAN13-SourceRetrieval* e respetivas $refs_{(\delta_i)}$, contabilizaram-se: o número de documentos candidatos *extremamente semelhastes*[34] aos documentos fonte de plágio $|D_{dup} \approx refs_{(\delta)}| : D_{dup} \subseteq \Sigma_{(\delta)}$; e o número de documentos fonte de plágio recuperados com o método proposto $|\Sigma_{(\delta)} \cap refs_{(\delta)}|$.

A Tabela 3.4, inspirada na **Tabela 1** de [PGH+13], apresenta os resultados de avaliação do segundo conjunto de métricas de avaliação, ver Subsecção 4.4.3, obtidos com a experiência, descrita anteriormente, do *método proposto* e o *corpus PAN13-SourceRetrieval*. Assim como, para efeitos de comparação, são apresentados os resultados de avaliação de *outros métodos de recuperação fonte* [Eli13, Gil13, HEB13, KQD+13, LCPJ13, SKB13, VFR13], originalmente apresentados e avaliados em [PGH+13]. Nesta tabela, na terceira linha, apresentam-se os resultados de validação do *método proposto* com o *corpus PAN13-SourceRetrieval*. Em termos gerais, na segunda coluna são apresentados os resultados de $F1$, o resultado do método proposto destacou-se como um dos melhores, estando na quarta posição entre os melhores de [PGH+13]. O resultado de *recall* do método proposto mostrou ser o quarto melhor perante os demais métodos avaliados [PGH+13]. Nas restantes colunas e respetivos resultados de avaliação, os resultados do método proposto, em termos gerais, mostraram-se pouco significativos perante os demais métodos avaliados em [PGH+13]. Salienta-se que os resultados de validação do método proposto com o *corpus PAN13-SourceRetrieval* distanciam-se dos resultados de avaliação com *o corpus PAN-PC-11*. Primeiro, é possível que o valor de amostragem de documentos suspeitos do *corpus PAN13-SourceRetrieval* não tenha sido significante, o bastante, para o método proposto se destacar. Dado que o número de documentos suspeitos no *PAN13-SourceRetrieval* é, "quase astronomicamente", menor que o número de documentos suspeitos em *PAN-PC-11* (*i.e.*, $|PAN13\text{-}SourceRetrieval| = 58 > |PAN\text{-}PC\text{-}11| = 10538$). Segundo, é possível que o desempenho do motor de pesquisa utilizado para as experiências de avaliação (*i.e.*, o *Lucene*) tenha suplantado o desempenho do motor de pesquisa utilizado na experiência de validação (*i.e.*, o *ChatNoir*), dado que o método proposto é notoriamente dependente das pesquisas e do índice fonte do motor de pesquisa adotado. Por último, é possível que a estimação do denominador

[39] http://webis15.medien.uni-weimar.de

[40] *E.g.*, $\lceil 3 \rceil = 3$, $\lceil 3.1 \rceil = 4$, $\lceil 3.5 \rceil = 4$ e $\lceil 3.9 \rceil = 4$.

$P(w)$ da função *FIT*, no $D_{font} \in$ *PAN-PC-11*, para formulação das chaves de pesquisa de $D_{sus} \in$ *PAN13-SourceRetrieval*, não tenha sito a decisão mais acertada, porém a alternativa, estimar $P(w)$ no *corpus ClueWeb09*, seria ainda menos acertada (*i.e.*, $|ClueWeb09| \geq 5 \times 10^8$ [PHVS13]). Deste modo, com as experiências apresentadas conclui-se a avaliação do método de recuperação fonte proposto. Na avaliação com o *corpus PAN-PC-11* os resultados mostraram-se promissores, com valores de *F-Measure* superiores a 0.9. Porém, na validação do método proposto, com o *corpus PAN13-SourceRetrieval*, os resultados mostraram-se desfavoráveis e suscetíveis a pequenos reajustes e melhorias.

3.5 Sumário

Neste capítulo foi abordado o método de recuperação fonte proposto para a redução do espaço de pesquisa no universo de documentos fonte. Inicialmente foram introduzidos os principais conceitos associados ao assunto em questão, assim como foi definida a problemática encontrada. Seguidamente, foi definida uma estratégia, divida em "momentos, para a resolução do problema em causa, assim como foi apresentada a notação que seria utilizada. Posteriormente, foram apresentados os "momentos" responsáveis para resolução do problema, sendo que esses constituem o método de recuperação fonte proposto.
Finalmente, foram apresentadas as experiências realizadas e os resultados obtidos com o método proposto: foi apresentado o conjunto de dados utilizado; assim como, na implementação foram descritas as ferramentas adotadas e os parâmetros utilizados; a seguir, foram apresentados os métodos de avaliação adotados para avaliar e validar o método proposto; e, por fim, formam descritos os resultados experimentais obtidos do método de recuperação fonte proposto.
A seguir, no Capítulo 4 apresenta-se o segundo ato da deteção de plágio externo, intitulado por método de análise proposto.

Capítulo 4

Método de Análise Detalhada Proposto

Neste capítulo aborda-se o método de análise detalhada proposto para a deteção de plágio no documento suspeito δ. Primeiro, introduzem-se os principais conceitos associados a análise detalhada, a seguir define-se problemática abordada e apresenta-se a solução para o problema em causa. Segundo, apresenta-se o método proposto para solucionar o problema, e a sua respetiva divisão em "momentos": métrica de deteção de plágio; e pesquisa e análise de plágio. A seguir, apresentam-se as métricas heurísticas criadas para a deteção de plágio. Mais adiante, apresentam-se as métricas inteligentes criadas por intermédio da extração de características de plágio para a posterior indução de padrões de classificação de plágio. Logo a seguir, apresenta-se o "momento" de pesquisa e análise de plágio nos dados suspeitos. Por último, apresentam-se as experiências realizadas e os resultados obtidos com o método proposto, dando ênfase ao conjunto de dados utilizado, as ferramentas adotadas e os parâmetros de configuração utilizados na implementação do método, assim como, apresentam-se aos métodos de avaliação adotados e os resultados experimentais obtidos do método de análise detalhada proposto.

A análise detalhada, abordada *neste capítulo*[1], constitui o segundo ato da deteção de plágio externo. Seguidamente, ao método de recuperação fonte proposto, descrito no Capítulo 3, e a respeitava identificação do conjunto de "documentos candidatos"[2] a terem sido alvo de plágio $\Sigma_{(\delta)} = \{d_1, d_2, ..., d_\omega\}$, inicia-se a análise detalhada entre os pares (documento suspeito *versus* documentos candidatos); conforme apresentado no diagrama da Figura 2.1. Nessa análise pretende-se detetar se existem indícios de plágio, identificando assim o(s) *excerto(s) com plágio* no documento suspeito e o(s) respetivo(s) *excerto(s) plagiado(s)* no documento candidato; para todos os pares (*i.e.*, δ *versus* $\forall d \in \Sigma_{(\delta)}$).

Porém, a deteção de indícios de plágio requer uma métrica robusta e capaz de reconhecer corretamente as *paridades textuais*[3] e as *divergências textuais*[4] entre os excertos de texto de um documento suspeito δ e os excertos de texto de um documento candidato d. Assim, na deteção de plágio textual existe a problemática do correto reconhecimento das *homogeneidades textuais* (*i.e.*, paridades textuais) e das *heterogeneidades textuais* (*i.e.*, divergências textuais) entre as cópias ilícitas de dados e os dados originais. O prelúdio dessa problemática assenta no infrator. Este ao perpetrar o plágio pode tê-lo feito com uma *cópia literal, sucinta ou prolixa* de texto, constituída por palavras, frases, parágrafos e/ou, até mesmo, por capítulos inteiros do texto original. E para não ser detido em flagrante delito, este, pode ter recorrido a *ofuscação textual* [PSBR10] para camuflar e ocultar o plágio perpetrado. Nessa ofuscação um texto original pode ser transformado em um texto equivalente, na forma suscita (*i.e.*, um resumo [Pri16]) ou na forma prolixa (*i.e.*, uma paráfrase [Pri16]). Nessa transformação o conjunto original de palavras pode ser reduzido para um pequeno conjunto de palavras relevantes (*i.e.*, uma cópia sucinta) ou pode ser expandido para um conjunto maior de palavras difusas (*i.e.*, uma cópia prolixa [Pri16]). Na redução e na expansão essas palavras podem ser reordenadas e/ou substituídas

[1] Credita-se a estrutura "expositivo-argumentativa" do Capítulo 4 ao Capítulo 3.

[2] Foram considerados como candidatos por possuírem várias semelhanças léxico-gramaticais com um documento suspeito.

[3] *I.e.*, tudo o que é igual ou idêntico entre dois textos. [Pri16]

[4] *I.e.*, tudo o que é diferente, e que divergiu do que era igual ou similar, entre dois textos.

por outras similares (*e.g.*, sinónimos[5], hipónimos[6], hiperónimos[7], *etc*). Logo, o infrator ao perpetrar o plágio faz uma *cópia literal, sucinta ou prolixa* de texto, na qual a *similaridade textual*[8] com o texto original ultrapassa os limites da similaridade lexical, baseada meramente em coocorrências de palavras.

Portanto, deparou-se com os seguintes desafios: *similaridade documental* [Hua08, SGM00, LG05], podendo ter pares de documentos similares com ou sem plágio; *mineração de dados* [WF05, GCF+15], visto que, a partir de uma conjunto de dados com plágio, pode-se extrair um conjunto de características textuais de plágio potencialmente relevantes para identificar regularidades típicas do plágio e da ofuscação textual.

Consequentemente, conjeturou-se que: se as cópias ilícitas de texto possuem indícios de plágio que ultrapassam os limites da similaridade textual, foi utilizada a ofuscação textual; se as cópias ilícitas de texto possuem características léxico-gramaticais implícitas, do comportamento humano do infrator na perpetração do plágio e na tentativa de ocultação do delito, que possam ser extraídas, algumas dessas características regularidades da ofuscação que possam ser identificadas;

então uma nova métrica para detetar os indícios de plágio nas paridades textuais e os indícios de ofuscação textual nas divergências textuais, através da combinação heurística das principais técnicas de deteção de similaridade textual; uma nova métrica para apreender os indícios de plágio e judiciar sob a sua existência, através da extração de características textuais de plágio que possam identificar regularidades típicas do plágio e da ofuscação textual, com a mineração de dados, e através da indução de padrões de classificação específicos de plágio para determinar a existência de plágio, com o recurso da inteligência computacional, mais especificamente, com o da programação genética.

A Figura 4.1 apresenta o diagrama do método de análise detalhada proposto para a resolução da problemática, descrita anteriormente, através da *pesquisa e análise de plágio* entre os pares de documentos (*i.e.*, δ *versus* $\forall d \in \Sigma_{(\delta)}$), com uma *métrica de deteção de plágio* robusta. Inicialmente, exploraram-se heuristicamente novas métricas de deteção de plágio, a partir das combinações heurísticas das principais técnicas de deteção de similaridade textual (Secção 4.1). Seguidamente, exploraram-se novas métricas inteligentes de deteção de plágio (Secção 4.2): através da extração de características textuais de plágio, a partir de conjunto de dados com plágio (Subsecção 4.2.1); para depois induzir padrões de classificação de plágio, com recurso à programação genética, e a consequente construção automática de uma métrica eficaz de deteção de excertos de plágio. Por fim, apresentam-se as experiências realizadas e os resultados obtidos com o método proposto (Secção 4.4).

Na abordagem utilizada representou-se um *corpus* como um conjunto de documentos D, um documento como uma sequência de frases $[s_1, s_2, ..., s_m]$, uma frase como uma sequência de palavras $[w_1, w_2, ..., w_n]$ e uma palavra como uma sequência de caracteres, sendo que $|radical_{(w_i)}| \leq |w_i|$.

Consequentemente, no método de análise detalhada proposto, considerou-se um ***corpus* de plágio** composto por dois *corpora*: D_{font} e D_{sus}, de tal forma que $corpus = D_{font} \cup D_{sus}$ e $D_{font} \cap D_{sus} = \emptyset$. Considerou-se o D_{sus} como sendo composto por dois *"sub-corpora"* menores: D_{white} e D_{black}, de tal forma que $D_{sus} = D_{white} \cup D_{black}$ e $D_{white} \cap D_{black} = \emptyset$; considerou-se, também, que, para um δ_k, existe uma ***função de valoração*** $f(.)$ que determina se $f(\delta_k) = white$ ou

[5] *I.e.*, palavras com o mesmo significado.
[6] *E.g.*, truta é hipónimo de peixe.
[7] *E.g.*, salmonídeo é um hiperónimo de salmão.
[8] *I.e.*, tudo o que é similar ou da mesma natureza entre dois textos. [Pri16]

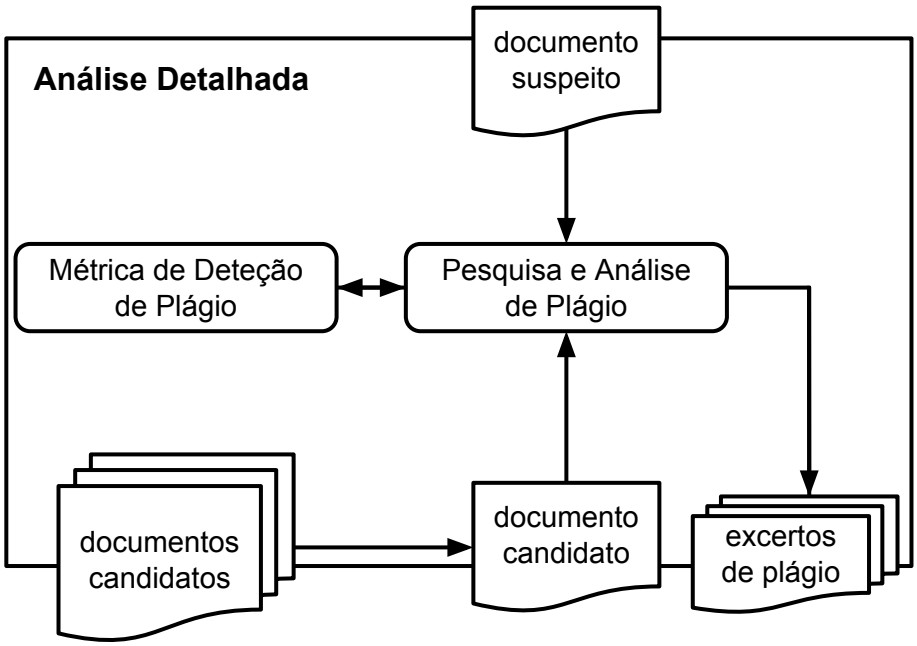

Figura 4.1: Diagrama do processo de análise detalhada.

$f(\delta_k) = black$. Assim, um ***corpus* suspeito sem plágio** D_{white} foi representado como um conjunto ordenado de elementos, com documentos suspeitos *sem plágio* e um ***corpus* suspeito com plágio** D_{black} foi representado como um conjunto ordenado de elementos, com documentos suspeitos *com plágio*. Por sua vez, **o plágio** num documento suspeito δ_i, tal que $f(\delta_{k=i}) = black$, foi representado como um conjunto ordenado de elementos, com *p* ***referências de plágio***[9] $(i.e.$, o(s) excerto(s) de texto com plágio, no documento com plágio, e o(s) respetivo(s) excerto(s) de texto plagiado(s), no(s) documento(s) original(ais)$)$, tal que $refs_{(\delta_i)} = \{r_1, r_2, ..., r_p\}$.
Finalmente, **uma referência de plágio** r_h foi representada como **um alinhamento entre dois excertos de texto**, tal que $r_h = \langle \delta_i^{a_h:A_h}, d_l^{b_h:B_h} \rangle$: onde $\delta_i^{a_h:A_h}$ representa o *h-ésimo* **excerto de texto com plágio** no δ_i, sendo $a_h{:}A_h$ o início (a_h) e o fim (A_h) do *h-ésimo* excerto de texto com plágio no *i-ésimo* documento suspeito δ_i com plágio $f(\delta_{k=i}) = black$; e onde $d_l^{b_h:B_h}$ representa o *h-ésimo* **excerto de texto plagiado** no d_l, sendo $b_h{:}B_h$ o início (b_h) e o fim (B_h) do *h-ésimo* excerto de texto plagiado no *l-ésimo* documento fonte d_l do D_{fonte}. Para simplificar a notação e abstrair dos detalhes dos limites dos excertos, representa-se, sempre que se torne conveniente, o par de excertos de texto por $r_h = \langle s_a, s_b \rangle$, sendo $s_a = \delta_i^{a_h:A_h}$ e $s_b = d_l^{b_h:B_h}$.

4.1 Métricas Heurísticas

Nesta secção aborda-se a exploração heurística de novas métricas de deteção de plágio entre dois excertos a partir das combinações heurísticas das principais técnicas de deteção de similaridade textual.
Numa primeira tentativa de encontrar uma nova *métrica de deteção de plágio* robusta e capaz de reconhecer corretamente as *paridades textuais* e as *divergências textuais* criaram-se novas métricas heurísticas a partir das principais técnicas de deteção de similaridade textual encontradas na literatura: similaridade documental [Hua08], similaridade vetorial [ASA12] e deteção

[9] *I.e.*, casos de plágio, zonas de plágio, passagens de plágio, pares de plágio, *etc*.

de paráfrases [CDB07b, CDB07c]. Testou-se o desempenho de cada métrica desse conjunto para uma pequena amostra de excertos de plágio com ofuscação textual, extraídos dos exemplos fornecidos por [PSBR10]. Depois, analisaram-se as características das métricas com melhor desempenho. Por último, reuniram-se as técnicas consideradas mais relevantes e combinando-as heuristicamente.

Inicialmente, identificaram-se uma conjunto de métricas de similaridade documental [Hua08], de similaridade vetorial [ASA12] e de deteção de paráfrases [CDB07b, CDB07c], capazes de detetar a similaridade textual entre excertos de plágio. Nesse conjunto, composto por mais de duas dezenas de métricas, as principais métricas são: o *coeficiente de Dice* [ASA12]; o *coeficiente de sobreposição* [ASA12] (*i.e., "Overlap (or containment) coefficient."* [ASA12]); a *distância Euclidiana* [ASA12]; a *distância de Manhattan* [ASA12]; a função entrópica [Sha48, CDB07c]; a *distância de Levenshtein* [Lev66, CDB07c]; a *função Gaussiana* [CDB07c]; a *métrica sumo* [CDB07a, CDB07b]); o *coeficiente de Jaccard*[10] [Jac01a, Jac01b, ASA12]; o *coeficiente do cosseno* [ASA12]; e a *métrica de palavra-n-gramas*[11] [LMD01, CDB07b]. Seguidamente, experimentou-se o desempenho de cada métrica desse conjunto para uma amostra de excertos de plágio, baseada nos exemplos fornecidos por [PSBR10]. Houve três métricas que se destacaram perante as demais na deteção das similaridades textuais entre excertos de plágio, nomeadamente: a *métrica de palavra-n-gramas*; o *coeficiente de Jaccard*; e o *coeficiente do cosseno*. A seguir, são analisadas as técnicas de deteção de paridades textuais e de divergências textuais dessas três métricas.

A *métrica palavra-n-gramas*) utiliza uma técnica de deteção de similaridade textual baseada nas sequências de palavras comuns entre dois excertos. Nesta técnica, definida na Equação (4.1):

$$palavra\text{-}n\text{-}gramas(x,\, y) = \frac{1}{n} \times \sum_{k=1}^{n} \frac{|x \wedge y|_k}{\min(|x|_k,\, |y|_k)} \quad (4.1)$$

o x e o y representam dois excertos de texto; o n representa o número máximo de sequências de k palavras; o $|x|_k$ representa o número de sequências de k palavras no x; assim como, o $|y|_k$ representa o número de sequências de k palavras no y; e o $|x \wedge y|_k$ representa o número de sequências de k palavras comuns ao x e ao y. Quando as divergências textuais são causadas pela reordenação das palavras, apenas as sequências comuns menores são contabilizadas com o "peso" $\frac{1}{n}$. Como no *exemplo*, obtido de [Val11], apresentado a seguir:

```
The dog hit a man. (x)
The man hit a dog. (y)
```

onde o $x = \{$*The dog bit a man.*$\}$; e onde o $y = \{$*The man bit a dog.*$\}$. Nesse exemplo, para o $n = 5$, é atribuído um peso 0.2 para cada conjunto de sequências k palavras comuns entre o x e o y, tal que $1 \leq k \leq 5$ e $palavra\text{-}5\text{-}gramas(x,\, y) = \frac{1}{5} \times \left(\frac{5}{5} + \frac{1}{5} + \frac{0}{5} + \frac{0}{5} + \frac{0}{5} \right)$, reduzindo, assim, o valor final de similaridade para 0.24. Para o exemplo dado, com poucas palavras por excerto, essa problemática de "peso" pode ser facilmente resolvida, através da redução do valor de n, de cinco para dois, consequentemente havendo um aumento no valor do "peso", de 0.2 para 0.5, assim como do valor de similaridade, de 0.24 para 0.6; entretanto, para excertos de textos com um número elevado de palavras reordenadas a métrica apresenta algumas dificuldades.

[10] Também conhecida como *o coeficiente de Tanimoto* [Tan58]. [ASA12]

[11] *Word n-gram*.

A métrica do *coeficiente de Jaccard* utiliza um técnica de deteção de similaridade textual baseada na proporção do vocabulário partilhado entre dois excertos. Nesta técnica, definida na Equação (4.2):

$$J(x,\ y) = \frac{|x \cap y|}{|x \cup y|} \quad (4.2)$$

o x e o y representam dois excertos de texto; o $|x \cap y|$ representa o vocabulário partilhado entre o x e o y; e o $|x \cup y|$ representa o vocabulário do x e do y (*i.e.*, a união). Assim, as paridades textuais são determinadas através da proporção do vocabulário partilhado, entre os excertos, em função do vocabulário existente nos excertos. Entretanto, quando as divergências textuais são causadas pela substituição das palavras por outras equivalentes, apenas o vocabulário partilhado é contabilizado. Como no *exemplo*, inspirado em [CDB07b], apresentado a seguir:

```
The President and Commander in Chief of The United States of America ordered the assault. (x)
                          The President ordered the final strike over the terrorists camp. (y)
```

onde o $|x \cap y| = 3$ e o $|x \cup y| = 17$. Nesse exemplo, o $J(x,\ y) = 0.18$, o valor da similaridade textual não evidencia as divergências textuais causada pela *ofuscação textual*. Contudo, pensou-se ser possível contornar o problema através da utilização de um *dicionário de sinónimos*[12].
A métrica do *coeficiente do cosseno* utiliza uma técnica de deteção de similaridade textual baseada na proporção de ocorrências das palavras dos excertos em função do vocabulário "global" dos dois excertos. Nessa técnica, definida na Equação (4.3):

$$Cos(x,\ y) = \frac{\sum_i (x_i \times y_i)}{\sqrt{\sum_i (x_i)^2} \times \sqrt{\sum_i (y_i)^2}} \quad (4.3)$$

o x e o y representam dois excertos de texto; o x_i representa o número vezes que a *i-ésima* palavra do vocabulário "global" repete-se no x; assim como, o y_i representa o número vezes que a *i-ésima* palavra do vocabulário "global" repete-se no y. Assim, as paridades textuais são determinadas através do ângulo do cosseno entre cada vetor dos dois excertos de texto em função do vocabulário de ambos os excertos. Entretanto, quando as divergências textuais são causadas pela substituição das palavras por outras equivalentes, essa proporção de ocorrências pode não o suficiente. Como no *exemplo*, obtido de [PSBR10], seguinte:

```
Dogs are lazy which is why brown foxes quickly jump over them. (x)
                The quick brown fox jumps over the lazy dog. (y)
```

onde o $\sum_i (x_i \times y_i) = 3$, a $\sqrt{\sum_i (x_i)^2} = \sqrt{12}$, e a $\sqrt{\sum_i (y_i)^2} = \sqrt{11}$. Nesse exemplo, o $Cos(x,\ y) = 0.26$, demonstrando que, assim como no exemplo anterior, o valor de similaridade textual não evidencia as divergências textuais causada pela *ofuscação textual*. Porém, pensou-se ser possível contornar o problema através da utilização dos *radicais das palavras*[13].
Consequentemente, exploraram-se novas métricas, inspiradas na combinação heurística das três técnicas de deteção de similaridade textual para reconhecer as paridades textuais entre dois

[12] *E.g.*, O *WordNet* [https://wordnet.princeton.edu/].
[13] *I.e.*, stemming: remoção dos sufixos das palavras. [Por06, Por80]

excertos. Assim, adotou-se a técnica utilizada pela métrica *palavra-n-gramas* para reconhecer paridades textuais através de várias sequências de palavras comuns aos dois excertos de texto. Porém, introduziu-se uma heurística de *atribuição binária de pesos*, que favorece as sequências menores de palavras comuns aos excertos. A partir da combinação entre a *deteção do vocabulário partilhado* e a *deteção do número de ocorrências de uma palavra em um excerto de texto*, introduziu-se uma heurística de *limitação do tamanho máximo das sequências de palavras em análise* (*i.e.*, o valor de *n*, ver Equação (4.1)), atribuindo automaticamente ao *n* o mínimo múltiplo comum de palavras partilhadas entre dois excertos. Para reconhecer as *divergências textuais* entre dois excertos, adotou-se um *dicionário de sinónimos* [Shi14, Tea07, Mil95] e uma função *stemming* [Por80], bem como a remoção das *palavras comuns* (*i.e.*, os *termos vulgares*[14]). Finalmente, a partir das combinações heurísticas das principais técnicas de deteção de similaridade textual, criaram-se as métricas heurísticas *nBinGram* e *nBinGramPlus*, apresentadas a seguir.

A métrica *nBinaryGram* utiliza uma técnica de deteção de similaridade textual baseada na contagem de sequências de palavras comuns entre dois excertos, sendo uma combinação geométrica de três rácios, para cada *k* considerado. Cada parcela da soma perde importância [BR09], uma vez que esta vai sendo dividida por 2^k [FC15]. Nessa técnica, definida nas Equações (4.4a) e (4.4b):

$$nBinGram(x,\, y) = \sum_{k=1}^{n} \frac{\sqrt[3]{\frac{|x \wedge y|_k}{|x|_k} \times \frac{|x \wedge y|_k}{|y|_k} \times \frac{\min(|x|_k,\, |y|_k)}{\max(|x|_k,\, |y|_k)}}}{2^k} \quad (4.4a)$$

$$n = \min\left(\sum_{i=1}^{|x \cap y|} (x \cap y)_i \times x_i,\ \sum_{i=1}^{|x \cap y|} (x \cap y)_i \times y_i \right) \quad (4.4b)$$

o *x* representa um excerto suspeito de plágio; o *y* representa um excerto fonte possivelmente plagiado; o *k* representa o comprimento da sequência em palavras; o $x \cap y$ representa o vocabulário partilhado entre o *x* e o *y*; o $|x \cap y|$ representa o número de palavras do $x \cap y$; o *i* representa o *i-ésimo* elemento de um conjunto; o $(x \cap y)_i$ representa a *i-ésima* palavra do $x \cap y$, tal que $1 \leq i \leq |x \cap y|$, o x_i representa a *i-ésima* palavra do *x*; o y_i representa a *i-ésima* palavra do *y*; o $\sum_{i=1}^{|x \cap y|} (x \cap y)_i \times x_i$ representa o somatório do número de vezes que cada palavra do $x \cap y$ ocorre em *x*; o $\sum_{i=1}^{|x \cap y|} (x \cap y)_i \times y_i$ representa o somatório do número de vezes que cada palavra do $x \cap y$ ocorre em *y*; o $|x|_k$ representa o número de sequências de *k* palavras no *x*; o $|y|_k$ representa o número de sequências de *k* palavras no *y*; e o $|x \wedge y|_k$ representa o número de sequências de *k* palavras comuns ao *x* e ao *y*. Assim, a métrica *nBinGram*, qualifica a similaridade textual entre um *x* e um *y*, através da quantificação das paridades textuais entre *x* e *y*, com um valor real compreendido entre zero (0) (*i.e.*, ausência de) e um (1) (*i.e.*, existência de). Os valores de similaridade textual para os excertos, *x* e *y*, considerados no primeiro, no segundo e no terceiro exemplo, apresentados anteriormente, são de $5BinGram(x, y) = 0.60$, $5BinGram(x, y) = 0.31$ e $3BinGram(x, y) = 0.20$, respetivamente. [FC15]

Inspirada na *nBinGram*, pensou-se numa versão mais elaborada, designada de métrica *nBinGramPlus* (Equações (4.5a), (4.5b), (4.5c), (4.5d), (4.5e), (4.5f), (4.5g), (4.5h) e (4.5i)), na

[14] Stopwords. [Uni15]

qual considerou-se os radicais das palavras e os vocabulários dos segmentos x e y.

$$nBinGramPlus(x,\ y) = \sum_{k=1}^{n} \frac{\sqrt[3]{\frac{|\theta_x \wedge \theta_y|_k}{|\theta_x|_k} \times \frac{|\theta_x \wedge \theta_y|_k}{|\theta_y|_k} \times \frac{\min(|\theta_x|_k,\ |\theta_y|_k)}{\max(|\theta_x|_k,\ |\theta_y|_k)}}}{2^k} \quad (4.5a)$$

$$n = \min\left(\sum_{i=1}^{|\theta_x \cap \theta_y|} (\theta_x \cap \theta_y)_i \times \theta_{x_i},\ \sum_{i=1}^{|\theta_x \cap \theta_y|} (\theta_x \cap \theta_y)_i \times \theta_{y_i} \right) \quad (4.5b)$$

$$\theta_x = V_{(xy)} + stem(V_{(xy)}) + \left(syns(U_{(x)}) \cap syns(U_{(y)})\right) + U'_{(x)} + stem(U'_{(x)}) \quad (4.5c)$$

$$\theta_y = V_{(xy)} + stem(V_{(xy)}) + \left(syns(U_{(x)}) \cap syns(U_{(y)})\right) + U'_{(y)} \cup stem(U'_{(y)}) \quad (4.5d)$$

$$V_{(xy)} = x \cap y \quad (4.5e)$$

$$U_{(x)} = V_{(x)} \setminus V_{(y)} \quad (4.5f)$$

$$U_{(y)} = V_{(y)} \setminus V_{(x)} \quad (4.5g)$$

$$U'_{(x)} = U_{(x)} \setminus V_{(sw)} \quad (4.5h)$$

$$U'_{(y)} = U_{(y)} \setminus V_{(sw)} \quad (4.5i)$$

Aqui, $\theta_x \cap \theta_y$ representa o vocabulário partilhado entre o θ_x e o θ_y; $(\theta_x \cap \theta_y)_i$ representa a *i-ésima* palavra do $\theta_x \cap \theta_y$; θ_{x_i} representa a *i-ésima* palavra do θ_x; θ_{y_i} representa a *i-ésima* palavra do θ_y; $\sum_{i=1}^{|\theta_x \cap \theta_y|} (\theta_x \cap \theta_y)_i \times \theta_{x_i}$ representa o somatório do número de vezes que cada palavra do $\theta_x \cap \theta_y$ ocorre em θ_{x_i}; $\sum_{i=1}^{|\theta_x \cap \theta_y|} (\theta_x \cap \theta_y)_i \times \theta_{y_i}$ representa o somatório do número de vezes que cada palavra do $\theta_x \cap \theta_y$ ocorre em θ_{y_i}; $|*|_k$ representa o número de sequências de k palavras em $*$; $V_{(xy)}$ representa o vocabulário partilhado $x \cap y$; $V_{(x)}$ representa o vocabulário do x; $V_{(y)}$ representa o vocabulário do y; $V_{(sw)}$ representa o vocabulário *de palavras comuns e frequentes* (*i.e.*, o vocabulário de *termos vulgares*[15], *e.g.*, "de", "um", "para", "que", etc), tal que $|V_{(sw)}| = 733$ [Uni15]; $U_{(x)}$ representa o vocabulário único do $V_{(x)}$; $U_{(y)}$ representa o vocabulário único do $V_{(y)}$; $U'_{(x)}$ representa o vocabulário único e relevante do $V_{(x)}$; $U'_{(y)}$ representa o vocabulário único e relevante de $V_{(y)}$; $stem(.)$ representa a função de *extração dos radicais das palavras*[16]; $syns(.)$ representa a função de *extração de sinónimos, hipónimos, hiperónimos, etc* [Shi14, Tea07, Mil95]; θ_x representa a reestruturação textual do x; e θ_y representa a reestruturação textual do y. Logo, a métrica *nBinGramPlus*, qualifica a similaridade textual entre um x e um y, através da quantificação das *paridades textuais* e das *divergências textuais* entre os excertos reestruturados de x e de y, com um valor real compreendido entre zero (0.0) (*i.e.*, ausência de) e um (1.0) (*i.e.*, existência de). Consequentemente, os valores de similaridade textual para os excertos, x e y, considerados no primeiro, no segundo e no terceiro exemplo, apresentados anteriormente nesta secção, são de $10BinGramPlus(x, y) = 1.00$, $10BinGramPlus(x, y) = 0.56$ e $137BinGramPlus(x, y) = 0.98$, respetivamente.

Com as métricas *nBinGram*(*., .*) e *nBinGramPlus*(*., .*) conclui-se a exploração de novas métricas de deteção de plágio a partir das principais métricas de deteção de similaridade textual existentes. A Secção 4.2 reporta a exploração de novas métricas inteligentes de deteção de plágio, através da extração de características textuais de plágio através da indução, com a programação genética, de padrões de classificação específicos de plágio que possam determinar a existência ou não deste.

[15] Stopwords. [Uni15]

[16] *I.e.*, stemming: remoção dos sufixos das palavras. [Por06, Por80]

4.2 Indução de Métricas Inteligentes de Deteção

Nesta secção aborda-se a exploração novas métricas inteligentes para a deteção de plágio entre dois excertos de texto. A fim de encontrar uma nova *métrica de deteção de plágio* robusta e capaz de reconhecer corretamente as *paridades textuais* e as *divergências textuais* na *pesquisa e análise de plágio* entre os excertos de texto de um documento suspeito δ_k e os excertos de texto de um documento candidato d_l, exploraram-se novas métricas inteligentes, com a extração de características textuais de plágio e a indução de padrões de classificação de plágio. Inicialmente, na Subsecção 4.2.1, utilizou-se um conjunto de dados com plágio, constituído por dois *corpora de plágio*, para extrair um conjunto de características textuais de plágio potencialmente relevantes para identificar regularidades típicas da ofuscação textual. Posteriormente, na Subsecção 4.2.2, utilizou-se este conjunto de características para induzir *padrões de classificação* com o recurso da programação genética, e consequente determinação da existência de plágio.

4.2.1 Extração de Características Textuais

A fim de extrair as características léxico-gramaticais, implícitas do comportamento humano do infrator na perpetração do plágio e na tentativa de ocultação do delito com a ofuscação textual, que possam servir para identificar regularidades típicas do plágio, extraíram-se características textuais de plágio a partir de um conjunto de dados com plágio. Inicialmente, utilizaram-se as referências de plágio pertencentes a dois *corpora* de plágio [PEB+11], para serem a base de extração das características; para cada referência identificou-se as "fontes de extração", e a parir destas, extraiu-se um conjunto de características de plágio.

Utilizando um conjunto de dados de plágio, composto por dois *corpora* de plágio, obteve-se um conjunto de referências de plágio. Cada *referência de plágio* é constituída por um *excerto com plágio* (s_a) do *documento com plágio* (δ_i) e por um *excerto plagiado* (s_b) do *documento fonte* (d_l) relacionados. Assim, em cada referência identificaram-se quatro "elementos" para a extração de características, nomeadamente s_a, s_b, δ_i e d_l. Destes, identificaram-se oito relações para a extração de características: a relação de identidade $\left(s_a \cap (\delta_i \cap d_l)\right.$[17] e $\left.s_b \cap (\delta_i \cap d_l)\right)$, a relação de exclusividade $\left(s_a \backslash s_b\right.$[18], $s_b \backslash s_a$, $\delta_i \backslash d_l$ e $\left.d_l \backslash \delta_i\right)$ e a relação de partilha $\left(s_a \cap s_b\right.$[19] e $\left.\delta_i \cap d_l\right)$. Logo, ***para cada referência de plágio*** identificaram-se doze "***fontes de extração***" (*i.e.*, os elementos mais as relações), como apresentado na Tabela 4.1, para serem utilizadas com as "métricas de extração" e com as "técnicas de extração" para extrair ***um conjunto de características de plágio***.

Utilizou-se um conjunto de "métricas de extração" de características de similaridade textual para cada referências de plágio. Este conjunto é constituído por métricas de similaridade documental [Hua08], similaridade vetorial [ASA12], deteção de paráfrases [CDB07b, CDB07c] e métricas heurísticas [Secção 4.1], capazes de detetar a similaridade textual e expressa-las em valores característicos. Neste conjunto de métricas (mais de vinte) destaca-se as seguintes: o *coeficiente de Dice* [ASA12]; o *coeficiente de sobreposição* [ASA12]; a *distância Euclidiana* [ASA12]; a *distância de Manhattan* [ASA12]; a função entrópica [Sha48, CDB07c]; a *distância de Levenshtein* [Lev66, CDB07c]; a *função Gaussiana* [CDB07c]; a *métrica sumo* [CDB07a, CDB07b]);

[17] *E.g.*, Se $A = [0, 1, 1, 2, 2, 3, 4]$, $B = [1, 2, 2, 3, 3, 5, 6]$ e $C = [-1, 0, 1, 2, 2, 7, 8, 8]$, e se $A \cap B = [1, 2, 2, 3]$, então $C \cap (A \cap B) = [1, 2, 2]$.

[18] *E.g.*, Se $A = [0, 1, 1, 2, 2, 3, 4]$ e $B = [1, 2, 2, 3, 3, 5, 6]$, então $(A \setminus B) = [0, 4]$.

[19] *E.g.*, Se $A = [0, 1, 1, 2, 2, 3, 4]$ e $B = [1, 2, 2, 3, 3, 5, 6]$, então $A \cap B = [1, 2, 2, 3]$.

o *coeficiente de Jaccard* [Jac01a, Jac01b, ASA12]; o *coeficiente do cosseno* [ASA12]; a *métrica de palavra-n-gramas* [LMD01, CDB07b]; a *métrica nBinGram* [Secção 4.1]; e a *métrica nBinGramPlus* [Secção 4.1]. Com este conjunto de "métricas de extração" aplicado aos excertos s_a e s_b de ***cada referência de plágio*** extraíram-se as ***características de similaridade textual***.

Utilizando um conjunto de "técnicas de extração" extraíram-se as principais características de plágio das "fontes de extração" de cada referências de plágio. Cada "técnica de extração" tentou explorar uma característica léxico-gramatical específica, implícita nos "elementos" (*i.e.*, s_a, s_b, δ_i e d_l) e/ou nas "relações" entre os "elementos" (*i.e.*, $s_a \cap s_b$, $s_a \setminus s_b$, $s_a \cap (\delta_i \cap d_l)$, $s_b \setminus s_a$, $s_b \cap (\delta_i \cap d_l)$, $\delta_i \cap d_l$, $\delta_i \setminus d_l$ e $d_l \setminus \delta_i$) de uma referência de plágio. As principais características extraídas foram o *vocabulário* $V_{(.)}$, a *extração das palavras comuns e frequentes*[20,21] $SW_{(.)}$, a *extração das palavras incomuns e infrequentes*[22] $\overline{SW}_{(.)}$, a *extração dos radicais das palavras*[23] $stem_{(.)}$, e a *extração dos sinónimos*[24] *das palavras* $syns_{(.)}$. Assim, na Tabela 4.1 são apresentadas as principais características de plágio extraídas na aplicação das "técnicas de extração" às respetivas "fontes de extração" de uma referência de plágio, onde "✓" representa uma caraterística específica extraída [FC15].

Tabela 4.1: Principais caraterísticas de plágio extraídas de uma referência de plágio.

"Técnicas de Extração"	**"Fontes de Extração"**											
	δ_i	s_a	s_b	d_l	$\delta_i \setminus d_l$	$\delta_i \cap d_l$	$d_l \setminus \delta_i$	$s_a \cap (\delta_i \cap d_l)$	$s_a \setminus s_b$	$s_a \cap s_b$	$s_b \setminus s_a$	$s_b \cap (d_l \cap \delta_i)$
$\lvert V_{(.)} \rvert$	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
$\lvert SW_{(V_{(.)})} \rvert$	✓	✓	✓	✓	✓		✓		✓	✓	✓	
$\lvert \overline{SW}_{(V_{(.)})} \rvert$	✓	✓	✓	✓	✓		✓		✓	✓	✓	
$\lvert stem_{(V_{(.)})} \rvert$		✓	✓						✓	✓	✓	
$\lvert stem_{(SW_{(V_{(.)})})} \rvert$		✓	✓						✓	✓	✓	
$\lvert stem_{(\overline{SW}_{(V_{(.)})})} \rvert$		✓	✓						✓	✓	✓	
$\lvert syns_{(V_{(.)})} \rvert$									✓		✓	
$\lvert syns_{(SW_{(V_{(.)})})} \rvert$									✓		✓	
$\lvert syns_{(\overline{SW}_{(V_{(.)})})} \rvert$									✓		✓	

Portanto, a aplicação das técnicas e métricas de extração permitiu obter conjuntos de características de plágio e formar as instâncias utilizadas na indução com programação genética (Subsecção 4.2.2). Guardaram-se essas instâncias com o formato dados *arff*[25] definido pela ferramenta de mineração de dados *Weka*[26]. Cada instância guardada representa um r_h de $refs_{(\delta_i)}$ e cada atributo de instância representa uma caraterística de plágio de r_h, potencialmente relevante para identificar regularidades típicas do plágio e da ofuscação textual.

[20] *I.e.*, os *termos vulgares*.
[21] Stopwords. [Uni15]
[22] *I.e.*, os termos invulgares. [Pri16]
[23] *I.e.*, stemming: remoção dos sufixos das palavras. [Por06, Por80]
[24] Assim como, hipónimos, hiperónimos, etc. [Shi14, Tea07, Mil95]
[25] Attribute-relation file format.
[26] http://www.cs.waikato.ac.nz/ml/weka/

4.2.2 Indução com Programação Genética

Com o objetivo de identificar regularidades típicas do plágio, implícitas, do comportamento humano do infrator na perpetração do plágio e na tentativa de ocultação do delito com a ofuscação textual, nas características textuais de plágio, exploraram-se métricas inteligentes baseadas na indução com a programação de padrões de classificação do plágio, a partir das regularidades típicas do plágio encontradas nas características textuais de plágio extraídas, para assim determinar a existência de plágio entre excertos suspeitos. Primeiro, apresenta-se uma pequena introdução aos principais conceitos relacionados com a programação genética. Segundo, utiliza-se a programação genética para encontrar, as principais regularidades típicas do plágio contidas nas características textuais de plágio extraídas, e apresentam-se as árvores de plágio que expressam essas regularidades. Por último, induzem-se padrões de classificação de plágio com base das regularidades típicas do plágio identificadas, apresentam-se as métricas inteligentes que expressam esses padrões.

Inicialmente, recorreu-se à ***inteligência computacional*** para utilizar as capacidades de uma classe especial de ***algoritmos evolucionários***[27] para encontrar regularidades típicas num conjunto de dados específico através da denominada ***programação genética***. Na ***inteligência computacional*** são utilizados *modelos algorítmicos* [Egg05], baseados nos fenómenos biológicos adaptativos de sistemas naturais [Egg05, Ale11a, Wik14], para solucionar *problemas complexos* [Egg05, Ale11a, Wik16]. Na ***computação evolucionaria*** são utilizados ***algoritmos evolucionários*** [Egg05, Ale11b], baseados no processo de ***evolução natural das espécies*** [Egg05, Dar59], para resolver *problemas de pesquisa* [Egg05, Ale11b, RLS+04]. Neste algoritmo os ***indivíduos*** representam as possíveis soluções para um *problema específico*. Cada *indivíduo* é constituído por valores característicos que representaram os parâmetros necessários para a resolução do problema. Esses ***indivíduos competem***, pela ***sobrevivência*** e pela ***reprodução***, durante ***várias gerações***, sendo apenas os ***mais aptos*** capazes de sobreviver e de se reproduzir [Egg05, Ale11b, Dar59]. A ***aptidão*** de cada indivíduo é definida pela ***natureza do problema***, com os indivíduos aptos representando os valores característicos que mais se aproximam da solução ótima do problema em questão. Na ***última geração***, o ***indivíduo mais apto*** representa a ***solução ótima*** segundo a especificidade do problema, sendo os valores característicos desse indivíduo representantes dos parâmetros necessários à resolução desse problema [Egg05, Ale11b, Dar59]. Assim, nos ***algoritmos genéticos***, cada ***indivíduo*** é representado como um ***cromossoma*** e os *valores característicos* de um indivíduo são representados como os ***genes*** de um cromossoma (*i.e.*, um ***indivíduo com valores característicos*** é um ***cromossoma com genes***). Cada cromossoma representa uma possível solução para um problema[28] de otimização[29] e de tal forma que, ao longo de várias gerações, o cromossoma mais apto é aquele que, com os *valores* dos seus genes, melhor solucionar (*i.e.*, maximizar ou minimizar) o problema (relação entre os parâmetros relevantes) [Egg05, Ale11c]. Na ***programação genética***, cada ***indivíduo*** é representado como

[27] A *computação evolucionária* é um ramo *inteligência computacional*. Assim como, *inteligência computacional* é um ramo da *inteligência artificial*. [Egg05]

[28] Nesse contexto, um problema "genérico" pode ser definido como a relação entre um conjunto de parâmetros relevantes, sendo o valor cada parâmetro desconhecido. Assim, nesse problema são conhecidos os parâmetros relevantes, assim como é conhecida a relação entre esses parâmetros relevantes; mas desconhecem-se os valores de cada parâmetro. Logo, a solução desse problema reside na pesquisa e otimização de um conjunto de valores para os parâmetros relevantes, para máxima ou minimizar a sua relação.

[29] *E.g.*, pesquisar as dimensões de uma lata cilíndrica com capacidade para 2 litros, sendo a base um disco, de modo a minimizar os custos do material utilizado. [Ba16]

uma ***árvore***[30] e os *valores característicos* de um indivíduo são representados como os ***nodos*** e as ***folhas*** de uma árvore [Egg05, Ale11d, Spe04] e, contrariamente aos *algoritmos genéticos*, cada árvore representa, não uma possível solução (os valores para os parâmetros relevantes) para o problema, mas sim o problema em si (relação entre os parâmetros relevantes). Ao longo de várias gerações, a árvore mais apta é aquela que, com os parâmetros relevantes expressos nas suas folhas e com a relação entre os parâmetros relevantes expressa em seus nodos, melhor representar (regularidades típicas) os valores (os dados) da solução (de um conjunto de dados específico). Logo, a ***programação genética*** permite encontrar ***regularidades típicas*** nos dados de um ***conjunto de dados específico*** (as relações entre os dados relevantes) e expressa-las na ***linguagem simbólica*** de ***árvores*** constituídas por *nodos*, com ***funções matemáticas*** (as relações), e por *folhas*, com os ***os dados relevantes*** (os parâmetros relevantes), como apresentado a seguir.

Consequentemente, tirou-se partido da capacidade de encontrar regularidades típicas nos dados e expressa-las com uma linguagem simbólica inteligível por humanos [BB07]. Nessa linguagem simbólica, cada árvore é constituída por nodos (*i.e.*, funções e/ou estruturas de decisão) e por folhas (*i.e.*, variáveis e/ou constantes) [Egg05, Eng07], como no exemplo apresentado na Figura 4.2. Os nodos podem interligar outros nodos e/ou podem ligar folhas; cada nodo é um elemento

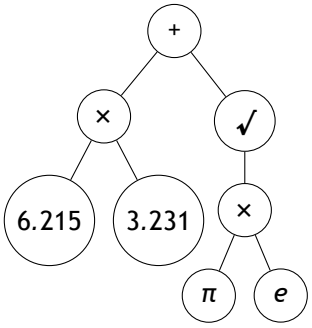

Figura 4.2: Exemplo de uma árvore na programação genética.

de um conjunto de funções matemáticas (*e.g.*, exp, log, sin, cos, *pow*, $\sqrt{}$, min, max, *etc*); operadores aritméticos (*e.g.* +, −, /, ×, *etc*); operadores booleanos (*e.g.*, ∧, ∨, ⊕, ¬, *etc*); estruturas de decisão e operadores relacionais (*e.g.*, *se-então-senão*, >, <, *etc*). Cada folha é um elemento de um conjunto terminal constituído por constantes (*e.g.*, π, *e*, φ, 1.414, *etc*) e por variáveis (*e.g.*, *x*, *y* e *z*). Assim, decidiu-se tirar partido da programação genética para encontrar regularidades típicas do plágio nas características de plágio extraídas (instâncias e atributos de instância), induzindo padrões de classificação específicos do plágio e da ofuscação textual, como descrito a seguir.

Aplicando a programação genética em concreto, obteve-se duas regularidades típicas de plágio para as características textuais, previamente extraídas (Subsecção 4.2.1), correspondendo a duas árvores de plágio aptas segundo programação genética. A primeira árvore, intitulada por ***árvore de plágio gp24n***, pode ser observada na Figura 4.3 [FC15]. A regularidade representada pela árvore da Figura 4.3 consiste numa expressão matemática envolvendo operadores aritméticos e relacionais, funções matemáticas e características de plágio, nomeadamente a *nBinGram*

[30] Originalmente representada com um programa executável [Koz89, Koz90], composto por conjunto de funções aplicáveis a um conjunto terminal de variáveis e constantes. [Ale11d, Egg05]

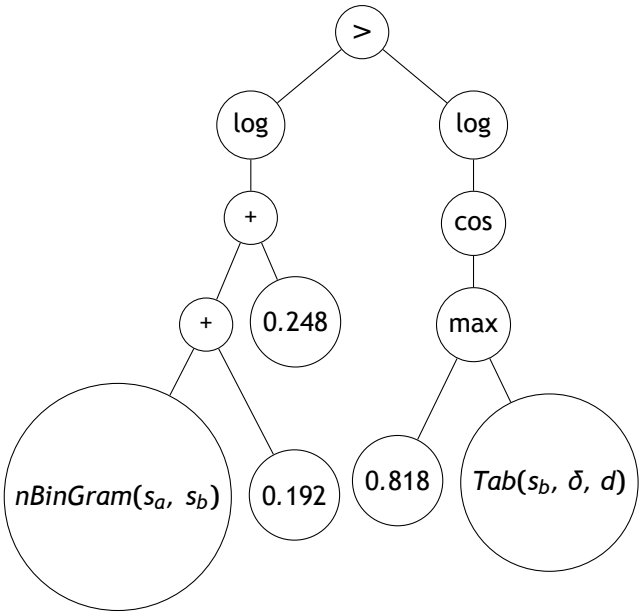

Figura 4.3: *Árvore de plágio gp24n.* [FC15]

(Secção 4.1) e uma nova característica designada por *Tab* (Equação (4.6)).

$$Tab(s_b,\ \delta,\ d) = \frac{|V_{(s_b \cap \delta \cap d)}|}{|V_{(s_b)}|} \quad (4.6)$$

O s_b representa um *excerto plagiado* do *documento fonte*, o δ representa o *documento com plágio* e o d representa o *documento fonte plagiado*. A função $V_{(.)}$ representa o vocabulário de texto em argumento, como habitualmente já mencionado.

A segunda árvore induzida com a programação genética foi a denominada ***gp24p***, apresentada na Figura 4.4. Assim, estas árvores representam as regularidades típicas de plágio induzidas das

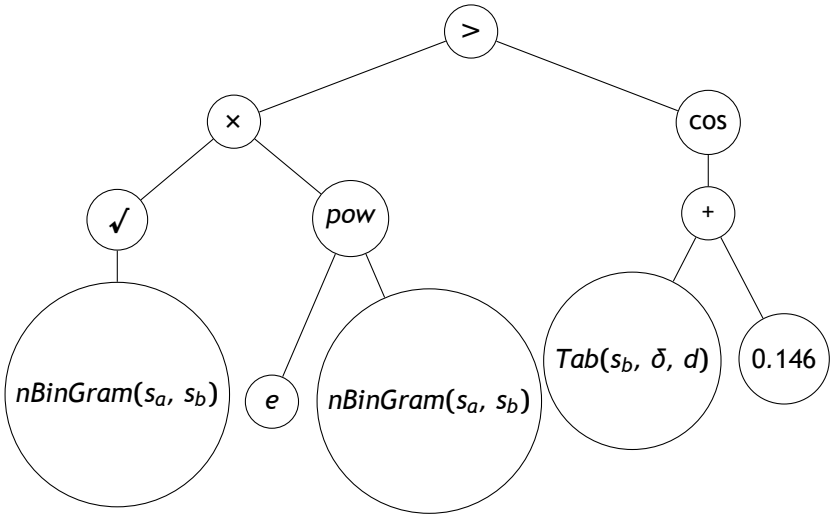

Figura 4.4: *Árvore de plágio gp24p.*

características de plágio extraídas, que podem ser utilizadas na classificação de plágio.

As duas árvores induzidas são representadas pelas funções ou métricas de classificação de plágio, mostradas nas Equações (4.7) e (4.8).

$$GPIM24n(x,\ y,\ \delta,\ d) = \begin{cases} 1, & \text{se } \log\big(0.44 + nBinGram(x,\ y)\big) > \log\Big(\cos\big(\max\big(0.818,\ Tab(y,\ \delta,\ d)\big)\big)\Big) \\ 0, & \text{caso contrário} \end{cases} \tag{4.7}$$

$$GPIM24p(x,\ y,\ \delta,\ d) = \begin{cases} 1, & \text{se } \big(\sqrt{nBinGram(x,\ y)} \times e^{nBinGram(x,\ y)}\big) > \cos\big(Tab(y,\ \delta,\ d) + 0.146\big) \\ 0, & \text{caso contrário} \end{cases} \tag{4.8}$$

Teve-se assim duas métricas de classificação de excertos de plágio, induzidas automaticamente, a partir dos dados reais de plágio.
Com as métricas *GPIM24n* e *GPIM24p* conclui-se a exploração inteligente de novas métricas de deteção de plágio, através da indução com a programação genética.

4.3 Pesquisa e Análise de plágio

Esta secção aborda a metodologia de deteção de plágio em pares de documentos. Começa-se por analisar os excertos de texto propícios de terem sido plágio e depois aplica-se uma métrica de deteção de plágio. Os excertos identificados como plágio são coletados para apresentação e confirmação por uma avaliador humano.
O Algoritmo 4.1 descreve a identificação do(s) *excerto(s) com plágio* no documento suspeito e o(s) respetivo(s) *excerto(s) plagiado(s)* nos documentos candidatos (*i.e.*, δ *versus* $\forall d_l \in \Sigma_{(\delta)}$), através da pesquisa pelos excertos propícios de plágio em cada par de documentos e através da analise dos indícios nesses excertos. Portanto, primeiro reduz-se o espaço de procura com a seleção de excertos propícios e depois aplica-se uma métrica de deteção de plágio.

Algoritmo 4.1: Algoritmo de pesquisa e análise de plágio.

```
 1: D_font = {d_1, d_2, ..., d_|D_font|}
 2: δ = [x_1, x_2, ..., x_m]
 3: Σ_(δ) = {d_1, d_2, ..., d_ω}
 4: r̂_h ← ⟨⟩
 5: refs̃_(δΣ_(δ)) ← ∅
 6: for l ← 1 to ω do
 7:    d_l = [y_1, y_2, ..., y_n]
 8:    V_(δd_l) ← ∅
 9:    V_(δd_l) ← redução(δ, d_l, D_font, V_(δ) ∩ V_(d_l))
10:    d_l ← filtragem(d_l, V_(δd_l), ψ)
11:    δ ← filtragem(δ, V_(δd_l), ψ)
12:    for i ← 1 to m do
13:       for j ← 1 to n do
14:          if |V_(x_i) ∩ V_(y_j)| ≥ ψ then
15:             if Métrica(x_i, y_j) > λ then
16:                r̂_h ← ⟨δ^(x_i), d_l^(y_j)⟩
17:                refs̃_(δΣ_(δ)) ← refs̃_(δΣ_(δ)) ∪ {r̂_h}
18:
19: return refs̃_(δΣ_(δ))
```

Inicialmente, utilizaram-se os documentos fonte, do conjunto de documentos candidatos $\Sigma_{(\delta)}$

identificado no Capítulo 3, relacionados com um documento suspeito δ, para pesquisar em cada par de documentos, δ versus d_l, pelos excertos propícios de serem *excertos de plágio*, por partilharem um vocabulário tido como relevante. Assim, primeiro, obteve-se o vocabulário partilhado por cada par de documentos (*i.e.*, $V_{(\delta)} \cap V_{(d_l)}$). A seguir, utilizou-se a métrica FIT (Equação (3.2)) para estimar a importância de cada palavra do vocabulário partilhado em função de cada documento do par, (*i.e.*, suspeito e candidato) e em função da *corpora* de documentos fonte (D_{font}). As palavras mais relevantes foram utilizadas para criar o vocabulário relevante partilhado ($V_{(\delta d_l)}$), conforme apresentado no Algoritmo 4.2.

Algoritmo 4.2: Algoritmo de redução.

$D_{font} = \{d_1, d_2, \ldots, d_{|D_{font}|}\}$
$\delta = [x_1, x_2, \ldots, x_{|\delta|}]$
$d_l = [y_1, y_2, \ldots, y_{|d_l|}]$
$V_{(\delta \cap d_l)} = \{w_1, w_2, \ldots, w_{|V_{(\delta \cap d_l)}|}\}$
$V_{(\delta d_l)} \leftarrow \emptyset$
for $i \leftarrow 1$ *to* $|V_{(\delta \cap d_l)}|$ **do**
    if $FIT(w_i|\delta) \geq 0 \wedge FIT(w_i|d_l) \geq 0 \wedge w_i \notin V_{(SW)} \wedge |w_i| \geq 3$ **then**
        $V_{(\delta d_l)} \leftarrow V_{(\delta d_l)} \cup \{w_i\}$
return $V_{(\delta d_l)}$

O $V_{(\delta d_l)}$ do par foi utilizado para identificar a sua distribuição entre os excertos do par de documentos, ou seja, utilizando vocabulário relevante partilhado pelo par e utilizando o vocabulário de cada excerto[31] dos documentos, analisou-se o vocabulário relevante por excerto, para os excertos do par, em função vocabulário relevante partilhado pelo par.
Com vocabulário relevante por excerto foi possível encontrar os excertos propícios de serem excertos de plágio, conforme apresentado no Algoritmo 4.3. Assim um excerto foi considerado propício de ser de plágio se a sua proporção do vocabulário relevante for superior ou igual a ψ; onde ψ representa, uma constante natural definida a *priori*, assim como o número mínimo de palavras relevantes no vocabulário relevante por excerto.

Algoritmo 4.3: Algoritmo de filtragem.

$d = [z_1, z_2, \ldots, z_{|d|}]$
$V_{(\delta d_l)} = \{w_1, w_2, \ldots, w_{|V_{(\delta d_l)}|}\}$
$\widehat{d} \leftarrow [\,]$
for $i \leftarrow 1$ *to* $|d|$ **do**
    if $V_{(z_i)} \cap V_{(\delta d_l)} \geq \psi$ **then**
        $\widehat{d} \leftarrow \widehat{d} + [z_i]$
return $\widehat{d}$

Uma vez identificados os excertos propício de serem excertos de plágio, nos pares, esses excertos foram submetidos a um métrica de deteção de plágio para a análise dos indícios de plágio entre os excertos; conforme apresentado no Algoritmo 4.1. Assim, se os indícios de plágio, de um par de excertos considerados propícios, ultrapassarem o limiar mínimo de paridades textuais e/ou divergências textuais estabelecido, para a métrica de deteção de plágio, a um λ, onde λ representa uma constante real definida a priori, então esses excertos, considerados como excertos de plágio, são identificados e apresentados para a confirmação final do plágio por um avaliador humano.

[31] Aqui a noção de um excerto no documento pode ser compreendida com uma frase no documento.

Deste modo conclui-se a descrição do método de análise detalhada proposto. A seguir, na Secção 4.4, descrevem-se as experiências realizadas e os resultados obtidos com este método.

4.4 Experiências e Discussão

Nesta secção aborda-se as experiências realizadas e os resultados obtidos para o segundo ato da metodologia descrita anteriormente. Primeiro, descreve-se o *corpus*, os métodos de avaliação [Rij79], as ferramentas e os parâmetros utilizados para desenvolver e testar a nossa metodologia. No final, apresentam-se os resultados de avaliação obtidos.

4.4.1 Conjunto de Dados

Nesta subsecção aborda-se o *conjunto de dados* utilizados para desenvolver, testar e avaliar o método de análise detalhada proposto. Inicialmente, descreve-se a divisão do *conjunto de dados* em dois subconjunto distintos, nomeadamente, o conjunto de dados de treino, e o conjunto de dados teste, utilizados na avaliação do método proposto. Seguidamente, descrevem-se os *corpora* que fazem parte de cada um dos conjuntos. Simultaneamente, descreve-se *um por um*, os elementos constituintes dos corpora; descreve-se e quantifica-se a estrutura aglomerativa de seus documentos, e os diferentes géneros de documentos.
Com o objetivo de desenvolver, de testar e de avaliar o método de recuperação fonte proposto, utilizou-se um conjunto de dados constituído por dois *corpora de plágio* escritos na língua inglesa. O conjunto de dados utilizado, é constituído por *corpora de plágio* criados para treinar e testar algoritmos de deteção de plágio externo em duas competições internacionais de deteção de plágio [PEB$^+$11, PGH$^+$13]. Inicialmente, dividiu-se esse conjunto de dados em dois subconjuntos distintos: o conjunto de dados de treino para ser utilizado no treino do método proposto; e o conjunto de dados de teste para ser utilizado na avaliação do método proposto. Assim, no *conjunto de treino* utilizou-se o *corpus de plágio PAN-PC-11*[32] da competição [PEB$^+$11]. Para *o conjunto de teste* utilizou-se o *corpus de plágio PAN13-TextAlignment*[33] da competição [PGH$^+$13]. Nesses *corpora de plágio* os dados são apresentados em ficheiros de texto com a extensão *txt*[34], e as referências de plágio são descritas em ficheiros com extensão *xml*[35], que elucidam os limites (*i.e.*, início e fim) e os atores (*i.e.*, origem e destino) intervenientes no plágio [PEB$^+$11, PGH$^+$13].
No *conjunto de treino* utilizou-se o *corpus de plágio PAN-PC-11* para se obter um conjunto de ***corpora* de plágio** constituído por dois *corpora*, tal que $corpora = \{D_{font}, D_{sus}\}$, de tal forma que $corpora = D_{font} \cup D_{sus}$ e $D_{font} \cap D_{sus} = \emptyset$. O ***corpus* fonte** D_{font} é constituído por um conjunto de documentos fonte, com 11053 documentos originais que possivelmente foram alvo de plágio. O ***corpus* suspeito** D_{sus} é constituído por dois *"sub-corpora"* menores, de tal forma que $D_{sus} = D_{white} \cup D_{black}$ e $D_{white} \cap D_{black} = \emptyset$. Sendo a distinção entre D_{white} e D_{black} dada pela existência de referência(s) de plágio nos ficheiros *xml* dos respetivos documentos, logo considerou-se que, para um δ, a existência de referência(s) de plágio seu ficheiros *xml* é dada por uma ***função de valoração*** $f(.)$ que determina se $f(\delta_k) = white$ ou $f(\delta_k) = black$. O ***corpus* suspeito sem plágio** D_{white} é constituído por um conjunto de documentos suspeitos, com 5546

[32] https://www.uni-weimar.de/en/media/chairs/webis/corpora/pan-pc-11
[33] https://web.archive.org/web/20160901172225/http://www.uni-weimar.de/medien/webis/corpora/corpus-pan-labs-09-today/pan-13/pan13-data/pan13-text-alignment-test-corpus2-2013-01-21.zip
[34] Plain text file.
[35] Extensible markup language.

documentos que, apesar de serem suspeitos, são de facto originais, tal que $D_{white} = \{\delta_1, \delta_2, ..., \delta_{5546}\}$ e $\forall\delta_j\ f(\delta_j) = white$. O ***corpus* suspeito com plágio** D_{black} é constituído por um conjunto de documentos suspeitos, com 4992 documentos que, além de serem suspeitos, possuem plágio, obtido dos documentos fonte que foram plagiados, tal que $D_{black} = \{\delta_1, \delta_2, ..., \delta_{4992}\}$ e $\forall\delta_i\ f(\delta_i) = black$. **O plágio** do D_{black} é constituído por um conjunto ***referências de plágio***, com aproximadamente 44479 ***referências***[36], tal que $refs_{(D_{black})} = \{r_1, r_2, ..., r_{44479}\}$. [PEB+11]
No *conjunto de teste* utilizou-se o *corpus de plágio PAN13-TextAlignment* para se obter um conjunto de ***corpora* de plágio** constituído por dois *corpora*, de tal forma que $corpora = D_{font} \cup D_{sus}$ e $D_{font} \cap D_{sus} = \emptyset$. O ***corpus* fonte** D_{font} é constituído por um conjunto de documentos fonte, com 3169 documentos originais que possivelmente foram alvo de plágio. O ***corpus* suspeito** D_{sus} é constituído por dois *"sub-corpora"* menores, de tal forma que $D_{sus} = D_{white} \cup D_{black}$ e $D_{white} \cap D_{black} = \emptyset$. Sendo a distinção entre D_{white} e D_{black} dada pela existência de referência(s) de plágio nos ficheiros *xml* dos respetivos documentos, logo considerou-se que, para um δ_k, a existência de referência(s) de plágio seu ficheiros *xml* é dada por uma ***função de valoração*** $f(.)$ que determina se $f(\delta_k) = white$ ou $f(\delta_k) = black$. O ***corpus* suspeito sem plágio** D_{white} é constituído por um conjunto de documentos suspeitos, com 1093 documentos que, apesar de serem suspeitos, são de facto originais, tal que $D_{white} = \{\delta_1, \delta_2, ..., \delta_{1093}\}$ e $\forall\delta_j\ f(\delta_j) = white$. O ***corpus* suspeito com plágio** D_{black} é constituído por um conjunto de documentos suspeitos, com 733 documentos que, além de serem suspeitos, possuem plágio, obtido dos documentos fonte que foram plagiados, tal que $D_{black} = \{\delta_1, \delta_2, ..., \delta_{733}\}$ e $\forall\delta_i\ f(\delta_i) = black$. **O plágio** do D_{black} é constituído por um conjunto ***referências de plágio***, com 4042 ***referências***[37], tal que $refs_{(D_{black})} = \{r_1, r_2, ..., r_{4042}\}$. [PGH+13]
Com os *corpora PAN-PC-11* e *PAN13-TextAlignment* avaliou-se as métricas heurísticas e as métricas inteligentes, assim como a pesquisa e análise de plágio, conforme apresentado na Secção 3.4.4.

4.4.2 Implementação

Com o objetivo de suprir as necessidades funcionais de implementação das métricas heurísticas, das métricas inteligentes e da pesquisa e análise de plágio do método de análise detalhada proposto, adotaram-se os recursos disponibilizados por um conjunto de ferramentas preexistentes. Essas ferramentas foram adotadas por possuírem um atrativo conjunto de vantagens inerentes às suas utilizações, sendo, assim, consideradas úteis para auxiliar a implementação do método proposto. Essas vantagens são, nomeadamente: ser implementada na linguagem de programação *Java* [GJS+15]; ter o *código aberto*[38] [Ini07]; ter o código disponível na *Internet*; ter uma documentação associada ao código (*e.g.*, classes, funções, *etc*), às soluções disponibilizadas (*e.g.*, o os resultados que a ferramenta permite alcançar) e aos exemplos de utilização da ferramenta; e ser robusta perante outras ferramentas do mesmo género (*e.g.*, escalabilidade, popularidade, *etc*).
Inicialmente, adotou-se a biblioteca *HultigLib*[39] [Cor06] como a ferramenta base, para processar o conjunto de dados (i.e., os *corpora de plágio*). Essa biblioteca fora concebida com intuito de proporcionar eficiência e escalabilidade no processamento de grandes volumes de dados textuais

[36] Sendo uma referência de plágio igual a um excerto de texto com plágio no documento com plágio e a um excerto de texto plagiado no documento fonte.
[37] Sendo uma referência de plágio igual a um excerto de texto com plágio no documento com plágio e a um excerto de texto plagiado no documento fonte.
[38] Open-source.
[39] http://www.di.ubi.pt/ jpaulo/hultiglib/

[Cor06], fazendo uso de um vasto conjunto de recursos disponibilizado por ferramentas externas (*e.g.*, a biblioteca *Apache OpenNLP*[40] [Com14]), para além dos recursos implementados [Cor06]. Utilizou-se e aperfeiçoou-se, sempre que possível e necessário, os algoritmos implementados nos métodos de leitura de ficheiros de texto, nos métodos de análise sintática de fases e palavras, nos métodos de representação textual com estruturas de dados e nos métodos de manipulação das estruturas de dados textuais. Com o auxílio desta biblioteca, primeiro, leram-se os dados textuais de cada documento como um conjunto ordenado de caracteres alfanuméricos. Segundo, normalizou-se o conjunto de caracteres através da conversão das letras maiúsculas para letras minúsculas. A seguir, segmentou-se o conjunto de caracteres em frases e palavras, através da identificação dos caracteres de pontuação (*e.g.*, o ponto final e o espaço), recorrendo aos *dicionários normativos de construção frásica da língua inglesa*[41] da biblioteca *Apache OpenNLP*. Por último, a partir dos excertos frásicos detetados, criou-se uma representação do texto com estruturas de dados [BT08]. Consequentemente, um documento "genérico" d_i foi representado por uma *lista ligada* com *n nodos*, sendo o *n-ésimo nodo* correspondente a *n-ésima* frase do documento, tal que $d_i = [s_1, s_2, ..., s_n]$. Por sua vez, uma frase "genérica" s_i foi representada por uma *lista ligada* com *n nodos*, sendo o *n-ésimo nodo* correspondente a *n-ésima* palavra da frase, tal que $s_i = [w_1, w_2, ..., w_n]$. Por fim, uma palavra "genérica" w_i foi representada por um *conjunto ordenado de n caracteres*, sendo o *n-ésimo* elemento correspondente ao *n-ésimo* caractere, tal que $w_i = [c_1, c_2, ..., c_n]$.

Na implementação das *métricas heurísticas*, definidas nas Equações (4.4a), (4.4b), (4.5a), (4.5b), (4.5c), (4.5d), (4.5e), (4.5f), (4.5g), (4.5h) e (4.5i), utilizaram-se, e aprimoram-se para as necessidades em questão, os algoritmos de processamento de *corpora de plágio* da biblioteca *HultigLib* [Cor06], para a manipulação de uma pequena amostra de excertos de plágio com ofuscação textual extraídos dos exemplos fornecidos por [PSBR10]. Na implementação do conjunto de métricas de similaridade documental [Hua08], de similaridade vetorial [ASA12] e de deteção de paráfrases [CDB07b, CDB07c], utilizaram-se, e aperfeiçoaram-se sempre que necessário para as necessidades em questão, as implementações disponibilizadas pela biblioteca *HultigLib* [Cor06], nomeadamente, as das métricas deteção de paráfrases e as de algumas métricas de similaridade documental, sendo as principais: a *métrica sumo* [CDB07a, CDB07b]; a função entrópica [Sha48, CDB07c]; a *distância de Levenshtein* [Lev66, CDB07c]; a *função Gaussiana* [CDB07c]; e a *métrica de palavra-n-gramas*[42] [LMD01, CDB07b]. Para definir um vocabulário de *palavras comuns e frequentes* adotou-se um lista de palavras disponibilizada por [Uni15], com 733 *termos*. Na implementação da função de *extração dos radicais das palavras*[43], utilizou-se o algoritmo de extração dos radicais definido por Porter (1980) e a sua implementação criada por Porter (2006). Na implementação da função de *extração de sinónimos, hipónimos, hiperónimos, etc*, utilizou-se a base de dados lexical da língua inglesa *WordNet*[44] [Mil95], a implementação da biblioteca *JWNL* [Tea07] de acesso ao dicionário relacional da *WordNet*, e a implementação dos exemplos de utilização da biblioteca *JWNL* e da *WordNet* [Shi14].

Na implementação das *métricas inteligentes*, definidas nas Equações (4.7) e (4.8), inicialmente, implementou-se a *extração de características textuais de plágio* e, posteriormente, implementou-se a indução com programação genética. Para armazenar as características tex-

[40] https://opennlp.apache.org/
[41] http://opennlp.sourceforge.net/models-1.5/
[42] *Word n-gram*.
[43] *I.e.*, stemming: remoção dos sufixos das palavras.
[44] https://wordnet.princeton.edu/

tuais de plágio extraídas, como instâncias e atributos de instâncias, utilizou-se o formato *ARFF*[45] associado ao *software* de mineração de dados *Weka*[46] [WF05]; assim como utilizou-se o *Weka* para manipular as instâncias e os atributos de instâncias extraídos [WF05].

Na implementação da *indução com programação genética* utilizou-se a ferramenta , com o código aberto, desenvolvida por Yan Levasseur (2010). Esta ferramenta, originalmente desenvolvida para o reconhecimento de objetos biológicos em imagens bidimensionais [Lev08], constitui uma implementação eficaz do algoritmo de programação genética [Lev08, Lev10]. A sua utilização está associada ao *Weka* como uma espécie de extensão. Assim o *Weka* faz a ligação entre a ferramenta de programação genética [Lev10] e as caraterísticas textuais de plágio extraídas e armazenadas como instâncias e atributos de instâncias num ficheiro *ARFF* do *Weka*. Assim, induziram-se 184 padrões[47] de classificação de plágio, com programação genética, a partir das caraterísticas textuais de plágio extraídas em combinação com a variação de diferentes opções da ferramenta de programação genética [Lev10, Lev08]. Estes opções (e respetivas variações) são: o número de gerações (20, 100, 200, 1000 ou 10000); o tamanho da população (25, 50, 100 ou 1000); os tipos de reprodução (*cross-over*, mutação, mutação de nodos funcionais, reprodução e/ou novo programa [Lev08]); os tipos de seleção (torneio ou proporcional ao valor de aptidão[48]); os operadores funcionais ($+$, $-$, $/$, $\times$, exp, log, sin, cos, *pow*, $\sqrt{}$, min, max, $\wedge$, $\vee$, $\oplus$, $\neg$, *se-então-senão*, $>$ e/ou $<$ [Lev08]); os avaliadores de aptidão (soma de erros - contínuo [Lev08], raiz quadrada do erro quadrático médio - contínuo [Lev08]; ou confiança no reconhecimento da classe - classificador híbrido com impulsão[49] [FS96, Lev08]); entre outros fatores.

Na implementação da *pesquisa e análise de plágio*, definida nos Algoritmos 4.1, 4.2 e 4.3, utilizaram-se, e aprimoram-se para as necessidades em questão, os algoritmos de processamento de *corpora de plágio* da biblioteca *HultigLib* [Cor06], para a manipulação do documentos suspeito δ, do conjunto de documentos candidatos $\Sigma_{(\delta)}$ e da *corpora* de documentos fonte D_{font}. Para a pesquisa e análise de plágio utilizou-se a métrica *nBinGram* e definiram-se como opções de configuração os valores de $\psi = 3$ e $\lambda = 0.3$.

A confluência das implementações destes algoritmos com as ferramentas externas adotadas mais os valores definidos para os parâmetros, formam a implementação do método de análise detalhada proposto. Apresenta-se a seguir os métodos utilizados para avaliar a estratégia de deteção de plágio do método proposto.

4.4.3 Métodos de Avaliação

Esta subsecção aborda[50] o conjunto de métodos de avaliação adotado para avaliar e validar o método de análise detalhada proposto. Primeiro, apresenta-se o conjunto de métodos de avaliação, a sua divisão em dois conjuntos distintos de métricas de avaliação, assim como o racional desta divisão. A seguir, descreve-se a utilização destes conjuntos na análise dos resultados obtidos nas diferentes fases de criação do método proposto, e com diferentes conjuntos de dados.

[45] Attribute-Relation File Format

[46] http://www.cs.waikato.ac.nz/ml/weka/

[47] *E.g.*, $\log\big(0.44 + nBinGram(x, y)\big) > \log\Big(\cos\big(\max\big(0.818, Tab(y, \delta, d)\big)\big)\Big)$.

[48] Fitness.

[49] Boosting.

[50] Alguns métodos de avaliação descritos nessa subsecção já foram abordados anteriormente, nomeadamente na Subsecção 3.4.3; assim, devido a natureza incremental dos métodos de avaliação e do conjuntos de dados utilizados, reutilizou-se alguns métodos de avaliação descritos anteriormente. [PGH+13]

Por último, apresentam-se e descrevem-se todas as métricas de avaliação dos dois conjuntos adotados.

O conjunto de métodos de avaliação divide-se em dois conjuntos distintos de métricas de avaliação. O primeiro [KBLP55, Rij79] foi utilizado para avaliar os resultados obtidos nas fases de desenvolvimento, implementação e teste das métricas heurísticas e das métricas inteligentes, a partir da experimentação destas métricas nos dados de treino (*i.e.*, o *corpus de plágio PAN-PC-11*, descritos na Subsecção 4.4.1). O segundo conjunto de métricas de avaliação [PGH$^+$13] foi utilizado para validar os resultados obtidos na pesquisa e análise de plágio com o método proposto e o conjunto de dados de teste (*i.e.*, o *corpus de plágio PAN13-TextAlignment*, descrito na Subsecção 4.4.1). A utilização de um segundo conjunto de métricas de avaliação permite corroborar que as métricas heurísticas e as métricas inteligentes utilizadas na pesquisa e análise do plágio no método proposto não se encontram "superajustadas"[51] [GCF$^+$15] ao conjunto de dados de treino, assim como permite corroborar que essas não se encontram "subajustadas"[52] [GCF$^+$15] ao conjunto de dados de teste, e por fim, também, permite viabilizar a comparação entre o método proposto e os demais métodos de análise detalhada [PGH$^+$13], pelo facto de ambos utilizarem o mesmo conjunto de métricas de avaliação e o mesmo conjunto de dados [PGH$^+$13].

O primeiro conjunto de métricas de avaliação, adotado para avaliar os resultados obtidos no desenvolvimento, implementação e teste do método proposto com o conjunto de dados de treino, é constituído por quatro métricas de avaliação. Estas métricas são, nomeadamente, o *Precision*, o *Recall*, o *F-Measure*, e a *Accuracy*. Utilizaram-se estas métricas pelo facto de, estas, serem as *normas de facto*[53] na avaliação de *sistemas de pesquisa de informação*[54] [KBLP55, Rij79, AAA11] e em *sistemas de aprendizagem automática*[55] [WF05]. Estas métricas são, nomeadamente, o $Precision = \frac{TP}{TP+FP}$, o $Recall = \frac{TP}{TP+FN}$, $F\text{-}Measure = \frac{(1+\beta^2)\times Precision\times Recall}{\beta\times Precision+Recall}$ e a $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$; onde, TP representa os verdadeiros positivos, FP representa os falsos positivos, TN representa os verdadeiros negativos e FN representa os falsos negativos; com $\beta = 1$ para a equidade de pesos entre o *Precision* e o *Recall*.

O segundo conjunto de métricas de avaliação, adotado para validar os resultados obtidos nas fases de teste e validação da pesquisa e análise de plágio do método proposto com o conjunto de dados de teste, é constituído por seis métricas de avaliação, nomeadamente, o *prec*, o *rec*, o *"s⊓r"*, o *gran*, o F_1 e o *plagdet*. Estas métricas, originalmente definidas em [PSBR10] e utilizadas por [PGH$^+$13], são derivações ajustadas do ***P**recision*, do ***R**ecall* e do ***F-Measure***, criadas para medir a performance de algoritmos de análise detalhada. A utilização do segundo conjunto de métricas de avaliação permitiu comparar os resultados obtidos com os de outros [PGH$^+$13], uma vez que o conjunto de dados, também, é igual [PGH$^+$13]. Nas Equações (4.9a), (4.9b), (4.9c),

[51] Overfitting.
[52] Underfitting.
[53] De facto standard – https://en.wikipedia.org/wiki/De_facto_standard
[54] Information retrieval systems.
[55] Machine learning systems.

(4.9d), (4.9e) e (4.9f) são apresentadas as métricas de avaliação do segundo conjunto

$$prec(S, R) = \frac{1}{|R|} \times \sum_{r \in R} \frac{\left| U_{s \in S}(s \sqcap r) \right|}{|r|} \quad (4.9a)$$

$$rec(S, R) = \frac{1}{|S|} \times \sum_{s \in S} \frac{\left| U_{r \in R}(s \sqcap r) \right|}{|s|} \quad (4.9b)$$

$$s \sqcap r = \begin{cases} s \cap r, & \text{se } r \text{ detetar } s \\ \emptyset, & \text{caso contrário} \end{cases} \quad (4.9c)$$

$$gran(S, R) = \frac{1}{|S_R|} \times \sum_{s \in S_R} |R_s| \quad (4.9d)$$

$$F_1(S, R) = \frac{2 \times prec(S, R) \times rec(S, R)}{prec(S, R) + rec(S, R)} \quad (4.9e)$$

$$plagdet(S, R) = \frac{F_1}{\log_2\left(1 + gran(S, R)\right)} \quad (4.9f)$$

onde: o S representa o conjunto de referências de plágio pertencentes ao D_{black}, tal que $\forall \delta_i \in D_{black} \Rightarrow f(\delta_i) = D_{black}$; o R representa o conjunto de *excertos de plágio* detetado na *pesquisa e análise de plágio* do método proposto, pertencentes ao D_{sus}; o s representa o **alinhamento entre dois excertos de plágio** de **uma referência de plágio** r_h; o r representa o **alinhamento entre dois excertos de plágio** (*i.e.*, x e y) detetado na *pesquisa e análise de plágio* do método proposto $\widehat{r_h}$, tal que $r \in R$ e de tal forma que $r = \widehat{r_h} = \langle \delta_k^{x_i}, d_l^{y_j} \rangle$; o $S_R \subseteq S$ representa as referências de plágio do D_{black} que foram detetadas *na pesquisa e análise de plágio*, tal que $S_R = \{s \mid s \in S \wedge \exists r \in R : r \text{ detetou } s\}$; o $R_S \subseteq S$ representa os excertos de plágio detetados *na pesquisa e análise de plágio*, tal que $R_S = \{r \mid r \in R \wedge r \text{ detetou } s\}$. [PGH+13].
Com a utilização dos métodos de avaliação, supracitados, e suas respetivas métricas de avaliação foi possível avaliar o método de análise detalhada proposto segundo as normas padrões de avaliação [KBLP55, Rij79, WF05], assim como validar o método proposto segundo as normas de avaliação de uma competição internacional do ramo da deteção de plágio [PGH+13].

4.4.4 Resultados Experimentais

Esta subsecção aborda os resultados experimentais obtidos com o método de análise detalhada proposto.
A fim de avaliar o desempenho das métricas heurísticas e das métricas inteligentes do método de análise detalhada proposto com o conjunto de dados de treino (*i.e.*, o *corpus de plágio PAN-PC-11*), realizam-se duas experiências com o conjunto de documentos suspeitos $D_{sus} \in$ *PAN-PC-11*: a primeira para o conjunto de documentos suspeitos com plágio confirmado $D_{black} \in D_{sus}$; e a segunda para o conjunto de documentos suspeitos sem plágio $D_{white} \in D_{sus}$.
Na primeira experiência, inicialmente, utilizou-se o *conjunto de dados de treino* para extrair as *referências de plágio* com mais de 1000 e menos de 2000 caracteres, ou seja, extraíram-se as ***referências de plágio*** $\left(refs_{(\delta)} = \{r_1, r_2, \ldots, r_p\}\right)$ dos ***documentos suspeitos com plágio*** $\left(D_{black} = \{\delta_1, \delta_2, \ldots, \delta_n\}\right)$ em que cada *referência de plágio* $r_h = \langle \delta^{a_h:A_h}, d^{b_h:B_h} \rangle = \langle s_a, s_b \rangle$ possui ***mais de*** 1000 ***caracteres e menos de*** 2000 ***caracteres***: $1000 < |s_a| < 2000$ e $1000 < |s_b| < 2000$. Ao conjunto dessas referências de plágio designou-se como ***"classe*** 1***"***. Submeteram-se as referências da "classe 1" às ***métricas heurísticas*** e às ***métricas inteligentes*** para a deteção da existência de plágio. Finalmente, contabilizaram-se os ***sucessos*** e os ***insucessos*** das métricas

heurísticas e das métricas inteligentes do método proposto em ***detetarem que a existência de plágio*** nas referências de plágio da "classe 1".

Na segunda experiência, com o *corpus de plágio PAN-PC-11*, fez-se o *teste de despistagem de erros* para as *métricas heurísticas* e as *métricas inteligentes* do método proposto. O objetivo deste teste consistiu em *determinar o fator de aleatoriedade dos "resultados" devolvidos* pelas métricas, ou seja, determinar se as métricas heurísticas e as métricas inteligentes detetaram corretamente a existência de plágio ou se essas deteções provêm de fatores desconhecidos. Para tal utilizou-se o *corpus suspeito sem plágio* e os *identificadores de início e fim de plágio* das *referências de plágio* do *corpus suspeito com plágio*, para simular *"pseudo-referências" de "pseudo-plágio"*[56] com mais de 1000 e menos de 2000 caracteres ou seja, simularam-se as ***"pseudo-referências" de "pseudo-plágio"*** $\left(\widetilde{refs}_{(\delta)} = \{\widetilde{r}_1, \widetilde{r}_2, \ldots, \widetilde{r}_p\}\right)$ dos ***documentos suspeitos sem plágio*** $\left(D_{white} = \{\delta_1, \delta_2, \ldots, \delta_t\}\right)$ sendo cada *"pseudo-referência" de "pseudo-plágio"* $\widetilde{r}_h = \langle \delta^{a_h:A_h}, d^{b_h:B_h} \rangle = \langle \widetilde{s}_a, \widetilde{s}_b \rangle$ com $\delta \in D_{white}$, $d \in D_{font}$ e os limites $a_h{:}A_h$ e $b_h{:}B_h$ escolhidos de um documento *aleatório*[57,58] do D_{black}, assim como com $1000 < |\widetilde{s}_a| < 2000$ e $1000 < |\widetilde{s}_b| < 2000$. Ao conjunto dessas "pseudo-referências" de "pseudo-plágio" designou-se como ***"classe 0"***. Seguidamente, submeteram-se as "pseudo-referências" da "classe 0" as ***métricas heurísticas*** e as ***métricas inteligentes*** para determinar o fator de aleatoriedade dos "resultados" devolvidos. Finalmente, contabilizaram-se os ***sucessos*** e os ***insucessos*** (*i.e.*, o fator de aleatoriedade) das métricas heurísticas e das métricas inteligentes do método proposto em ***não detetarem que a existência de plágio*** nas "pseudo-referências" de "pseudo-plágio" da "classe 0".

Tabela 4.2: *Matriz confusão* das detenções de plágio com a *métrica heurística nBinGram* em função das *classes de referências*.

		nBinGram(*., .*)		
		1	0	Total de referências:
Classes de Referências	1	3716	128	3844
	0	4	4477	4481
Total de deteções:		3720	4605	

Na Tabela 4.2 apresentam-se os valores da *matriz confusão*, obtidos nas duas experiências supracitados, a primeira com as referências de plágio da classe 1 e a segunda com as "pseudo-referências" de "pseudo-plágio" da classe 0, com a *métrica heurística nBinGram*, definida nas Equações (4.4a) e (4.4b), e um limiar (*i.e.*, *threshold*) de paridade textual estabelecido[59], para essa métrica, de 0.3. Assim, evidencia-se que, para as 3844 referências de plágio da classe 1 submetidas a *métrica heurística nBinGram*, houve 3716 referências da classe 1 que a *nBinGram*(*., .*) *detetou corretamente a existência de plágio* e para as 4481 "pseudo-referências" de "pseudo-plágio" da classe 0 houve 4477 "pseudo-referências" da classe 0 que a *nBinGram*(*., .*) *detetou corretamente a inexistência de plágio*.

[56] *I.e.*, um alinhamento entre dois excertos de texto, sendo cada um pertence a um documentos texto diferentes, no qual não existe plágio entre os excertos, assim como não existe plágio entre os documentos.

[57] Segundo uma distribuição uniforme.

[58] ***int*** *random* = ***new*** *Random().nextInt(*$|D_{black}|$*);*

[59] Com o *algoritmo de classificação J48* [Qui93] e com a validação cruzada em blocos de 10 (*i.e.*, *10-fold cross-validation*) [Cor03] dos "resultados" $\left(e.g., J(x, y) = 0.12345; i.e., \forall IR \in [0, 1]\right)$ devolvidos pela(s) métrica(s) em função da classe de cada referência (*i.e.*, "0 ⊕ 1") das referências das classes. [WF05]

Tabela 4.3: *Matriz confusão* das detenções de plágio com a *métrica heurística nBinGramPlus* em função das *classes de referências*.

nBinGramPlus(*., .*)

		1	0	Total de referências:
Classes de Referências	1	3418	426	3844
	0	176	4305	4481
Total de deteções:		3594	4731	

Na Tabela 4.3 apresentam-se os valores da *matriz confusão*, obtidos nas duas experiências supracitadas, a primeira com as referências de plágio da classe 1 e a segunda com as "pseudo-referências" de "pseudo-plágio" da classe 0, com a *métrica heurística nBinGramPlus*, definida nas Equações (4.5a), (4.5b), (4.5c), (4.5d), (4.5e), (4.5f), (4.5g), (4.5h) e (4.5i), e um limiar (*i.e., threshold*) de paridade e divergência textual estabelecido[59], para essa métrica, de 0.808.

Tabela 4.4: *Matriz confusão* das detenções de plágio com a *métrica inteligente GPIM24n* em função das *classes de referências*.

*GPIM*24*n*(*., ., ., .*)

		1	0	Total de referências:
Classes de Referências	1	3804	40	3844
	0	896	3585	4481
Total de deteções:		4700	3625	

Na Tabela 4.4 apresentam-se os valores da *matriz confusão*, obtidos nas duas experiências supracitadas, a primeira com as referências de plágio da classe 1 e a segunda com as "pseudo-referências" de "pseudo-plágio" da classe 0, com a *métrica inteligente GPIM24n*, definida na Equação (4.7), e sua atribuição, exclusiva ("$\oplus$"), do valor 0 (inexistência de plágio) ou do valor 1 (existência de plágio).

Tabela 4.5: *Matriz confusão* das detenções de plágio com a *métrica inteligente GPIM24p* em função das *classes de referências*.

*GPIM*24*p*(*., ., ., .*)

		1	0	Total de referências:
Classes de Referências	1	3766	78	3844
	0	471	4010	4481
Total de deteções:		4237	4088	

Na Tabela 4.5 apresentam-se os valores da *matriz confusão*, obtidos nas duas experiências supracitadas, a primeira com as referências de plágio da classe 1 e a segunda com as "pseudo-referências" de "pseudo-plágio" da classe 0, com a *métrica inteligente GPIM24p*, definida na Equação (4.8), e sua atribuição, exclusiva ("$\oplus$"), do valor 1 (*i.e.*, existência de plágio) ou do valor 0 (inexistência de plágio).

Na Tabela 4.6 apresentam-se os resultados de avaliação do primeiro conjunto de métricas de avaliação, ver Subsecção 4.4.3, obtidos nas duas experiências supracitadas, 3844 referências de

Tabela 4.6: Resultados de avaliação das *métricas de deteção de plágio* para as *classes referências*.

Métricas	Precision	Recall	F-Measure	Accuracy	Classes
nBinGram(*., .*)	**0.999**	0.967	**0.983**	**0.984144**	**1**
	0.972	**0.999**	**0.985**		**0**
nBinGramPlus(*., .*)	0.951	0.889	0.919	0.927688	1
	0.910	0.961	0.935		0
*GPIM*24*n*(*., ., ., .*)	0.809	**0.990**	0.890	0.887568	**1**
	0.989	0.800	0.885		**0**
*GPIM*24*p*(*., ., ., .*)	0.889	0.980	0.932	0.934054	1
	0.981	0.895	0.936		0
J(*., .*)	0.996	0.967	0.981	0.982943	1
	0.972	0.997	0.984		0
Cos(*., .*)	0.952	0.854	0.900	0.912432	1
	0.885	0.963	0.922		0
palavra-2-gramas(*., .*)	0.996	0.954	0.975	0.977297	1
	0.962	0.997	0.979		0

plágio da classe 1 e 4481 "pseudo-referências" de "pseudo-plágio" da classe 0, com as métricas heurísticas, as métricas inteligentes e algumas das melhores métricas de similaridade textual. Na primeira coluna dessa tabela, apresentam-se: as duas *métricas heurísticas*, nomeadamente, a *nBinGram*(*., .*) e a *nBinGramPlus*(*., .*), as duas *métricas inteligentes GPIM*24*n*(*., ., ., .*) e *GPIM*24*p*(*., ., ., .*); e as três *métricas de similaridade textual*, consideradas como umas das melhores [ASA12], nomeadamente, a métrica do *coeficiente de Jaccard*, a métrica do *coeficiente do cosseno* e a *métrica de palavra-n-gramas*[60] (com $n = 2$ [BR09]), definidas nas Equações (4.2), (4.3) e (4.1), com os limiares (*i.e.*, *thresholds*) de deteção de paridades textuais, estabelecidos[59] em 0.176, em 0.779 e em 0.251, respetivamente. Deste modo destaca-se a supremacia dos resultados obtidos pela métrica *nBinGram*, quer em termos de **F-Measure**, quer em termos de **Accuracy**.

A fim de validar o desempenho da *pesquisa e análise de plágio* (Algoritmo 4.3) do método proposto com o *conjunto de dados de teste* (*i.e.*, o *corpus de plágio PAN13-TextAlignment*), realizou-se uma experiência com o *corpus suspeito* D_{sus}, tal que $D_{sus} \in$ *PAN13-TextAlignment*.

Nessa terceira experiência, inicialmente, utilizou-se o *conjunto de dados de treino* (Subsecção 4.4.1) para obter os documentos suspeitos $D_{sus} = \{\delta_1, \delta_2, ..., \delta_{1826}\}$ e os respetivos conjuntos de documentos candidatos. Seguidamente, submeteram-se os documentos suspeitos e candidatos ao Algoritmo 4.3 para a ***pesquisa e análise de plágio*** com a métrica *nBinGram* (Equações (4.4a) e (4.4b)), $\psi = 3$ e $\lambda = 0.3$ (limiar mínimo de deteção de plágio da métrica). Finalmente, contabilizaram-se os ***sucessos*** e os ***insucessos*** da *pesquisa e análise de plágio* do *método de análise detalhada proposto* com o segundo conjunto de métricas de avaliação (Equações (4.9a), (4.9b), (4.9d) e (4.9f)).

A Tabela 4.7, inspirada na **Tabela 2** de [PGH+13], apresenta os resultados de avaliação do segundo conjunto de métricas de avaliação (Subsecção 4.4.3) obtidos com a experiência, descrita anteriormente, do *método proposto* e o *corpus PAN13-TextAlignment* (Subsecção 4.4.1). Na segunda linha dessa tabela, apresentam-se os resultados de validação do método proposto

[60] *Word n-gram*.

Tabela 4.7: Resultados de avaliação da análise detalhada para pan2013, inspirada na **Tabela 2** de [PGH+13].

Métodos	plagdet	rec	prec	gran	tempo
Método Proposto	0.13246	0.18324	0.58352	3.30331	81.5 m
[RM13]	**0.82220**	0.76190	**0.89484**	1.00141	**1.2 m**
[KQD+13]	0.81896	**0.81344**	0.82859	1.00336	6.1 m
[SKB13]	0.74482	0.76593	0.72514	1.00028	28.0 m
[SS13]	0.69551	0.73814	0.87461	1.22084	684.5 m
[Gil13]	0.40059	0.25890	0.88487	**1.00000**	21.3 m

(*pesquisa e análise de plágio* e a métrica *nBinGram*), e para efeitos de comparação também são apresentados os resultados de avaliação de *outros métodos de análise detalhada* [RM13, KQD+13, SKB13, SS13, Gil13], originalmente apresentados e avaliados na *pan2013* [PGH+13]. Em termos gerais, na segunda coluna são apresentados os resultados de *plagdet* (métrica de avaliação que engloba todas as outras), o resultado do método proposto mostrou-se pouco significativo perante os demais métodos avaliados em [PGH+13], estando na sexta posição dessa tabela. Salienta-se que os resultados de validação[61] do algoritmo de *pesquisa e análise de plágio* com a métrica *nBinGram* e o *corpus PAN13-SourceRetrieval* distanciam-se dos resultados de avaliação[62] da métrica *nBinGram* com o *corpus PAN-PC-11*. Esse facto deveu-se a um conjunto de fatores, dos quais destaca-se por um lado a incapacidade de ter acesso à totalidade dos dados originais da PAN13 e consequentemente impossibilidade de replicar exatamente as mesmas condições. Por outro lado, suspeita-se que os parâmetros ψ e λ denotem um elevado grau de sensibilidade com implicação direta nos resultados. Todavia, não foi possível explorar uma otimização destes parâmetros, devendo proceder-se a esta análise no futuro.
Deste modo, com as experiências apresentadas conclui-se a avaliação do método de análise detalhada proposto. Na avaliação com o *corpus PAN-PC-11* os resultados obtidos com as métricas criadas no método proposto mostraram-se promissores, com valores de *F-Measure* superiores a 0.88. Entretanto, na validação com o *corpus PAN13-SourceRetrieval* os resultados obtidos com o algoritmo de pesquisa e análise de plágio e uma das métricas criadas no método proposto mostraram-se desfavoráveis e suscetíveis a pequenos reajustes nos valores de ψ e λ, e pequenas melhorias no algoritmo em si.

4.5 Sumário

Este capítulo abordou o método proposto de análise detalhada para a deteção de plágio. Inicialmente, foram introduzidos os principais conceitos associados à análise detalhada, a seguir foi definida problemática abordada e apresentada a proposta de solução para a mesma. Seguidamente, foi apresentado o método proposto para solucionar o problema: criação de métricas de deteção de plágio; e pesquisa e análise de plágio. Posteriormente, foram apresentadas as métricas heurísticas criadas para detetar os indícios de plágio nos excertos. Assim como, foram apresentadas as métricas inteligentes criadas por intermédio da extração de características de

[61] **Um** único teste de avaliação.
[62] **Vários** testes de treino e **um** único teste de avaliação com diferentes dados.

plágio para a posterior indução de padrões de classificação de plágio. A seguir, foi apresentada a pesquisa e análise de plágio entre o documento suspeito e os documentos candidatos. Finalmente, foram apresentadas as experiências realizadas e os resultados obtidos com o método proposto. Assim foi dando ênfase ao conjunto de dados utilizado, as ferramentas adotadas e os parâmetros de configuração utilizados na implementação do método, assim como foram apresentados os métodos de avaliação adotados e os resultados experimentais obtidos do método de análise detalhada proposto.

Capítulo 5

Conclusões

Nesta dissertação apresentou-se uma metodologia para deteção de plágio em grandes corpora fonte. A metodologia proposta divide-se em dois atos, recuperação fonte e análise detalhada (Capítulos 3 e 4) [FC15].

Na recuperação fonte (primeiro ato) propôs-se um método para indexação da fonte (Secção 3.1), formulação da chave de pesquisa (Secção 3.2), e pesquisa e filtragem da fonte (Secção 3.3) [FC15]. Na indexação da fonte propôs-se a utilização de um motor de pesquisa (código aberto) para obtenção um pequeno conjunto de documentos fonte relacionados com um documento suspeito (Algoritmo 3.1). Na formulação da chave de pesquisa propôs-se a utilização de uma métrica criada em trabalhos anteriores [Men13] (Equação (3.2)) juntamente com um novo algoritmo, para a extração de palavras relevantes do documento suspeito (Algoritmo 3.2). Na pesquisa da fonte propôs-se uma nova métrica (Equação (3.3)) para o ajuste automático da extensão das chaves (submetidas em pesquisa) em função do tamanho do documento suspeito. Na filtragem da fonte propôs-se um novo algoritmo para a seleção dos documentos fonte (candidatos a terem sido alvo de plágio) devolvidos pelo motor de pesquisa (Algoritmo 3.3).

Na análise detalhada (segundo ato) propôs-se um método para pesquisa, análise e deteção de indícios de plágio entre documentos suspeitos e documentos fonte (Secções 4.1, 4.2 e 4.3) [FC15]. Na deteção de plágio propuseram-se quatro novas métricas para a deteção de plágio. Duas dessas métricas (Equações (4.4a) e (4.5a)) foram criadas através da combinação heurística das principais métricas de similaridade textuais (Equações (4.1), (4.2) e (4.3)) encontradas na literatura. As outras duas métricas (Equações (4.7) e (4.8)) foram induzidas (automaticamente) com a programação genética, a partir de regularidades típicas (Figuras 4.3 e 4.4), implícitas no plágio e na ofuscação textual, encontradas num conjunto de características textuais extraídas (Tabela 4.1) de dados com plágio. Na pesquisa e análise de plágio propôs-se um novo algoritmo de pesquisa e análise de plágio (Algoritmo 4.1), para ser utilizado juntamente com uma das novas métricas de deteção de plágio criadas (Equação (4.4a)).

Consequentemente, no primeiro ato da deteção de plágio, utilizando corpora de plágio (*corpus PAN-PC-11* e *corpus PAN13-SourceRetrieval*, Subsecção 3.4.1), mostrou-se que os atuais motores de pesquisa para além de serem eficientes na devolução de páginas *Web* pesquisadas com palavras-chave, também o são na devolução de documentos fonte (candidatos a terem sido alvo de plágio) pesquisados com palavras informativas (chave de pesquisa) extraídas de documentos suspeitos (Tabela 3.2). Destaca-se a obtenção de resultados muito animadores (Tabelas 3.3 e 3.4). Consequentemente, no segundo ato da deteção de plágio, utilizando corpora de plágio (*corpus PAN-PC-11* e *corpus PAN13-TextAlignment*, Subsecção 4.4.1), mostrou-se que as cópias ilícitas de texto possuem semelhanças léxico-gramaticais que ultrapassam os limites da similaridade textual (Tabelas 4.2 e 4.3); e também, mostrou-se que essas semelhanças possuem características textuais que podem ser extraídas, assim como possuem regularidades típicas (implícitas no plágio e na ofuscação textual) que podem ser identificadas e utilizadas para criar padrões específicos de classificação de plágio (Tabelas 4.4 e 4.5). Salienta-se a obtenção de resultados muito animadores (Tabelas 4.6 e 4.7).

5.1 Trabalho Futuro

Num trabalho futuro pretende-se aprimorar a metodologia proposta através da afinação dos parâmetros de configuração α, β e ω do método de recuperação fonte proposto, e dos parâmetros de configuração ψ e λ do método de análise detalhada proposto.
Pretende-se também reunir a implementação dos novos algoritmos e métricas (propostos neste trabalho) numa "ferramenta" com o código aberto, disponibilizando aos utilizadores a opção de escolha de diferentes métricas (existentes na literatura e propostas neste trabalho) e diferentes graus de sensibilidade (α, β, ω, ψ e λ) para de deteção de plágio entre documentos de texto na linguagem inglesa e portuguesa. Para tal, poderão ser explorados algoritmos de *aprendizagem máquina*, usando os diferentes corpora da coleção PAN para treino, teste e validação.
Na recuperação fonte pretende-se experimentar o método proposto com diferentes motores de pesquisa com código aberto, com o objetivo de identificar as principais vantagens associadas a cada um, e emula-las num novo motor de pesquisa exclusivamente dedicado às temáticas do plágio. Na análise detalhada pretende-se criar, testar e validar novos algoritmos de pesquisa e análise de plágio, para seleção automática, inteligente e eficiente dos excertos propícios de terem plágio (documento suspeito) e serem plagiados (documento fonte) em pares de documentos com elevadas dimensões.
Considera-se que uma das contribuições, menos convencionais (talvez mesmo exótica) mas mais promissora, deste trabalho foi o uso da *programação genética* que permitiu a indução automática de conhecimento simbólico, nomeadamente padrões matemáticos (condições e funções) caracterizadores do problema em estudo — a deteção automática de plágio. Por um lado, será importante explorar novas fronteiras desta metodologia, possivelmente com melhor desempenho para o problema tratado. Por outro lado, a breve incursão feita no domínio da *programação genética* aplicada ao processamento de linguagem natural, veio revelar um vasto conjunto de outras possibilidades, para outros problemas desta área científica. Ignorar estas possibilidades seria certamente uma enorme lacuna.

Bibliografia

[AAA11] Bernard Ijesunor Akhigbe, Babajide Samuel Afolabi, and Emmanuel Rotimi Adagunodo. Assessment of measures for information retrieval system evaluation: A user-centered approach. *Int. J. Comput. Appl.*, 25(7):6–12, 2011. 21, 47

[Ale11a] Luís Filipe Barbosa de Almeida Alexandre. Inteligência Computacional, aulas teóricas; aula 1: Apresentação, critérios, programa. introdução à inteligência computacional. [online]. 2011. Available from: `https://web.archive.org/web/20110626234334/http://www.di.ubi.pt/~lfbaa/ic/files/aula1.pdf` [cited 2016-08-28]. 38

[Ale11b] Luís Filipe Barbosa de Almeida Alexandre. Inteligência Computacional, aulas teóricas; aula 6: Introdução à computação evolucionária (ce). componentes dum ae. cromossomas. função de aptidão. inicialização da população. operadores de selecção e de reprodução. algoritmo evolucionário genérico. ce versus optimização clássica. [online]. 2011. Available from: `https://web.archive.org/web/20110627000935/http://www.di.ubi.pt/~lfbaa/ic/files/aula6.pdf` [cited 2016-08-28]. 38

[Ale11c] Luís Filipe Barbosa de Almeida Alexandre. Inteligência Computacional, aulas teóricas; aula 7: Algoritmos genéticos. codificação dos cromossomas: valores binários, nominais e reais. operadores de cross-over e mutação. exemplo real. [online]. 2011. Available from: `https://web.archive.org/web/20110626135523/http://www.di.ubi.pt/~lfbaa/ic/files/aula7.pdf` [cited 2016-08-28]. 38

[Ale11d] Luís Filipe Barbosa de Almeida Alexandre. Inteligência Computacional, aulas teóricas; aula 8: Programação genética: representação dos cromossomas; população inicial; função de aptidão; operadores de cross-over e de mutação. uma variante de pg. [online]. 2011. Available from: `https://web.archive.org/web/20110626135529/http://www.di.ubi.pt/~lfbaa/ic/files/aula8.pdf` [cited 2016-08-28]. 39

[ASA12] Salha Alzahrani, Naomie Salim, and Ajith Abraham. Understanding plagiarism linguistic patterns, textual features, and detection methods. *IEEE Trans. Syst., Man, Cybern., Syst., Part C*, 42(2):133–149, 2012. xi, 1, 3, 4, 6, 31, 32, 36, 37, 45, 51

[Ba16] Daniel Smania Brandão. Máximos e mínimos - otimização [online]. 2016. Available from: `https://web.archive.org/web/20160829142347/http://eaulas.usp.br/portal/video.action?idItem=2750` [cited 2016-08-29]. 38

[BB07] Markus F. Brameier and Wolfgang Banzhaf. *Linear Genetic Programming*. Springer US, 2007. 39

[BR09] Alberto Barrón-Cedeño and Paolo Rosso. On automatic plagiarism detection based on n-grams comparison. In Mohand Boughanem, Catherine Berrut, Josiane Moth, and Chantal Soule'-Dupuy, editors, *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, pages 696–700, Berlin, Heidelberg, 2009. Springer-Verlag. 34, 51

[BT08] Granville Barnett and Luca Del Tongo. *Data Structures and Algorithms: Annotated Reference with Examples*. DotNetSlackers, 1st edition, 2008. Available from: https://web.archive.org/web/20160411101811/http://lib.mdp.ac.id/ebook/Karya%20Umum/Dsa.pdf. 19, 45

[CC04] Gerardo Canfora and Luigi Cerulo. A taxonomy of information retrieval models and tools. *J. Comput. Inf. Technol.*, 12(3):175–194, 2004. 9

[CDB07a] João Cordeiro, Gaël Dias, and Pavel Brazdil. Learning paraphrases from wns corpora. In David C. Wilson and Geoffrey C. J. Sutcliffe, editors, *The 20th International Florida Artificial Intelligence Research Society Conference (FLAIRS-07)*, pages 193–198, Key West, Florida, May 2007. The AAAI Press. Available from: https://web.archive.org/web/20160802190406/http://www.aaai.org/Papers/FLAIRS/2007/Flairs07-040.pdf. 32, 36, 45

[CDB07b] João Cordeiro, Gaël Dias, and Pavel Brazdil. A metric for paraphrase detection. In *International Multi-Conference on Computing in the Global Information Technology (ICCGI'07)*., pages 1–6. IEEE Comp. Soc., March 2007. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4137062. 32, 33, 36, 37, 45

[CDB07c] João Cordeiro, Gaël Dias, and Pavel Brazdil. New functions for unsupervised asymmetrical paraphrase detection. *J. Softw.*, 2(4):12–23, 2007. Available from: https://web.archive.org/web/20160802165929/http://www.di.ubi.pt/~jpaulo/competence/publications/jsw02041223.pdf. 32, 36, 45

[Com14] The Apache OpenNLP Development Community. Apache OpenNLP developer documentation [online]. 2014. Version 1.6.0. *The Apache Software Foundation*. Available from: https://web.archive.org/web/20160407111955/https://opennlp.apache.org/documentation/1.6.0/manual/opennlp.html [cited 2016-04-07]. 19, 45

[Cor03] João Paulo da Costa Cordeiro. Extracção de elementos relevantes em texto/páginas da world wide web. In *Tese para obtenção do Grau de Mestre em Inteligência Artificial e Computação*, pages 131–134. Departamento de Ciência de Computadores, Faculdade de Ciências da Universidade do Porto, 2003. 25, 49

[Cor06] João Paulo da Costa Cordeiro. The HultigLib: "nuggets" for text processing in java [online]. 2006. Version 1.1. *Centre For Human Language Technology and Bioinformatics* (HULTIG). Available from: https://web.archive.org/web/20160405113417/http://www.di.ubi.pt/~jpaulo/hultiglib/ [cited 2016-04-05]. 18, 19, 44, 45, 46

[CP90] D. Cutting and J. Pedersen. Optimization for dynamic inverted index maintenance. In *13th International Conference on Research and Development in Information Retrieval*, SIGIR '90, pages 405–411, New York, NY, USA, 1990. ACM. 11

[Dar59] Charles Darwin. *On the Origin of Species by Means of Natural Selection*. John Murray, Albemarle Street, London, 1859. or the Preservation of Favored Races in the Struggle for Life. Available from: https://web.archive.org/web/20160828140105/https://en.wikisource.org/wiki/On_the_Origin_of_Species_(1859). 38

[EBR09] Samhaa R. El-Beltagy and Ahmed Rafea. Kp-miner: A keyphrase extraction system for english and arabic documents. *Infor. Syst.*, 34(1):132–144, 2009. Available from: http://www.sciencedirect.com/science/article/pii/S0306437908000537. 5

[Egg05] Jeroen Eggermont. *Data Mining using Genetic Programming: Classification and Symbolic Regression*. PhD thesis, Institute for Programming research and Algorithmics, Leiden Institute of Advanced Computer Science, Faculty of Mathematics & Natural Sciences, Leiden University, The Netherlands, 2005. 38, 39

[Eli13] Victoria Elizalde. Using statistic and semantic analysis to detect plagiarism—notebook for pan at clef 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: https://web.archive.org/web/20150319025821/http://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-Elizalde2013.pdf. 26

[Eng07] Andries P. Engelbrecht. *Computational Intelligence: An Introduction*. Wiley Publishing, 2nd edition, 2007. 2, 39

[FC15] Bruno Felipe and João Cordeiro. Deteção automática de plágio em dois atos. In Luís Veiga and Ricardo Rocha, editors, *INFORUM 2015 - Atas do 7.º Simpósio Nacional de Informática*, pages 311–326. UBI - Universidade da Beira Interior. Serviços Gráficos, 2015. Available from: https://web.archive.org/web/20160715150120/https://www.dcc.fc.up.pt/~ricroc/homepage/publications/2015-INFORUM.pdf. v, vii, xi, 1, 2, 23, 26, 34, 37, 39, 40, 55

[Fer14] Bruno Miguel Fernandes. Sumarização personalizada e subjectiva de texto. In *Dissertação para obtenção do Grau de Mestre*, pages 33–34. Departamento de Informática, Universidade da Beira Interior, Covilhã, 2014. 12

[FS96] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. In Lorenza Saitta, editor, *Proceedings of the Thirteenth International Conference on Machine Learning*, pages 148–156. Morgan Kaufmann, 1996. Available from: http://www.biostat.wisc.edu/~kbroman/teaching/statgen/2004/refs/freund.pdf. 46

[GCF+15] João Gama, André Ponce de Leon Carvalho, Katti Faceli, Ana Carolina Lorena, and Márcia Oliveira. *Extração de Conhecimento de Dados - Data Mining*. Edições Sílabo, 2nd edition, 2015. Available from: https://web.archive.org/web/20160518142206/http://www.silabo.pt/livros.asp?tit=Extra%E7%E3o+de+Conhecimento+de+Dados&aut=&is=&ano=&Submit=Procurar&num=473. 20, 30, 47

[Gil13] Lee Gillam. Guess again and see if they line up: Surreyś runs at plagiarism detection—notebook for pan at clef 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: https://web.archive.org/web/20150319003103/http://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-Gillam2013.pdf. 26, 52

[GJS+15] James Gosling, Bill Joy, Guy Steele, Gilad Bracha, and Alex Buckley. *The Java® Language Specification, Java SE 8 Edition*. Oracle America, Inc. and/or its affiliates. 500 Oracle Parkway, Redwood City, California 94065, U.S.A., 8th edition, 2015. Available from: https://docs.oracle.com/javase/specs/jls/se8/jls8.pdf. 18, 44

[Hea97] Marti A. Hearst. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Comput. Linguist.*, 23(1):33–64, 1997. Available from: https://web.archive.org/web/20160114075043/http://www.aclweb.org/anthology/J97-1003. 5

[HEB13] Osama Haggag and Smhaa El-Beltagy. Plagiarism Candidate Retrieval Using Selective Query Formulation and Discriminative Query Scoring—Notebook for PAN at CLEF 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *CLEF 2013 Evaluation Labs and Workshop - Working Notes Papers, 23-26 September, Valencia, Spain*. CLEF and CEUR-WS.org, September 2013. Available from: https://web.archive.org/web/20160912205301/http://ceur-ws.org/Vol-1179/. 5, 26

[HPS15] Matthias Hagen, Martin Potthast, and Benno Stein. Source retrieval for plagiarism detection from large web corpora: Recent approaches. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, 2015. Available from: http://ceur-ws.org/Vol-1391/. xi, 3, 4, 5, 6, 8, 17, 18, 22

[Hua08] Anna Huang. Similarity measures for text document clustering. In *Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC 2008), Christchurch, New Zealand*, pages 49–56, 2008. Available from: https://web.archive.org/web/20150624133639/http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.332.4480&rep=rep1&type=pdf. 30, 31, 32, 36, 45

[Ini07] Open Source Initiative. The open source definition [online]. 2007. Licenses & Standards. Available from: https://web.archive.org/web/20160406100131/https://opensource.org/definition [cited 2016-04-06]. 18, 44

[Jac01a] Paul Jaccard. Distribution de la flore alpine dans le bassin des dranses et dans quelques régions voisines. *B. Soc. Vaud. Sci. N.*, 37:241–272, 1901. 32, 37

[Jac01b] Paul Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *B. Soc. Vaud. Sci. N.*, 37:547–579, 1901. 32, 37

[Jay12] Arun kumar Jayapal. Similarity Overlap Metric and Greedy String Tiling at PAN 2012: Plagiarism Detection—Notebook for PAN at CLEF 2012. In Pamela Forner, Jussi Karlgren, and Christa Womser-Hacker, editors, *CLEF 2012 Evaluation Labs and Workshop - Working Notes Papers, 17-20 September, Rome, Italy*. CLEF and CEUR-WS.org, September 2012. Available from: https://web.archive.org/web/20160912205220/http://ceur-ws.org/Vol-1178/. 4, 5

[KBLP55] Allen Kent, Madeline M. Berry, Fred U. Luehrs, Jr., and J. W. Perry. Machine literature searching VIII. Operational criteria for designing information retrieval systems. *Am. Doc.*, 6(2):93–101, 1955. 16, 20, 21, 22, 47, 48

[KLH+15] Leilei Kong, Zhimao Lu, Yong Han, Haoliang Qi, Zhongyuan Han, Qibo Wang, Zhenyuan Hao, and Jing Zhang. Source Retrieval and Text Alignment Corpus Construction for Plagiarism Detection—Notebook for PAN at CLEF 2015. In Linda Cappellato, Nicola Ferro, Gareth Jones, and Eric San Juan, editors, *CLEF 2015 Evaluation Labs and Workshop - Working Notes Papers, 8-11 September, Toulouse, France*. CLEF and CEUR-WS.org, September 2015. Available from: https://web.archive.org/web/20160912205452/http://ceur-ws.org/Vol-1391/. 5, 6

[Koz89] John R. Koza. Hierarchical genetic algorithms operating on populations of computer programs. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, volume 1 of *IJCAI'89*, pages 768–774, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc. 39

[Koz90] John R. Koza. Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems, 1990. 39

[KQD+13] Leilei Kong, Haoliang Qi, Cuixia Du, Mingxing Wang, and Zhongyuan Han. Approaches for Source Retrieval and Text Alignment of Plagiarism Detection—Notebook for PAN at CLEF 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *CLEF 2013 Evaluation Labs and Workshop - Working Notes Papers, 23-26 September, Valencia, Spain*. CLEF and CEUR-WS.org, September 2013. Available from: https://web.archive.org/web/20160912205301/http://ceur-ws.org/Vol-1179/. 7, 26, 52

[KQW+12] Leilei Kong, Haoliang Qi, Shuai Wang, Cuixia Du, Suhong Wang, and Yong Han. Approaches for Candidate Document Retrieval and Detailed Comparison of Plagiarism Detection—Notebook for PAN at CLEF 2012. In Pamela Forner, Jussi Karlgren, and Christa Womser-Hacker, editors, *CLEF 2012 Evaluation Labs and Workshop - Working Notes Papers, 17-20 September, Rome, Italy*. CLEF and CEUR-WS.org, September 2012. Available from: https://web.archive.org/web/20160912205220/http://ceur-ws.org/Vol-1178/. 6

[LCPJ13] Taemin Lee, Jeongmin Chae, Kinam Park, and Soonyoung Jung. Copycaptor: Plagiarized source retrieval system using global word frequency and local feedback—notebook for pan at clef 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: https://web.archive.org/web/20150319022626/http://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-LeeEt2013.pdf. 26

[Lev66] Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics-Doklady*, 10(8):707–710, 1966. Available from: https://web.archive.org/web/20160802165607/https://nymity.ch/sybilhunting/pdf/Levenshtein1966a.pdf. 32, 36, 45

[Lev08] Yan Levasseur. Techniques de l'intelligence artificielle pour la classification d'objets biologiques dans des images bidimensionnelles. In *Mémoire de Maîtrise Électronique, École de Technologie Supérieure, Montréal*, pages 50–90, 2008. 46

[Lev10] Yan Levasseur. Le Yan genetic programming [online]. 2010. Available from: https://web.archive.org/web/20150128030042/http://www.leyan.org/tiki-index.php?page=Genetic%20Programming [cited 2016-09-04]. 46

[LG05] Tao Liu and Jun Guo. Text similarity computing based on standard deviation. In DeShuang Huang, Xiao-Ping Zhang, and Guang-Bin Huang, editors, *Advances in Intelligent Computing: International Conference on Intelligent Computing, Proceedings, Part I*, ICIC 2005, pages 456–464, Berlin, Heidelberg, 2005. Springer-Verlag Berlin Heidelberg. 30

[LMD01] Caroline Lyon, James Malcolm, and Bob Dickerson. Detecting short passages of similar text in large document collections. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, 2001. Available from: https://web.archive.org/web/20160803090920/http://www.aclweb.org/anthology/W01-0515. 32, 37, 45

[Men13] Henrique da Costa Mendes. Similaridade documental e detecção de plágio. In *Dissertação para obtenção do Grau de Mestre*, pages 25–28. Departamento de Informática, Universidade da Beira Interior, Covilhã, 2013. 12, 55

[MHG10] Michael McCandless, Erik Hatcher, and Otis Gospodnetic. *Lucene in Action, Second Edition: Covers Apache Lucene 3.0*. Manning Publications Co., Greenwich, CT, USA, 2010. 11, 14, 19

[Mil95] George A. Miller. Wordnet: A lexical database for english. *Commun. of The ACM*, 38(11):39–41, Nov 1995. Available from: https://web.archive.org/web/20160809092505/http://nlp.cs.swarthmore.edu/~richardw/papers/miller1995-wordnet.pdf. 34, 35, 37, 45

[Mit14] Mitzimorris. Lingpipe blog, natural language processing and text analytics - Lucene 4 essentials for text search and indexing [online]. 2014. Available from: http://lingpipe-blog.com/2014/03/08/lucene-4-essentials-for-text-search-and-indexing/ [cited 2015-08-25]. 11

[PB14] Yurii Palkovskii and Alexei Belov. Developing High-Resolution Universal Multi-Type N-Gram Plagiarism Detector—Notebook for PAN at CLEF 2014. In Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, editors, *CLEF 2014 Evaluation Labs and Workshop - Working Notes Papers, 15-18 September, Sheffield, UK*. CLEF and CEUR-WS.org, September 2014. Available from: https://web.archive.org/web/20160912205400/http://ceur-ws.org/Vol-1181/. 7

[PBE+10] Martin Potthast, Alberto Barrón-Cedeño, Andreas Eiselt, Benno Stein, and Paolo Rosso. Overview of the 2nd international competition on plagiarism detection. In Martin Braschler, Donna Harman, and Emanuele Pianta, editors, *Working Notes Papers of the CLEF 2010 Evaluation Labs*, 2010. Available from: http://ceur-ws.org/Vol-1176/. 3, 4, 6

[PEB+11] Martin Potthast, Andreas Eiselt, Alberto Barrón-Cedeño, Benno Stein, and Paolo Rosso. Overview of the 3rd international competition on plagiarism detection. In Vivien Petras, Pamela Forner, and Paul D. Clough, editors, *Working Notes Papers of the CLEF 2011 Evaluation Labs*, 2011. Available from: http://ceur-ws.org/Vol-1177/. 3, 4, 6, 16, 17, 36, 43, 44

[PGH+12] Martin Potthast, Tim Gollub, Matthias Hagen, Jan Graßegger, Johannes Kiesel, Maximilian Michel, Arnd Oberländer, Martin Tippmann, Alberto Barrón-Cedeño, Parth Gupta, Paolo Rosso, and Benno Stein. Overview of the 4th international competition on plagiarism detection. In Pamela Forner, Jussi Karlgren, and Christa Womser-Hacker, editors, *Working Notes Papers of the CLEF 2012 Evaluation Labs*, 2012. Available from: http://ceur-ws.org/Vol-1178/. xi, 3, 4, 5, 6, 7, 8, 17, 18

[PGH+13] Martin Potthast, Tim Gollub, Matthias Hagen, Martin Tippmann, Johannes Kiesel, Paolo Rosso, Efstathios Stamatatos, and Benno Stein. Overview of the 5th international competition on plagiarism detection. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: https://web.archive.org/web/20150319024053/http:

//ceur-ws.org/Vol-1179/CLEF2013wn-PAN-PotthastEt2013.pdf. xi, xiii, 3, 4, 5, 6, 7, 8, 16, 17, 18, 20, 21, 22, 23, 25, 26, 43, 44, 46, 47, 48, 51, 52

[PHB+14] Martin Potthast, Matthias Hagen, Anna Beyer, Matthias Busse, Martin Tippmann, Paolo Rosso, and Benno Stein. Overview of the 6th international competition on plagiarism detection. In Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, editors, *Working Notes Papers of the CLEF 2014 Evaluation Labs*, CEUR Workshop Proceedings. CLEF and CEUR-WS.org, 2014. Available from: http://ceur-ws.org/Vol-1180/. xi, 3, 4, 5, 6, 7, 8, 17, 18, 22

[PHVS13] Martin Potthast, Matthias Hagen, Michael Völske, and Benno Stein. Crowdsourcing interaction logs to understand text reuse from the web. In Pascale Fung and Massimo Poesio, editors, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1212–1221. Assoc. Comput. Linguist., 2013. Available from: http://www.aclweb.org/anthology/P13-1119. 16, 17, 18, 20, 27

[Por80] Martin F. Porter. An algorithm for suffix stripping. *Program-Electron. Lib.*, 14(3):130–137, 1980. Available from: https://web.archive.org/web/20160808180619/https://tartarus.org/martin/PorterStemmer/def.txt. 33, 34, 35, 37

[Por06] Martin Porter. The porter stemming algorithm [online]. 2006. Available from: https://web.archive.org/web/20160808181101/https://tartarus.org/martin/PorterStemmer/ [cited Accessed: 2016-08-08]. 33, 35, 37

[Pri16] Priberam Informática S.A. Dicionário Priberam da língua portuguesa [online]. 2008-2016. Available from: http://www.priberam.pt/dlpo/ [cited 2016-08-06]. 29, 30, 37

[PSBR10] Martin Potthast, Benno Stein, Alberto Barrón-Cedeño, and Paolo Rosso. An evaluation framework for plagiarism detection. In Chu-Ren Huang and Dan Jurafsky, editors, *23rd International Conference on Computational Linguistics (COLING 10)*, pages 997–1005, Stroudsburg, Pennsylvania, 2010. Assoc. Comput. Linguist. 16, 17, 29, 32, 33, 45, 47

[PSE+09] Martin Potthast, Benno Stein, Andreas Eiselt, Alberto Barrón-Cedeño, and Paolo Rosso. Overview of the 1st international competition on plagiarism detection. In Benno Stein, Paolo Rosso, Efstathios Stamatatos, Moshe Koppel, and Eneko Agirre, editors, *SEPLN 09 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse*, pages 1–9. CLEF and CEUR-WS.org, 2009. Available from: http://ceur-ws.org/Vol-502. xi, 3, 4, 6

[Qui93] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. 49

[RG15] Riya Ravi N and Deepa Gupta. Efficient Paragraph based Chunking and Download Filtering for Plagiarism Source Retrieval—Notebook for PAN at CLEF 2015. In Linda Cappellato, Nicola Ferro, Gareth Jones, and Eric San Juan, editors, *CLEF 2015 Evaluation Labs and Workshop - Working Notes Papers, 8-11 September, Toulouse, France*. CLEF and CEUR-WS.org, September 2015. Available from: https://web.archive.org/web/20160912205452/http://ceur-ws.org/Vol-1391/. 5

[Rij79] C. J. Van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979. 16, 20, 21, 22, 43, 47, 48

[RLS+04] Flávio Lopes Rodrigues, Helio Garcia Leite, Heleno do Nascimentos Santos, Agostinho Lopes de Souza, and Gilson Fernandes da Silva. Metaheurística algoritmo genético para solução de problemas de planejamento florestal com restrições de integridade. *R. Árvore*, 28(2):233–245, 2004. Available from: http://www.scielo.br/pdf/rarv/v28n2/20988.pdf [cited 2016-08-29]. 38

[RM13] Diego Antonio Rodríguez Torrejón and José Manuel Martín Ramos. Text Alignment Module in CoReMo 2.1 Plagiarism Detector—Notebook for PAN at CLEF 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *CLEF 2013 Evaluation Labs and Workshop - Working Notes Papers, 23-26 September, Valencia, Spain*. CLEF and CEUR-WS.org, September 2013. Available from: https://web.archive.org/web/20160912205301/http://ceur-ws.org/Vol-1179/. 7, 52

[SB88] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Infor. Proces. Manage.*, 24(5):513–523, 1988. 12

[SGM00] Alexander Strehl, Joydeep Ghosh, and Raymond Mooney. Impact of similarity measures on web-page clustering. In *In Workshop on Artificial Intelligence for Web Search (AAAI 2000*, pages 58–64. The AAAI Press., 2000. Available from: https://web.archive.org/web/20151203204915/http://www.aaai.org/Papers/Workshops/2000/WS-00-01/WS00-01-011.pdf. 30

[Sha48] C. E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(3):379–423, July 1948. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6773024. 32, 36, 45

[Shi14] Daniel Shiffman. WordNet examples and exercises [online]. 2014. Available from: https://web.archive.org/web/20140712174037/http://shiffman.net/teaching/a2z/wordnet/ [cited 2016-08-08]. 34, 35, 37, 45

[Sin01] Amit Singhal. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 24(4):35–43, 2001. 9

[SKB12] Šimon Suchomel, Jan Kasprzak, and Michal Brandejs. Three Way Search Engine Queries with Multi-feature Document Comparison for Plagiarism Detection—Notebook for PAN at CLEF 2012. In Pamela Forner, Jussi Karlgren, and Christa Womser-Hacker, editors, *CLEF 2012 Evaluation Labs and Workshop - Working Notes Papers, 17-20 September, Rome, Italy*. CLEF and CEUR-WS.org, September 2012. Available from: https://web.archive.org/web/20160912205220/http://ceur-ws.org/Vol-1178/. 4, 7

[SKB13] Šimon Suchomel, Jan Kasprzak, and Michal Brandejs. Diverse queries and feature type selection for plagiarism discovery—notebook for pan at clef 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: https://web.archive.org/web/20150319021914/http://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-SuchomelEt2013.pdf. 26, 52

[SMP07] Benno Stein, Sven Meyer zu Eißen, and Martin Potthast. Strategies for retrieving plagiarized documents. In Charles Clarke, Norbert Fuhr, Noriko Kando, Wessel Kraaij, and Arjen P. de Vries, editors, *30th International ACM Conference on Research and Development in Information Retrieval (SIGIR 07)*, pages 825–826, New York, 2007. ACM. xi, 3, 4, 6

[Spe04] Lee Spector. A genetic programming approach. In *Automatic Quantum Computer Programming*, volume 7 of *Genetic Programming*, pages 43–54. Springer US, 2004. 39

[SPR+15] Efstathios Stamatatos, Martin Potthast, Francisco Rangel, Paolo Rosso, and Benno Stein. Overview of the pan/clef 2015 evaluation lab. In Josiane Mothe, Jacques Savoy, Jaap Kamps, Karen Pinel-Sauvagnat, Gareth J.F. Jones, Eric SanJuan, Linda Cappellato, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 6th International Conference of the CLEF Initiative (CLEF 15)*, pages 518–538, Berlin Heidelberg New York, 2015. Springer. 4, 6

[SPSG14] Miguel A. Sanchez-Perez, Grigori Sidorov, and Alexander Gelbukh. A Winning Approach to Text Alignment for Text Reuse Detection at PAN 2014—Notebook for PAN at CLEF 2014. In Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, editors, *CLEF 2014 Evaluation Labs and Workshop - Working Notes Papers, 15-18 September, Sheffield, UK*. CLEF and CEUR-WS.org, September 2014. Available from: `https://web.archive.org/web/`20160912205400`/http://ceur-ws.org/Vol-`1181`/`. 7

[SS13] Prasha Shrestha and Thamar Solorio. Using a variety of n-grams for the detection of different kinds of plagiarism—notebook for pan at clef 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: `https://web.archive.org/web/`20150319014529`/http://ceur-ws.org/Vol-`1179`/CLEF`2013`wn-PAN-ShresthaEt`2013`.pdf`. 52

[Sta11] Efstathios Stamatatos. Plagiarism detection using stopword n-grams. *J. Am. Soc. Inf. Sci. Technol.*, 62(12):2512–2527, December 2011. Available from: `https://web.archive.org/web/`20160914180822`/http://www.icsd.aegean.gr/lecturers/stamatatos/papers/JASIST`11`-preprint.pdf`. 7

[SYY75] G. Salton, C. S. Yang, and C. T. Yu. A theory of term importance in automatic text analysis. *J. Am. Soc. Inf. Sci.*, 26(1):33–44, 1975. 12

[Tan58] T. Tanimoto. An elementary mathematical theory of classification and prediction. IBM Internal Report, 1958. 32

[Tea07] The JWNL Development Team. JWNL java wordnet library [online]. 2000–2007. Available from: `https://web.archive.org/web/`20160809091557`/https://sourceforge.net/projects/jwordnet/` [cited 2016-08-09]. 34, 35, 37, 45

[Uni15] University of Glasgow - School of Computing Science. Terrier IR Platform, stopwords list with 733 words [online]. 2015. Available from: `https://web.archive.org/web/`20160808163544`/https://github.com/RxNLP/text-mining-and-nlp-apis/blob/master/terrier-stop-word-list.txt` [cited 2016-08-08]. 12, 34, 35, 37, 45

[Val11] Kjetil Valle. Graph-based representations for text classification [online]. 2011. Available from: https://web.archive.org/web/20160803123504/http://kjetilvalle.com/posts/graph-based-representations-for-text-classification.html [cited 2016-08-23]. 32

[Ven14] João Miguel Jones Ventura. Automatic extraction of concepts from texts and applications. In *Dissertação para obtenção do Grau de Doutor em Informática*, pages 18–. Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, 2014. 12

[VFR13] Ondřej Veselý, Tomáš Foltýnek, and Jiří Rybička. Source retrieval via naïve approach and passage selection heuristics—notebook for pan at clef 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013. Available from: https://web.archive.org/web/20150319012028/http://ceur-ws.org/Vol-1179/CLEF2013wn-PAN-VeselyEt2013.pdf. 26

[WCCG13] Kyle Williams, Hung-Hsuan Chen, Sagnik Ray Chowdhury, and C. Lee Giles. Unsupervised Ranking for Plagiarism Source Retrieval—Notebook for PAN at CLEF 2013. In Pamela Forner, Roberto Navigli, and Dan Tufis, editors, *CLEF 2013 Evaluation Labs and Workshop - Working Notes Papers, 23-26 September, Valencia, Spain*. CLEF and CEUR-WS.org, September 2013. Available from: https://web.archive.org/web/20160912205301/http://ceur-ws.org/Vol-1179/. 5

[WCG14] Kyle Williams, Hung-Hsuan Chen, and C. Lee Giles. Supervised Ranking for Plagiarism Source Retrieval—Notebook for PAN at CLEF 2014. In Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, editors, *CLEF 2014 Evaluation Labs and Workshop - Working Notes Papers, 15-18 September, Sheffield, UK*. CLEF and CEUR-WS.org, September 2014. Available from: https://web.archive.org/web/20160912205400/http://ceur-ws.org/Vol-1181/. 5

[WF05] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005. xiii, 21, 22, 25, 30, 46, 47, 48, 49

[Wik14] Fundación Wikimedia. Wikipedia, la enciclopedia de contenido libre: inteligencia computacional [online]. 2014. Available from: https://web.archive.org/web/20160828111156/https://es.wikipedia.org/wiki/Inteligencia_computacional [cited 2016-08-28]. 38

[Wik16] Foundation Wikimedia. Wikipedia, the free encyclopedia: computational intelligence [online]. 2016. Available from: https://web.archive.org/web/20160828111115/https://en.wikipedia.org/wiki/Computational_intelligence [cited 2016-08-28]. 38

[ZS14] Denis Zubarev and Ilya Sochenkov. Using Sentence Similarity Measure for Plagiarism Source Retrieval—Notebook for PAN at CLEF 2014. In Linda Cappellato, Nicola Ferro, Martin Halvey, and Wessel Kraaij, editors, *CLEF 2014 Evaluation Labs and Workshop - Working Notes Papers, 15-18 September, Sheffield, UK*. CLEF and CEUR-WS.org, September 2014. Available from: https://web.archive.org/web/20160912205400/http://ceur-ws.org/Vol-1181/. 5