TITULO DO TRABALHO

Hugo Veríssimo - 124348 - hugoverissimo@ua.pt

***Abstract* – abstrato em ingles**

***Resumo* – abstrato em pt resumo**

I. Introdução

A análise de texto é uma área de estudo fundamental, com diversas aplicações tais como análise de sentimentos ou de opiniões, personalização da experiência do utilizador, recomendação de conteúdo, entre outras [1]. Uma das tarefas centrais nesta área é a identificação da frequência de palavras em grandes volumes de texto, tal como livros, bases de dados ou redes sociais, de modo extrair informações relevantes sobre o conteúdo e estrutura dos textos em análise.

Contudo, a identificação precisa da frequência de palavras em textos de larga escala apresenta desafios significativos, especialmente em termos de memória. Métodos de contagem precisa, que mantêm o registo exato da contagem de cada palavra, revelam-se ineficientes devido ao elevado consumo de memória. Há assim a necessidade do estudo de métodos mais eficientes e escaláveis, principalmente em situações em que os dados estão em constante fluxo, como em *streams* de dados. Neste contexto, algoritmos de contagem aproximada e identificação de itens frequentes têm vindo a ganhar destaque, uma vez que permitem a identificação de palavras mais frequentes de forma eficiente e com uma margem de erro controlada [2].

Este relatório visa explorar três abordagens para este problema: contadores exatos, contadores aproximados e identificação de itens frequentes em *streams* de dados. Para cada uma destas abordagens, será apresentado um algoritmo e .....

II. Metodologia da Análise

Para realizar a análise de frequência de palavras, foram selecionados três livros, a partir livraria online *Project Gutenberg*, nomeadamente: *Pinocchio: The Tale of a Puppet* (inglês), *Le avventure di Pinocchio: Storia di un burattino* (italiano) e *Pinocchion seikkailut: Kertomus marioneteista* (finlandês). Estes livros foram selecionados por serem traduções do mesmo livro original, conhecido em português como *As Aventuras de Pinóquio*, de Carlo Collodi. A escolha destes livros permite a comparação da frequência de palavras em diferentes idiomas, bem como a análise de semelhanças e diferenças entre as traduções.

Numa primeira fase, os ficheiros de texto descarregados a partir do *Project Gutenberg* foram processados removendo informações irrelevantes, como metadados e licenças, palavras insignificantes e sinais de pontuação. Para além disso todas as palavras foram convertidas para minúsculas e lematizadas. Estas transformações são fundamentais, de modo a simplificar o texto e concentrar a análise nas palavras mais relevantes, garantindo uma avaliação mais precisa e eficiente da frequência de termos. É importante referir que estas transformações foram realizadas com recurso à biblioteca *spaCy*, através do *Python*.

implementar algortimos, analise dos dados, correr ns quantas vezes, ...

III. contagem 1

Contador Exato

O primeiro algoritmo é a contagem toda

Algoritmo 1 Contador Exato

Entrada: texto processado (`T`)
Saída: dicionário onde as palavras são as chaves e os valores são as suas frequências (`D`)

```
1: D ← empty dictionary
2: words ← list of words from T
3: for each word in words do
4:     if word ∉ D then
5:         D[word] ← 0
6:     end if
7:     D[word] ← D[word] + 1
8: end for
9: return D
```

Quanto ao número de soluções testadas, a partir da Fig.

IV. contagem 2

Contadores Aproximados

lalalla

DIZER QUE É 1/16 ANTES DO PSEUDOCODIGO

Algoritmo 2 Contador Aproximado

Entrada: texto processado (`T`)
Saída: dicionário onde as palavras são as chaves e os valores são as suas frequências estimadas (`D`)

```
D ← empty dictionary
words ← list of words from T
for each word in words do
    r ← Uniform(0, 1)
    if r < 1/16 then
        if word ∉ D then
            D[word] ← 0
        end if
        D[word] ← D[word] + 1
    end if
end for
for each word in D do  ▷ Estimate the total count
    D[word] ← D[word] ×16
end for
return D
```

V. CONTAGEM 3

lalallala
http://dimacs.rutgers.edu/ graham/pubs/papers/freqcacm.pdf
fonte do PSEUDOCODIGO

Algoritmo 3 Contador *Space-Saving*

Entrada:
- texto processado (`T`)
- número máximo de itens a manter (`k`)

Saída: dicionário com a estimativa das `k` palavras mais frequentes e respetivas frequências (`D`)

```
D ← empty dictionary
words ← list of words from T
for each word in words do
    if word ∈ D then
        D[word] ← D[word] + 1
    else if |D| < k then
        D[word] ← 1
    else
        j ← arg min_{j∈D} D[j]
        D[word] ← D[j] + 1
        D ← D \{j}
    end if
end for
return D
```

VI. RESULTADOS

TABELA I: CAPTION CAPTION CAPTION

Algoritmo	Complexidade
a	$O(m)$
b	$O(m)$
c	$O(m^2 \times n)$

ffyf

VII. CONCLUSÃO

conclusaoooo

BIBLIOGRAFIA

[1] Amazon Web Services, "What is text analysis?", 2024, https://aws.amazon.com/what-is/text-analysis/. Accessed: 2024-12-11.

[2] Hongyan Liu, Ying Lu, Jiawei Han, e Jun He, "Error-adaptive and time-aware maintenance of frequency counts over data streams", 2006.