Search engine

Pratik Gaikwad

10420318

Contents

[Approach on solving problem: 2](#_Toc500097128)

[To run the project: 2](#_Toc500097129)

[Explanation of the code: 2](#_Toc500097130)

[Program.cs 2](#_Toc500097131)

[TrieProcessor class: 3](#_Toc500097132)

[Itrie.cs 3](#_Toc500097133)

[Trie.cs 3](#_Toc500097134)

[TrieFactory.cs 3](#_Toc500097135)

[TrieNode.cs 3](#_Toc500097136)

[Sample Output: 3](#_Toc500097137)

# Approach on solving problem:

I have used Trie data-structure to save the words since Insert and search costs **O(key\_length)**. I have taken a hard-coded webpage of Schaefer School from Stevens website. After parsing this webpage, I have collected href value of first 5 anchor tags to collect additional pages from Stevens website. This number can be increased up to unlimited. I have inserted these urls in a dictionary with an integer as key.

Once all the webpages are read from the internet, I have then removed all the html tags from the body content of those pages. Then for each page, I have split the content with space (“ “) as delimiter. All the stop words of English language are removed from these contents. The result is then sorted so that collection will contain all the same words next to each other. This way frequency counting will be easier and faster. A dictionary of integer, integer is created to save the document url reference number and frequency of the word in that document.

The each word of the collection is then searched in the Trie structure. If found, only the occurrence of the frequency for the document reference number is updated in the node of Trie. Else a new node is created. During an add operation word index is also generated.

When user searches for the word, the user input is split based on space(“ “) delimitor. Each word is then searched in the Trie structure. And displayed the result based on the descending order of the frequency of word occurrence in the document. However, if intersection of search result is not found, then “no result found” is displayed.

# To run the project:

1. Install .net core environment for your operating system(<https://www.microsoft.com/net/download>).
2. Once installation is completed, open a terminal and Navigate to the project folder
3. Type in below commands:
   1. dotnet restore
   2. dotnet build
   3. dotnet run
4. follow the instructions on screen

# Explanation of the code:

## Program.cs

This is the entry point of the code. When “dotnet run” command is executed, the Main method from class Program in this file is invoked. This file loads page from stevens website. As mentioned above this method then proceeds to obtain data from other webpages.

## TrieProcessor class:

This class is responsible for interacting with trie data-structure. Method ReadWebPage will download the contents from the webpage and return only body contents of the page. Method StripHtmlTags will remove all the html tags to give us pure English content. Methods CreateTrie will generate the trie and start adding the words to it as mentioned above in the logic. Method SearchWord will search for user input word in the trie and returns the result.

## Itrie.cs

This is interface to be implemented by Trie class. This defines the signature of the properties and methods required for trie implementation.

## Trie.cs

This file is where actual trie data-structure is generated and modified. Method AddWord creates the nodes for each character and appends to appropriate node as necessary with index. Method Contains searches for the word in the trie and returns appropriate node or null based on result.

## TrieFactory.cs

This file is where trie structure is initiated. This is also factory class for node of the trie datastructure.

## TrieNode.cs

This file represents the structure of the trie node.

# Sample Output:

PS C:\Users\pratik\Desktop\CS600FinalProject> dotnet build

Microsoft (R) Build Engine version 15.4.8.50001 for .NET Core

Copyright (C) Microsoft Corporation. All rights reserved.

CS600FinalProject -> C:\Users\pratik\Desktop\CS600FinalProject\bin\Debug\netcoreapp2.0\CS600FinalProject.dll

Build succeeded.

0 Warning(s)

0 Error(s)

Time Elapsed 00:00:01.22

PS C:\Users\pratik\Desktop\CS600FinalProject> dotnet run

Started reading webpages

1, https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science

2, https://www.stevens.edu/admissions/undergraduate-admissions/how-apply

3, https://www.stevens.edu/give-now

4, https://www.stevens.edu/learn-about-giving

5, https://www.stevens.edu/future-student

6, https://www.stevens.edu/parents-and-family-members

Finished reading webpages.

Creating trie from the contents of the webpages

Creation of trie completed

Enter input to search or just press 'Enter' key to exit

Hoboken

Word 'Hoboken' is found in the pages as below based on frequency in descending order

Url: https://www.stevens.edu/parents-and-family-members with frequency: 7

Url: https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science with frequency: 2

Url: https://www.stevens.edu/give-now with frequency: 1

The complete search input 'Hoboken' is found in the pages as below

Url: https://www.stevens.edu/parents-and-family-members

Url: https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science

Url: https://www.stevens.edu/give-now

Enter input to search or just press 'Enter' key to exit

sssss

'sssss' not found

Enter input to search or just press 'Enter' key to exit

hoboken new

Word 'hoboken' is found in the pages as below based on frequency in descending order

Url: https://www.stevens.edu/parents-and-family-members with frequency: 7

Url: https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science with frequency: 2

Url: https://www.stevens.edu/give-now with frequency: 1

Word 'new' is found in the pages as below based on frequency in descending order

Url: https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science with frequency: 1

The complete search input 'hoboken new' is found in the pages as below

Url: https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science

Enter input to search or just press 'Enter' key to exit

hoboken ssss

Word 'hoboken' is found in the pages as below based on frequency in descending order

Url: https://www.stevens.edu/parents-and-family-members with frequency: 7

Url: https://www.stevens.edu/news/stevens-names-dr-jean-zu-dean-schaefer-school-engineering-science with frequency: 2

Url: https://www.stevens.edu/give-now with frequency: 1

'hoboken ssss' not found

Enter input to search or just press 'Enter' key to exit

Good bye!!!