

CONCURRENT WEB CRAWLER AND INDEXER

Final Project Report

submitted by

NAMDEV PARTH DEENDAYAL

ARYAK ROY

SOHAM VIJAYKUMAR FALDU

HEET THAKRAR

19BCE0440

19BCE2165

19BCI0024

19BCI0274



Parallel and Distributed Computing-CSE4001

Lab Slot - L45 + L46

COMPUTER SCIENCE AND ENGINEERING

NOVEMBER 2021

CONTENTS

| | Page No. |
|---|----------|
| List of Figures | i |
| List of Tables | ii |
| Chapter 1: INTRODUCTION | 8 |
| Chapter 2: LITERATURE REVIEW | 14 |
| Chapter 3: METHODOLOGY AND IMPLEMENTATION | |
| Chapter 4: RESULTS AND DISCUSSION | |
| 4.1 Comparison and Analysis | |
| 4.2 Positives and Negatives | |
| Chapter 5: CONCLUSION AND FUTURE WORK | |
| REFERENCES | |

1. INTRODUCTION

In this modern world of technology and especially in this pandemic, everything just came online. Also, we know that the internet is full of information and knowledge. Everything is available on this vast network and anyone can access it from any corner of the world. Well-liked search engines like Google, Bing, Yahoo, DuckDuckGo etc. have made it very easier and useful. They use complex algorithms to process the user query and fetch the matching results and finally provide users with the data.

The process of data retrieval is becoming tougher day by day. The data added to the internet is in different forms. Every hour hundreds and thousands of new web pages are created and added to it. These pages consist of images, text, media like songs, videos and much more.

Search engines use crawlers or spiders to fetch or read the data over the internet. This data is classified and processed first to find the required data, as it can be many mime-types. The web pages are ranked and classified according to their relevance and data. The biggest challenge for these crawlers is that the data is not organised and don't have the same format. Even the languages change for each web page crawled.

After classification and fetching the data, it is processed using various NLP (Natural Language Processing) methods. Segmentation of words and sentences is carried out and unnecessary words are removed. The output is then parsed and tagged according to needs. This process gives important keywords and data tokens related to the query.

An information retrieval system is a system that finds unorganised data from another huge collection of data that meets the informational requirement of the user. In the case of web information retrieval, this huge data is the world wide web. These systems are trained again and again to improve their efficiency. Feedbacks are collected to know the pitfalls in the process. In every IR (information retrieval) model there are three components - document and query representation and the retrieval function.

2. LITERATURE REVIEW

| Sr. no | Paper and Year | Method/Ideology | Advantage and Limitation |
|--------|---|--|---|
| 1 | Paper [1] (The number in square braces is according to the papers mentioned in references) 2017 | The paper shows a method based on Ontology. It uses the semantic web data. | <ul style="list-style-type: none">• <i>Advantage</i> It makes it easy for the machine to understand the data by adding tags and annotations.• <i>Limitation</i> The entire data available isn't in the semantic form. We need to transform it, that is not easy as the World wide web is a massive collection of data. Also, there might be some data that can't be converted. |
| 2 | Paper [2] 2019 | It mainly focuses on the extraction of news data using ontology and semantic web concepts. | <ul style="list-style-type: none">• <i>Advantage</i> Queries related to search of news article can be processed faster and will improve efficiency.• <i>Limitation</i> The news database needs to be changed to semantic form. It could be difficult to make changes to such a huge database. |

| | | | |
|---|-----------------------|---|--|
| 3 | Paper [3] 2017 | The paper shows an algorithm making relations based on weight using the semantic web. It makes use of cosine similarity for the same. | <ul style="list-style-type: none"> • <i>Advantage</i> The concept of making relations using weights will improve the query searching process • <i>Limitation</i> This needs to train the system using large and relevant dataset otherwise it can give wrong results and output. |
| 4 | Paper [4] 2018 | It focuses on improving information retrieval by removing unnecessary informational data on the webpages. | <ul style="list-style-type: none"> • <i>Advantage</i> It makes the data succinct and more relevant to the query by filtering redundant content by using several hash algorithms. • <i>Limitation</i> Using it might also remove some important information if accuracy is low. It can cause loss to data retrieval in such case. |

| | | | |
|---|-----------------------|--|---|
| 5 | Paper [5] 2018 | It talks about the increasing data on social media and proposes a system to use semantic instead of syntactic approach. It also shows how selection of features can increase the efficiency of IR. | <ul style="list-style-type: none"> • <i>Advantage</i> It could improve performance of social media and also in other domains if we could deliver a similar architecture. • <i>Limitation</i> The system should be trained properly otherwise the efficiency could be low. And use of intelligence in this, might lead to wrong results. |
| 6 | Paper [6] 2020 | IR using the swarm technology and recurring patterns | <ul style="list-style-type: none"> • <i>Advantage</i> Faster and more accurate than traditional methods of information retrieval. Swarm technology increases the speed and efficiency of search. • <i>Limitation</i> Faster retrieval for recurrent models, but it might be slow for non-frequent phrases. |

| | | | |
|---|-----------------------|--|--|
| 7 | Paper [7] 2019 | It talks about increasing efficiency by reduction of dimension complexity. | <ul style="list-style-type: none"> • <i>Advantage</i> Reduction of complexity of vector increases efficiency of the IR process. • <i>Limitation</i> If there is some issue in reduction process, the vector might not remain compliant. |
| 8 | Paper [8] 2018 | It speaks about enhancing the performance of IR based on user's activity. | <ul style="list-style-type: none"> • <i>Advantage</i> User focused model gives better performance for individuals. • <i>Limitation</i> User focused model might be less efficient in some cases where things are more generalised. |
| 9 | Paper [9] 2018 | It shows use of TF-IDF method in IR and speaks about the result of research. | <ul style="list-style-type: none"> • <i>Advantage</i> It is a research on the website Detik.com. It shows how accurate it is, and where it can be increased. It retrieves all relevant document in most cases. • <i>Limitation</i> The website shows some non-relevant document in addition to the relevant. |

Paper [1] talks about a scheme of information retrieval based on ontology. In ontology, the documents are classified based on their properties or attributes and then grouped or connected with similar ones based on the relations among them. It also talks about the unorganised data of the web. It is very hard for the computer or crawlers to understand this data. Without understanding properly, the data cannot be retrieved easily.

This becomes easy using semantic web and ontology. The semantic web makes it easy for a machine to understand the data by adding tags and annotations to it.

Paper [2] mainly focuses on searching relevant news data from the collection or database of news available on the web. It also proposes to use semantic web technology like in paper [1]. The news articles contain various types of information and data. Every hour thousands of articles are updated to the web. Getting relevant article and providing the user with the result is a challenging task as it is very unorganised. The semantic web and the concept of ontology make it easy by adding proper tags to the articles stored in the database.

A few years back, data was searched from the collection based on keywords and titles. But it can't be used for the latest databases and collections. It might take ages for some complex queries to get the required result. The paper [3] talks about the formation of a table of relationships among data objects, by calculating weights. They have proposed a system that uses the cosine similarity method to calculate weights. This requires training the data accurately to improve the efficiency of the system.

Paper [4] talks about improving the retrieval of information by eliminating redundant informational data from web pages. These can include various advertisements and notifications. It can also be promoted content by several brands and companies. It shows the

usage of hash algorithms to measure the importance of information blocks on the webpage. The unnecessary blocks are removed from the fetched document and then served to the user.

Paper [5] speaks about the increasing data on social media like Twitter Instagram and Facebook. It proposes a system to use semantic instead of syntactic approach. It also shows how the selection of features and organising data can increase the efficiency of Information Retrieval. These algorithms based on the determination of features and distribution will help in swift retrieval from such a huge data collection.

Paper [6] proposes a system of information retrieval using swarm technology. It shows the complete process in two phases. The first is to disintegrate the data into assortments and then elicit recurring patterns from them. Two types of algorithms are used in this proposed system. Bio-inspired for the disintegration and Recursive Elimination for the recurring models. The paper [6] claims that the proposed system is better than the traditional approaches for IR by showing the results of various experiments.

Paper [7] talks about increasing efficiency by reduction of dimension complexity. The feature vector is lessened that leads to increased efficiency and speed of the process. In this process of reduction, the vector should remain compliant otherwise, it might create issues. The paper achieved this using extraction and selection of feature.

Paper [8] speaks about enhancing performance based on the user's activity. The main focus of the paper is the user. The proposed system is shaped in such a manner that it benefits the user's querying process. It also uses the semantic data methodology for making it understandable to the machine. It shows assuring results of their implementation in the cultural field. But this user focused model might be less efficient in some cases where things are more generalised.

Paper [9] shows research on information retrieval using the TF-IDF method. It then follows with the results of this research showing recall and precision values. The result discusses the average accuracy of the system in the case taken and some articles are retrieved that are not relevant.

3. METHODOLOGY AND IMPLEMENTATION

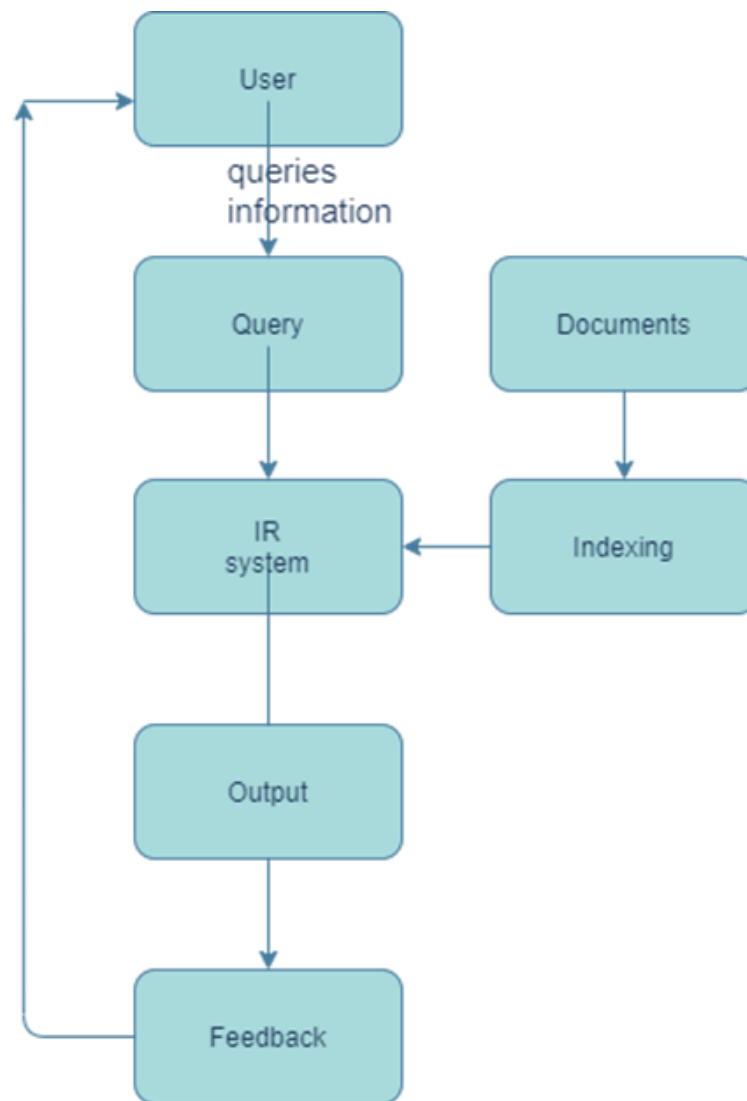


Fig. 1.1 IR system

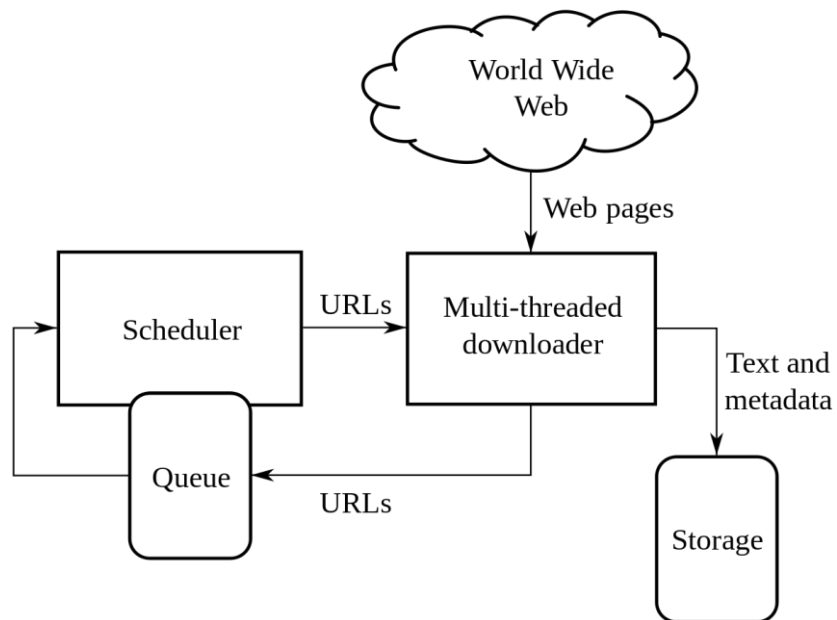


Fig. 1.2 Working of IR

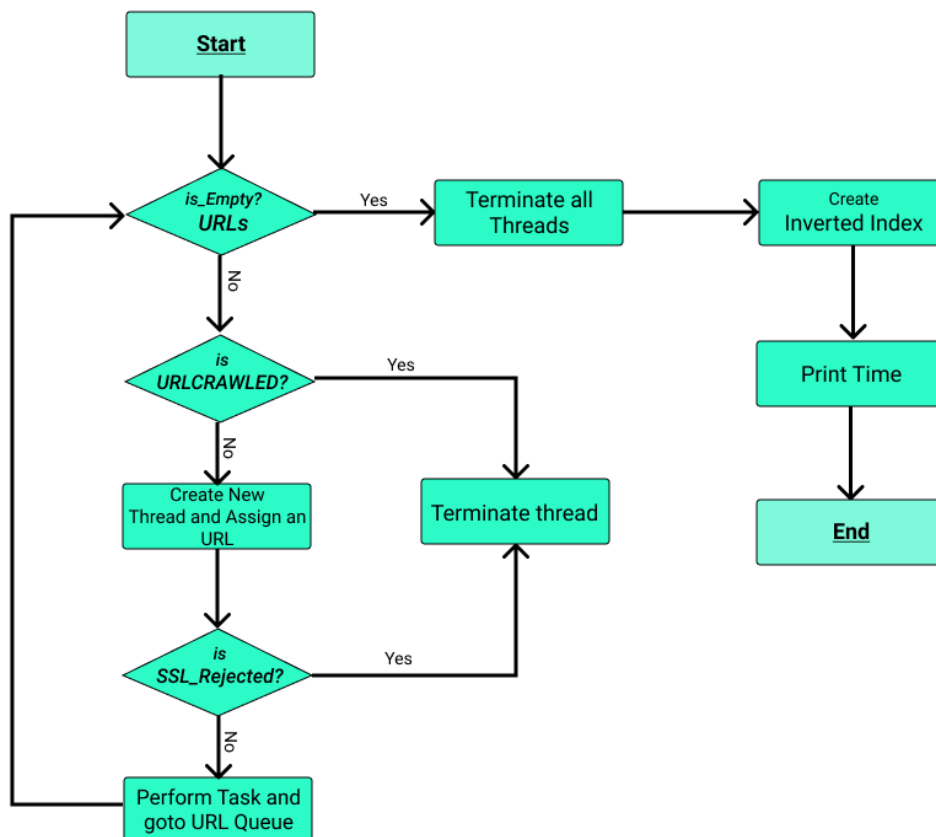


Fig. 2 Algorithm of Crawler

The process starts with the crawler where it receives the base URL and performs a limited depth first search till the specified level. While crawling links it maintains a list to which it adds the parsed URLs and before adding a URL it checks for the link's extension to exclude any backend code files such as .css, .js, and only include files with extensions as .htm, .html, .php or .jsp as they contain information. It also checks for links that have already been parsed and skips them to optimize performance.

Once a link is parsed then the link is crawled and its content and connected links are extracted and processed. The extracted content is parsed, tokenized from which all stop words are removed and then the unique terms remaining are sorted and their frequency is computed. If the link content is successfully extracted it is added to a list of crawled URLs or else the crawler moves to the next link in the list of URLs parsed.

After parsing and crawling all links. The inverted index is created from all of the extracted and processed content using an indexer function. In the inverted index each term has a list associated with it that contains the URL index, the frequency of that term in that URL and the index of instances in that specific URL content.

Three crawlers with different levels of parallelism have been implemented and analysed for the project. The configurations are as follows:

1. Serial Crawler and Indexer - SCSI
2. Concurrent Crawler and Indexer - CCCI
3. Concurrent Crawler with Serial Indexer - CCSI

4. RESULTS AND DISCUSSION

4.1 Comparison and Analysis

The third version - "Concurrent Crawler with serial Indexer", showed the most optimum results when tested.

```
File Edit Selection View Go Run Terminal Help serial_crawler.py - Crawler - Visual Studio Code

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Main thread - Crawling : https://en.wikipedia.org/wiki/Conformal_field_theory
Main thread - Crawling : https://en.wikipedia.org/wiki/Black_star_(semiclassical_gravity)
Main thread - Crawling : https://en.wikipedia.org/wiki/Raman_Sundrum
Main thread - Crawling : https://en.wikipedia.org/wiki/Gravitational_field
Main thread - Crawling : https://en.wikipedia.org/wiki/Dissipative_system
Main thread - Crawling : https://en.wikipedia.org/wiki/E7_(mathematics)
Main thread - Crawling : https://en.wikipedia.org/wiki/Stellar_mass_black_hole
Main thread - Crawling : https://en.wikipedia.org/wiki/Julius_Wess
Main thread - Crawling : https://en.wikipedia.org/wiki/Tachyon
Main thread - Crawling : https://en.wikipedia.org/wiki/NGC_3115
Main thread - Crawling : https://en.wikipedia.org/wiki/Pauli_exclusion_principle
Main thread - Crawling : https://en.wikipedia.org/wiki/Gravitomagnetism
Main thread - Crawling : https://en.wikipedia.org/wiki/Special:RecentChanges
Main thread - Crawling : https://en.wikipedia.org/wiki/Second_law_of_thermodynamics
Main thread - Crawling : https://en.wikipedia.org/wiki/Polarization_(waves)
Main thread - Crawling : https://en.wikipedia.org/wiki/Neutron_star_merger
Main thread - Crawling : https://en.wikipedia.org/wiki/Scientific_American
Main thread - Crawling : https://en.wikipedia.org/wiki/Hulse%E2%80%93Taylor_binary
Main thread - Crawling : https://en.wikipedia.org/wiki/Hermann_Bondi
Main thread - Crawling : https://en.wikipedia.org/wiki/Nonsingular_black_hole_models
Main thread - Crawling : https://en.wikipedia.org/wiki/George_Volkoff
Main thread - Crawling : https://en.wikipedia.org/wiki/Life_(magazine)
Main thread - Crawling : https://en.wikipedia.org/wiki/Compact_object
Main thread - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-0-471-87316-7
Main thread - Crawling : https://en.wikipedia.org/wiki/Galactic_Center
Main thread - Crawling : https://en.wikipedia.org/wiki/Vadim_Knizhnik

URLs parsed : 841
Total valid links found : 63429

Time taken for crawling
770.1447141170502

Time taken for indexing
46.758116722106934

(base) V:\PDC\Crawler>
```

Fig. 3 LEVEL-2 Serial Crawler and Indexer

```
hybrid_crawler.py - Crawler - Visual Studio Code

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Thread-809 - Crawling : https://en.wikipedia.org/wiki/Birkhoff%27s_theorem(Thread-817 - Crawling : https://en.wikipedia.org/wiki/Wikipedia:Good_articles
Thread-783 - Crawling : https://en.wikipedia.org/wiki/List_of_quasars(Thread-811 - Crawling : https://en.wikipedia.org/wiki/First_law_of_black_hole_mechanics(Thread-816 - Crawling : https://en.wikipedia.org/wiki/Ethan_Siegel
Thread-800 - Crawling : https://en.wikipedia.org/wiki/Second_superstring_revolution

Thread-823 - Crawling : https://en.wikipedia.org/wiki/Category:Articles_with_BNF_identifiers
Thread-815 - Crawling : https://en.wikipedia.org/wiki/Chandra_X-Ray_Observatory
Thread-814 - Crawling : https://en.wikipedia.org/wiki/Bosonic_string_theory
Thread-821 - Crawling : https://en.wikipedia.org/wiki/Kelvin

Thread-792 - Crawling : https://en.wikipedia.org/wiki/CERN
Thread-818 - Crawling : https://en.wikipedia.org/wiki/Quiver_diagram(Thread-798 - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-0-521-45586-0

Thread-808 - Crawling : https://en.wikipedia.org/wiki/Hawking_temperature(Thread-824 - Crawling : https://en.wikipedia.org/wiki/Point_mass

Thread-822 - Crawling : https://en.wikipedia.org/wiki/Blackbody_radiation
Thread-802 - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-0-471-19704-1
Thread-805 - Crawling : https://en.wikipedia.org/wiki/Stephen_Hawking
Thread-827 - Crawling : https://en.wikipedia.org/wiki/Tolman%E2%80%93Oppenheimer%E2%80%93Volkoff_limit
Thread-839 - Crawling : https://en.wikipedia.org/wiki/Dark_energy_star
Thread-803 - Crawling : https://en.wikipedia.org/wiki/Albert_Einstein
Thread-820 - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-0-393-31276-8
Thread-833 - Crawling : https://en.wikipedia.org/wiki/Vertex_operator_algebra
Thread-795 - Crawling : https://en.wikipedia.org/wiki/BKL_singularity

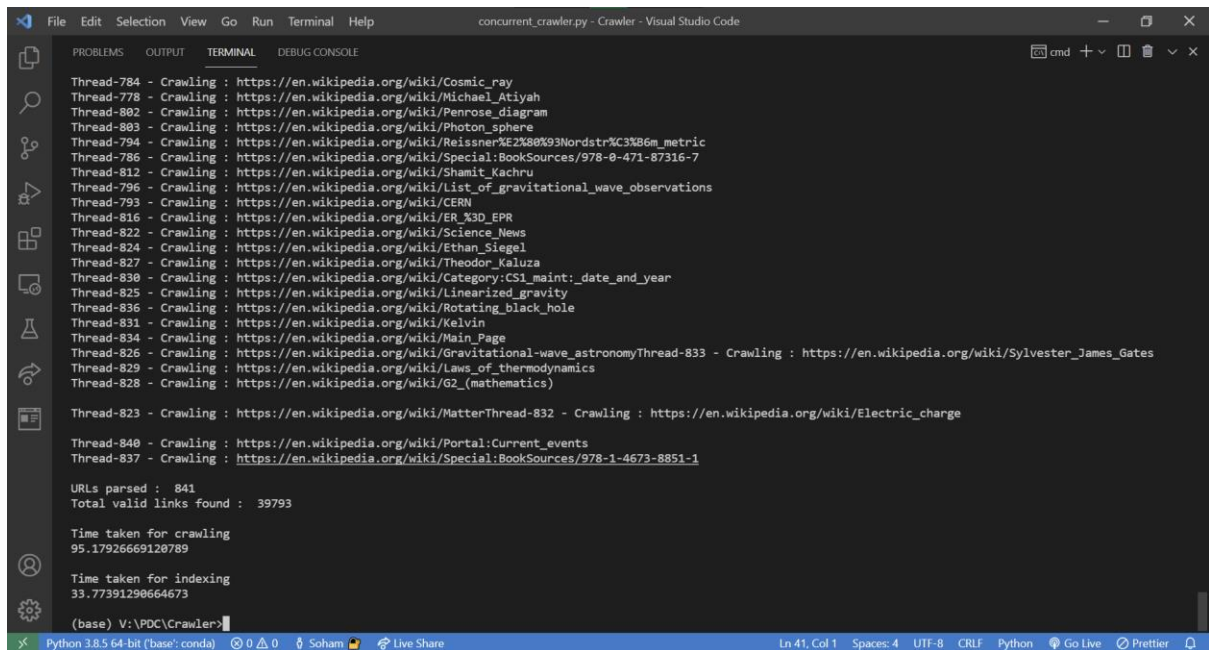
URLs parsed : 841
Total valid links found : 39756

Time taken for crawling
94.36630845069885

Time taken for indexing
22.610867500305176

(base) V:\PDC\Crawler>
```

Fig. 4 LEVEL-2 Concurrent Crawler with serial Indexer



```
File Edit Selection View Go Run Terminal Help concurrent_crawler.py - Crawler - Visual Studio Code
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
Thread-784 - Crawling : https://en.wikipedia.org/wiki/Cosmic_ray
Thread-778 - Crawling : https://en.wikipedia.org/wiki/Michael_Atiyah
Thread-802 - Crawling : https://en.wikipedia.org/wiki/Penrose_diagram
Thread-803 - Crawling : https://en.wikipedia.org/wiki/Photon_sphere
Thread-794 - Crawling : https://en.wikipedia.org/wiki/Reissner%E2%80%93Nordstr%C3%B6m_metric
Thread-786 - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-0-471-87316-7
Thread-812 - Crawling : https://en.wikipedia.org/wiki/Shamit_Kachru
Thread-796 - Crawling : https://en.wikipedia.org/wiki/List_of_gravitational_wave_observations
Thread-793 - Crawling : https://en.wikipedia.org/wiki/CERN
Thread-816 - Crawling : https://en.wikipedia.org/wiki/ER_N30_Epp
Thread-822 - Crawling : https://en.wikipedia.org/wiki/Science_News
Thread-824 - Crawling : https://en.wikipedia.org/wiki/Ethan_Siegel
Thread-827 - Crawling : https://en.wikipedia.org/wiki/Theodor_Kaluza
Thread-830 - Crawling : https://en.wikipedia.org/wiki/Category:CS1_maint:_date_and_year
Thread-825 - Crawling : https://en.wikipedia.org/wiki/Linearized_gravity
Thread-836 - Crawling : https://en.wikipedia.org/wiki/Rotating_black_hole
Thread-831 - Crawling : https://en.wikipedia.org/wiki/Kelvin
Thread-834 - Crawling : https://en.wikipedia.org/wiki/Main_Page
Thread-826 - Crawling : https://en.wikipedia.org/wiki/Gravitational-wave_astronomyThread-833 - Crawling : https://en.wikipedia.org/wiki/Sylvester_James_Gates
Thread-829 - Crawling : https://en.wikipedia.org/wiki/Laws_of_thermodynamics
Thread-828 - Crawling : https://en.wikipedia.org/wiki/G2_(mathematics)

Thread-823 - Crawling : https://en.wikipedia.org/wiki/MatterThread-832 - Crawling : https://en.wikipedia.org/wiki/Electric_charge

Thread-840 - Crawling : https://en.wikipedia.org/wiki/Portal:Current_events
Thread-837 - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-1-4673-8851-1

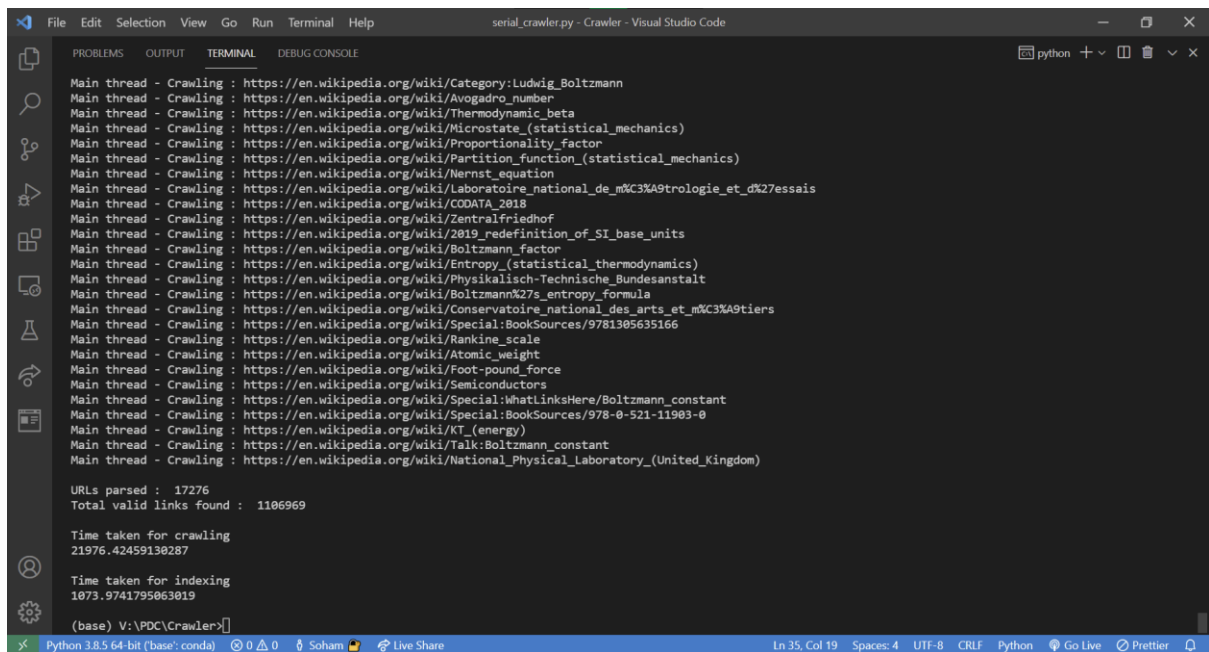
URLs parsed : 841
Total valid links found : 39793

Time taken for crawling
95.17926669120789

Time taken for indexing
33.77391290664673

(base) V:\PDC\Crawler>
```

Fig. 5 LEVEL-2 Concurrent Crawler and Indexer



```
File Edit Selection View Go Run Terminal Help serial_crawler.py - Crawler - Visual Studio Code
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
Main thread - Crawling : https://en.wikipedia.org/wiki/Category:Ludwig_Boltzmann
Main thread - Crawling : https://en.wikipedia.org/wiki/Avogadro_number
Main thread - Crawling : https://en.wikipedia.org/wiki/Thermodynamic_beta
Main thread - Crawling : https://en.wikipedia.org/wiki/Microstate_(statistical_mechanics)
Main thread - Crawling : https://en.wikipedia.org/wiki/Proportionality_factor
Main thread - Crawling : https://en.wikipedia.org/wiki/Partition_function_(statistical_mechanics)
Main thread - Crawling : https://en.wikipedia.org/wiki/Nernst_equation
Main thread - Crawling : https://en.wikipedia.org/wiki/Laboratoire_national_de_m%C3%A9tologie_et_d%27essais
Main thread - Crawling : https://en.wikipedia.org/wiki/CODATA_2018
Main thread - Crawling : https://en.wikipedia.org/wiki/Zentralfriedhof
Main thread - Crawling : https://en.wikipedia.org/wiki/2019_redefinition_of_SI_base_units
Main thread - Crawling : https://en.wikipedia.org/wiki/Boltzmann_factor
Main thread - Crawling : https://en.wikipedia.org/wiki/Entropy_(statistical_thermodynamics)
Main thread - Crawling : https://en.wikipedia.org/wiki/Physikalisch-Technische_Bundesanstalt
Main thread - Crawling : https://en.wikipedia.org/wiki/Boltzmann%27s_entropy_formula
Main thread - Crawling : https://en.wikipedia.org/wiki/Conservatoire_national_des_arts_et_m%C3%A9tiers
Main thread - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/9781305635166
Main thread - Crawling : https://en.wikipedia.org/wiki/Rankine_scale
Main thread - Crawling : https://en.wikipedia.org/wiki/Atomic_weight
Main thread - Crawling : https://en.wikipedia.org/wiki/Foot-pound_force
Main thread - Crawling : https://en.wikipedia.org/wiki/Semiconductors
Main thread - Crawling : https://en.wikipedia.org/wiki/Special:WhatLinksHere/Boltzmann_constant
Main thread - Crawling : https://en.wikipedia.org/wiki/Special:BookSources/978-0-521-11903-0
Main thread - Crawling : https://en.wikipedia.org/wiki/KT_(energy)
Main thread - Crawling : https://en.wikipedia.org/wiki/Talk:Boltzmann_constant
Main thread - Crawling : https://en.wikipedia.org/wiki/National_Physical_Laboratory_(United_Kingdom)

URLs parsed : 17276
Total valid links found : 1106969

Time taken for crawling
21976.42459130287

Time taken for indexing
1073.9741795063019

(base) V:\PDC\Crawler>
```

Fig. 6 LEVEL-3 Serial Crawler and Indexer

```
File Edit Selection View Go Run Terminal Help hybrid_crawler.py - Crawler - Visual Studio Code
```

```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE python + - [ ] [x] [v] [x]
```

```
Thread-15678 - Crawling : https://en.wikipedia.org/wiki/Current_(stream)
Thread-15680 - Crawling : https://en.wikipedia.org/wiki/Goldfinder
Thread-15685 - Crawling : https://en.wikipedia.org/wiki/Solar_radiation
Thread-15682 - Crawling : https://en.wikipedia.org/wiki/Star_Canopus_diving_accident
Thread-15690 - Crawling : https://en.wikipedia.org/wiki/HMS_Royal_George_(1756)
Thread-15700 - Crawling : https://en.wikipedia.org/wiki/Kelvin_temperature_scale
Thread-15670 - Crawling : https://en.wikipedia.org/wiki/Diving_physiology
Thread-15787 - Crawling : https://en.wikipedia.org/wiki/WOGI_Awards
Thread-15699 - Crawling : https://en.wikipedia.org/wiki/Steve_Lewis_(diver)
Thread-15723 - Crawling : https://en.wikipedia.org/wiki/Pressure-gradient_force
Thread-15730 - Crawling : https://en.wikipedia.org/wiki/Thermodynamic_limit
Thread-15721 - Crawling : https://en.wikipedia.org/wiki/Edwin_Clayton_Link
Thread-15740 - Crawling : https://en.wikipedia.org/wiki/Swedish_warship_Mars
Thread-15720 - Crawling : https://en.wikipedia.org/wiki/Partial_pressure
Thread-15739 - Crawling : https://en.wikipedia.org/wiki/Convective_condensation_level
Thread-15713 - Crawling : https://en.wikipedia.org/wiki/Diving_supervisor
Thread-15746 - Crawling : https://en.wikipedia.org/wiki/Atmospheric_convection
Thread-15711 - Crawling : https://en.wikipedia.org/wiki/Scott_Carpenter
Thread-15731 - Crawling : https://en.wikipedia.org/wiki/Reduced_gradient_bubble_model
Thread-15732 - Crawling : https://en.wikipedia.org/wiki/Underwater_Archaeology_Branch,_Naval_History_326_Heritage_Command
Thread-15737 - Crawling : https://en.wikipedia.org/wiki/Hyperthermia
Thread-15741 - Crawling : https://en.wikipedia.org/wiki/National_Oceanic_and_Atmospheric_Administration
Thread-15758 - Crawling : https://en.wikipedia.org/wiki/Global_Explorer_ROV
Thread-15729 - Crawling : https://en.wikipedia.org/wiki/Scuba_configuration
Thread-15747 - Crawling : https://en.wikipedia.org/wiki/Seahorse_ROUV
Thread-15750 - Crawling : https://en.wikipedia.org/wiki/Equipartition_theorem

URLs parsed : 9419
Total valid links found : 144765

Time taken for crawling
840.5122447013855

Time taken for indexing
161.81454372406006

(base) V:\PDC\Crawler>
```

Fig. 7 LEVEL-3 Concurrent Crawler with serial Indexer

```
concurrent_crawler.py - Crawler - Visual Studio Code

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Thread-16384 - Crawling : https://en.wikipedia.org/wiki/Fundamental_theorem_of_calculus
Thread-16388 - Crawling : https://en.wikipedia.org/wiki/Paterson%27s_worms
Thread-16393 - Crawling : https://en.wikipedia.org/wiki/Website
Thread-16403 - Crawling : https://en.wikipedia.org/wiki/D%3C38Crer%27s_solid
Thread-16405 - Crawling : https://en.wikipedia.org/wiki/Radiosonde
Thread-16409 - Crawling : https://en.wikipedia.org/wiki/Wikipedia:Shortcut
Thread-16390 - Crawling : https://en.wikipedia.org/wiki/Pythagoreanism
Thread-16414 - Crawling : https://en.wikipedia.org/wiki/Central_force
Thread-16417 - Crawling : https://en.wikipedia.org/wiki/Combinatorial_geometry
Thread-16422 - Crawling : https://en.wikipedia.org/wiki/Help:IPA/Hungarian
Thread-16423 - Crawling : https://en.wikipedia.org/wiki/Paul_Gordan
Thread-16425 - Crawling : https://en.wikipedia.org/wiki/Well-order
Thread-16420 - Crawling : https://en.wikipedia.org/wiki/Time_in_physics
Thread-16412 - Crawling : https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Mathematics/List_of_mathematicians_(G)
Thread-16429 - Crawling : https://en.wikipedia.org/wiki/Combinatorial_optimization
Thread-16415 - Crawling : https://en.wikipedia.org/wiki/Technical_University_of_Berlin
Thread-16433 - Crawling : https://en.wikipedia.org/wiki/Order_theory
Thread-16424 - Crawling : https://en.wikipedia.org/wiki/Trigonometry
Thread-16426 - Crawling : https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Mathematics/List_of_mathematics_articles_(F)
Thread-16413 - Crawling : https://en.wikipedia.org/wiki/Welsh_people
Thread-16430 - Crawling : https://en.wikipedia.org/wiki/Decimal_representation
Thread-16434 - Crawling : https://en.wikipedia.org/wiki/Model_theory
Thread-16427 - Crawling : https://en.wikipedia.org/wiki/PseudoforestThread-16435 - Crawling : https://en.wikipedia.org/wiki/Musica_universalis
Thread-16432 - Crawling : https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Mathematics/List_of_mathematicians_(K)
Thread-16428 - Crawling : https://en.wikipedia.org/wiki/Undirected_graph

URLs parsed : 9159
Total valid links found : 199760

Time taken for crawling
756.8589005470276

Time taken for indexing
115.78711152876721

(base) V:\PDC\Crawler>
```

Fig. 8 LEVEL-3 Concurrent Crawler and Indexer

Fig. 9 Indexer results

| | | Serial Crawler and Indexer | Concurrent Crawler and Indexer | Concurrent Crawler with serial Indexer |
|-----------|-----------------------------------|-------------------------------|--------------------------------------|--|
| Level - 2 | URLs parsed | 841 | 841 | 841 |
| | Valid links found | 63429 | 39793 | 39756 |
| | Crawling time(s) | 770.14 | 95.18 | 94.37 |
| | Indexing time(s) | 46.76 | 33.77 | 22.61 |
| | Lines written in indexing JSON | 896001 | 462584 | 455126 |

| | | Serial Crawler and Indexer | Concurrent Crawler and Indexer | Concurrent Crawler with serial Indexer |
|-----------|-------------------|-------------------------------|--------------------------------------|--|
| Level - 3 | URLs parsed | 17276 | 9159 | 9419 |
| | Valid links found | 1106969 | 199760 | 144765 |
| | Crawling time(s) | 21976.42 | 756.86 | 840.51 |

| | | | | |
|--|---------------------------------------|----------|---------|---------|
| | Indexing time(s) | 1073.97 | 115.79 | 161.81 |
| | Lines written in indexing JSON | 17439445 | 1600817 | 2295320 |

For the above results, URL used was
urlinput = https://en.wikipedia.org/wiki/Black_hole
base = <https://en.wikipedia.org/wiki/>

As displayed in the above table, SCSi is the slowest among all three. CCSi and CCCi both take 88% less time for crawling (In other words, CCSi and CCCi are 8 times faster than SCSi). This ratio becomes 96% for level-3 crawling (26 times faster). Whereas for indexing, CCCi takes more time than both CCSi and SCSi. This is because, for parallel indexing, sleep() is used to make sure threads don't collide or interfere with file writing during indexing. If sleep() is not used for CCCi, it may lead to improper file writing or data skipping. Hence, CCSi is the fastest among all three, as it uses parallel crawling and serial indexing of documents.

4.2 Positives and Negatives

As seen in the analysis section, parallel crawling takes about 88% less time than serial, but it has some limitations.

For level-3, both CCSi and CCCi take 96% less time, but the URLs crawled become almost half of SCSi. Whereas for level-2, even though they take 88% less time, there is no significant difference in URLs crawled. The difference in URLs crawled can lead to a drastic increase in Valid URLs found and indexing lines written by the SCSi.

In parallel crawling, many threads are requesting the same website or URL from the same IP. It might lead to the rejection of the SSL connection. Hence, that thread is aborted, decreasing the number of valid URLs for that current level. Due to this, the number of valid URLs found is more in the serial crawler. Sometimes this number is almost double that of the parallel crawler, but we can't unsee that parallel takes 88% less time than serial.

Now the lines written in the indexing file depend on the valid URLs found. Hence, the number of lines written after links crawled using a serial crawler is way more than that of a parallel crawler.

5. REFERENCES

- [1] Sunny Sharma, Arjun Kumar, Vijay Rana, "Ontology based informational retrieval system on the semantic web: Semantic Web Mining" , 2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS)

- [2] Smriti Arora, Niyati Baliyan, “Extraction and Analysis of Information in News Domain Using Semantic Web”, 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)
- [3] C S Saravana Kumar, M. Mohanapriya, C Kalaiarasan, “A New Approach for Information Retrieval in Semantic Web Mining Involving Weighted Relationship”, 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)
- [4] Uma R., Latha B., “Noise elimination from web pages for efficacious information retrieval”, Cluster Computing 22, 14583–14602 (2019). <https://doi.org/10.1007/s10586-018-2366-x>
- [5] Selvalakshmi B., Subramaniam M., “Intelligent ontology based semantic information retrieval using feature selection and classification”, Cluster Computing 22, 12871–12881 (2019). <https://doi.org/10.1007/s10586-018-1789-8>
- [6] Amol P. Bhopale, Ashish Tiwari, “Swarm optimized cluster based framework for information retrieval”, Expert Systems with Applications, Volume 154, 15 September 2020, 113441
- [7] C. S. Saravana Kumar, R. Santhosh, “Effective information retrieval and feature minimization technique for semantic web data”, Computers & Electrical Engineering, Volume 81, January 2020, 106518
- [8] Antonio M. Rinaldi, Cristiano Russo, “User-centered Information Retrieval using Semantic Multimedia Big Data”, 2018 IEEE International Conference on Big Data (Big Data)
- [9] Arfiani Nur Khusna, Indri Agustina, “Implementation of Information Retrieval Using Tf-Idf Weighting Method On Detik.Com’s”, 2018 12th International Conference on Telecommunication Systems, Services, and Applications (TSSA)