INFORMATION RETRIEVAL

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
UNIVERSITY OF WEST ATTICA

DEPARTMENT OF INFORMATION AND COMPUTER ENGINEERING

INFORMATION RETRIEVAL PROJECT

BUILDING AN ACADEMIC PAPER SEARCH ENGINE

WORK DETAILS

LABORATORY RESPONSIBILITY: TSELENTI PANAGIOTA
DELIVERY DATE: 1/29/2024
SUBMISSION DEADLINE: 1/29/2024

INFORMATION RETRIEVAL

STUDENT DETAILS 1

STUDENT PHOTO:

NAME: ATHANASIOU VASILIOS EVANGELOS
STUDENT ID: 19390005
STUDENT SEMESTER: 9th
STUDENT STATUS : UNDERGRADUATE
PROGRAM OF STUDY : UNIWA

RECOVERY OF INFORMATION

STUDENT DETAILS 2

STUDENT PHOTO:

NAME: TATSIS PANTELIS
STUDENT ID: 20390226
STUDENT SEMESTER: 7th
STUDENT STATUS : UNDERGRADUATE
PROGRAM OF STUDY : UNIWA

CONTENTS

Steps

1. Web Crawler

1.a. Select target site

1.a.1 Planning

The target site is the academic papers repository arXiv . The design model concerns the retrieval of the page with the results produced by the website's search engine from one or more random user queries, with the ultimate goal of creating the repository (dataset) of the local search engine, on which searches will be made with user queries.

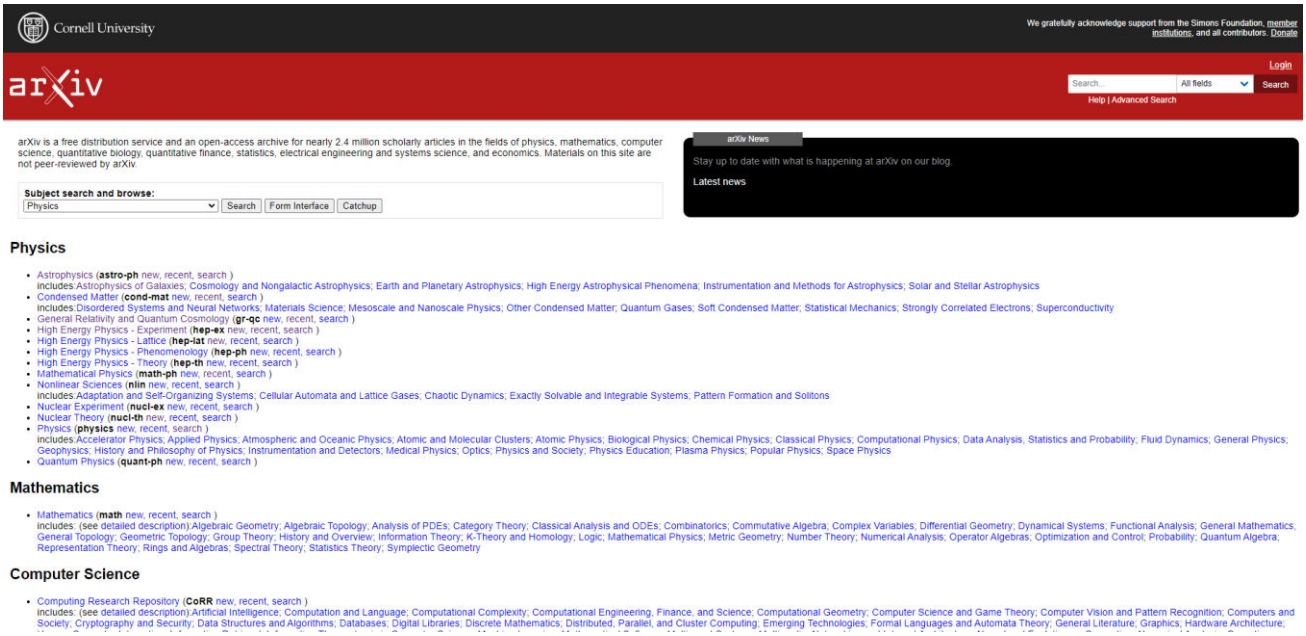


Figure 1.A.1 The target site arXiv

1.a.2 Implementation

Initially, the number of user queries and their selection is random in each execution of the local search engine (main.py). The questions are limited from 2 to 8 and specifically it has been chosen to be the titles of the basic courses on which academic papers have been posted. The implementation model involves sending an HTTP-GET request to the website's search engine link along with the corresponding random user query to display results. The process is iterative since the number of questions is more than 1. It is worth noting that for each question, the link that displays 100 results of academic papers has been selected. For example, for the user queries Physics, Statistics and Computer, 3 HTTP-GET requests are sent to the respective links:

https://arxiv.org/search/?query=Physics&searchtype=all&abstracts=show&order=-announced_date_first&size=100

https://arxiv.org/search/?query=Statistics&searchtype=all&abstracts=show&order=-announced_date_first&size=100

https://arxiv.org/search/?query=Computer&searchtype=all&abstracts=show&order=-announced_date_first&size=100

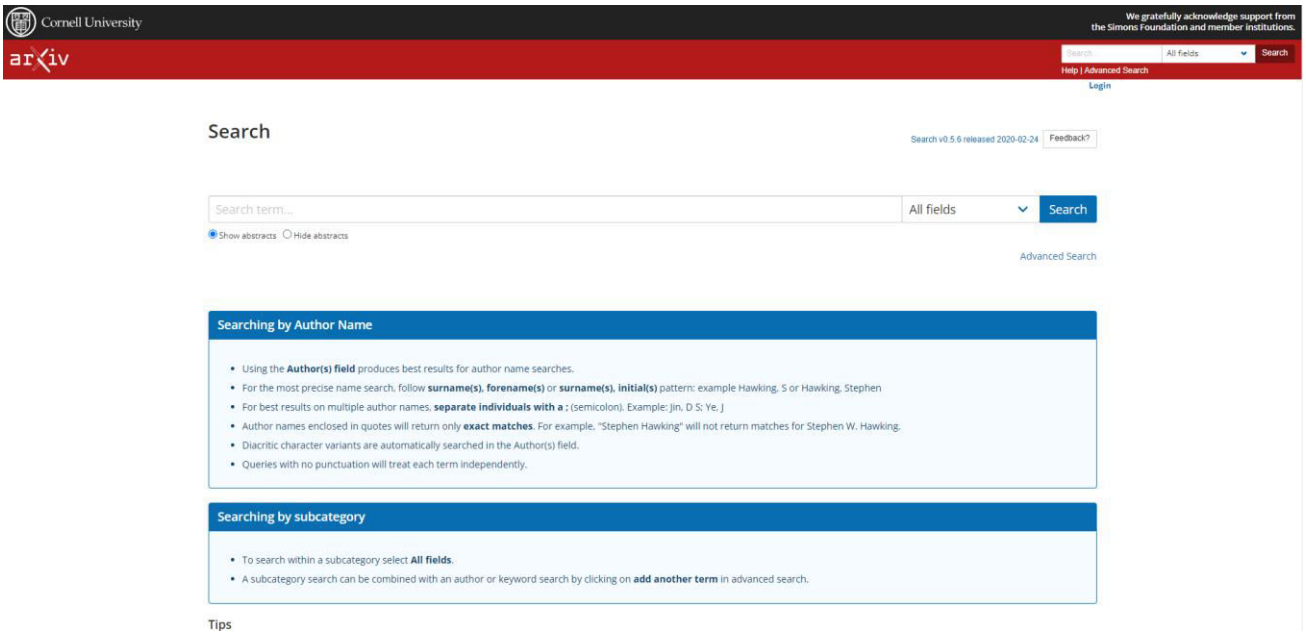


Figure 1.A.2 The arXiv search engine

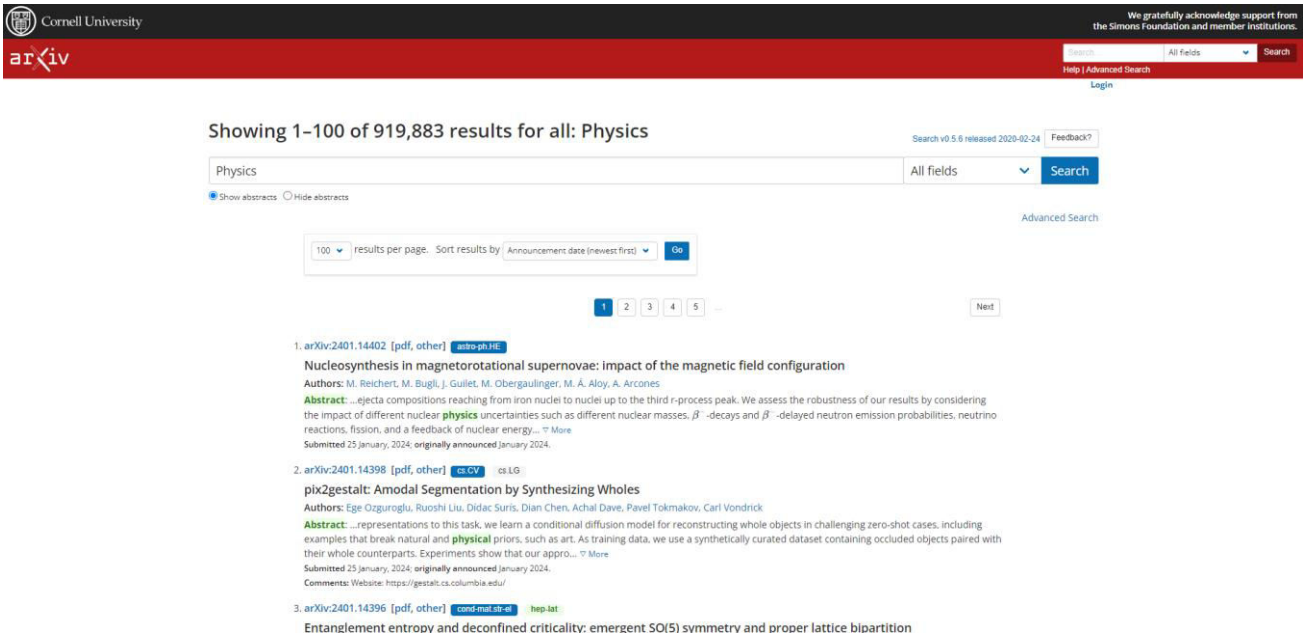


Figure 1.A.3 The first 100 search results for Physics

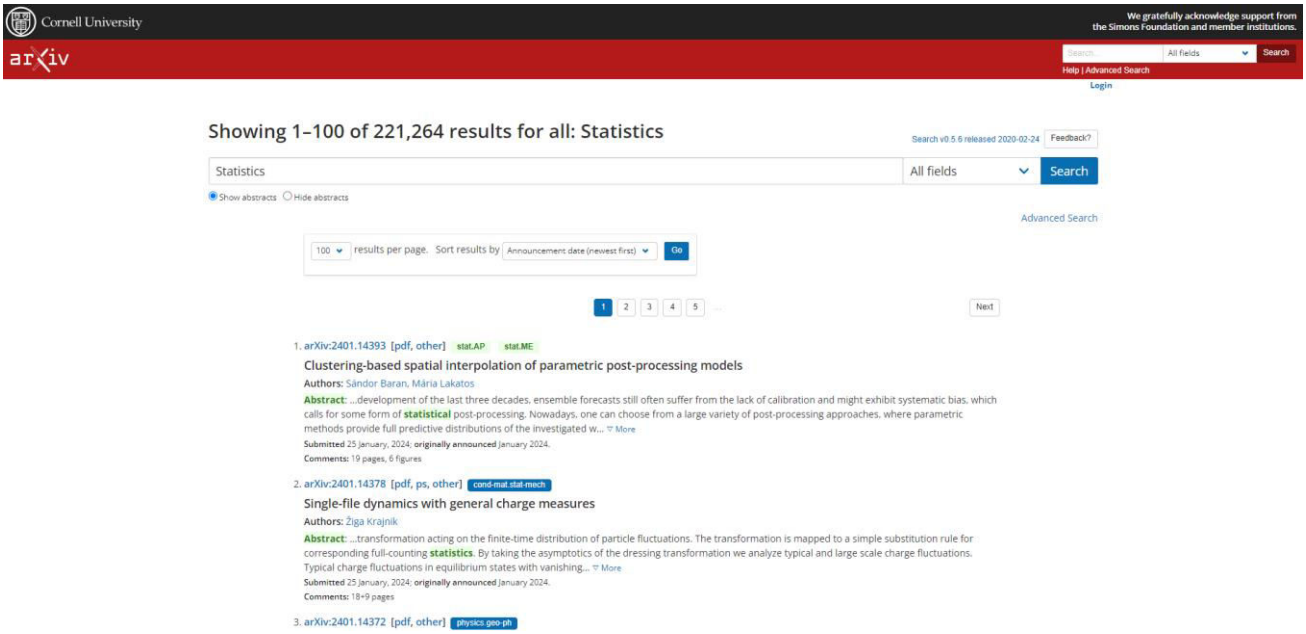


Figure 1.A.4 The first 100 search results of the query Statistics

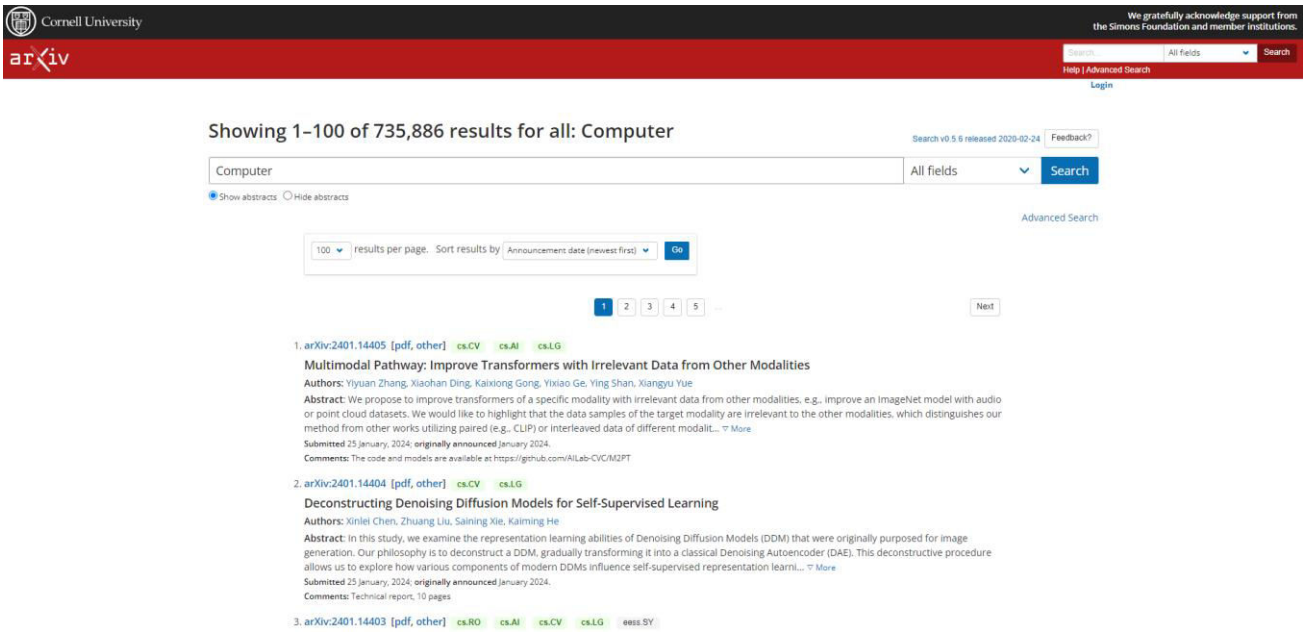


Figure 1.A.5 The first 100 search results for the query Computer

1.a.3 Evaluation

The local engine supports exception handling for the case that the return code of an HTTP-GET request is other than 200. Strictly speaking, a code of 200 means that the request was served and the page on which the request was made was successfully retrieved. It is worth noting that the selection of random user queries for the creation of the dataset is from 2 to 8, so that the size of the dataset is from 200 to 800 tasks.

1.b. Web crawler implementation

1.b.1 Planning

The web crawler is designed to collect the metadata of the academic papers (web scrape) from the page displayed by the website search engine by inputting one of the random user queries. The process is iterative from the fact documented in the implementation of step 1.a Select a target site.

1.b.2 Implementation

The implementation follows the Python web-crawler BeautifulSoup and takes place in a separate module (web_crawler.py), where it returns the page requested for retrieval through a successful HTTP-GET request, in unstructured HTML format. The data collected from the tasks are as follows:

- Title
- Authors
- Courses and related sub-courses
- Summary
- Comments
- Publication date
- Link to download the work in pdf format

In the implementation, an identifier integer (doc_id), which is unique, is added to the metadata of each task. This technique aims to easily retrieve a task from the local search engine without having to have all its metadata as an argument. The metadata is collected exactly as it is embedded in the unstructured HTML format and with few variations to store the main content of each field. It is worth noting that the authors and the courses and sub-courses related to the work, have been stored in a structure instead of a single textual content for the better organization of the data. Also, some tasks have no content in the comments field, so it is replaced with a blank. Finally, each task's data is stored in a dictionary structure.

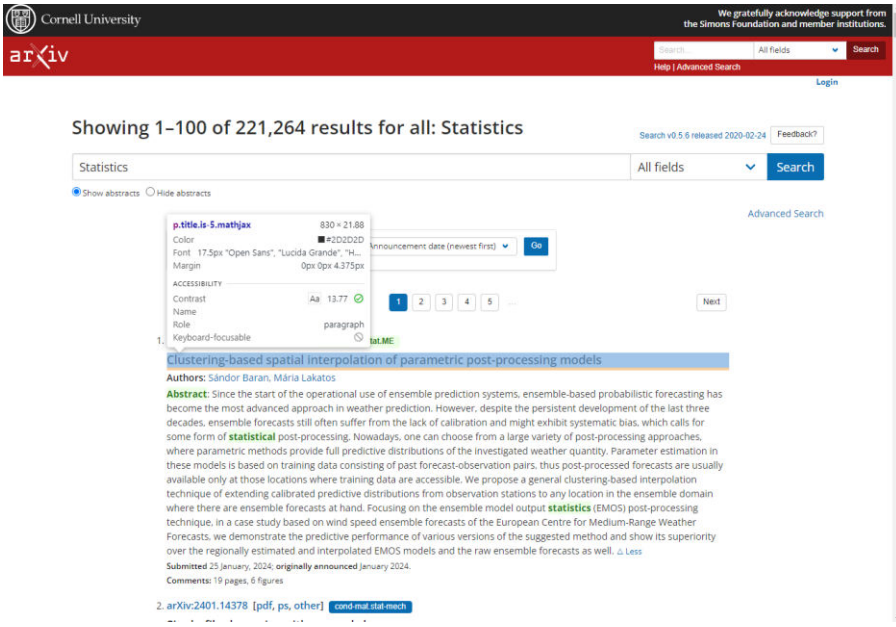


Figure 1.B.1 Retrieving the title of the work

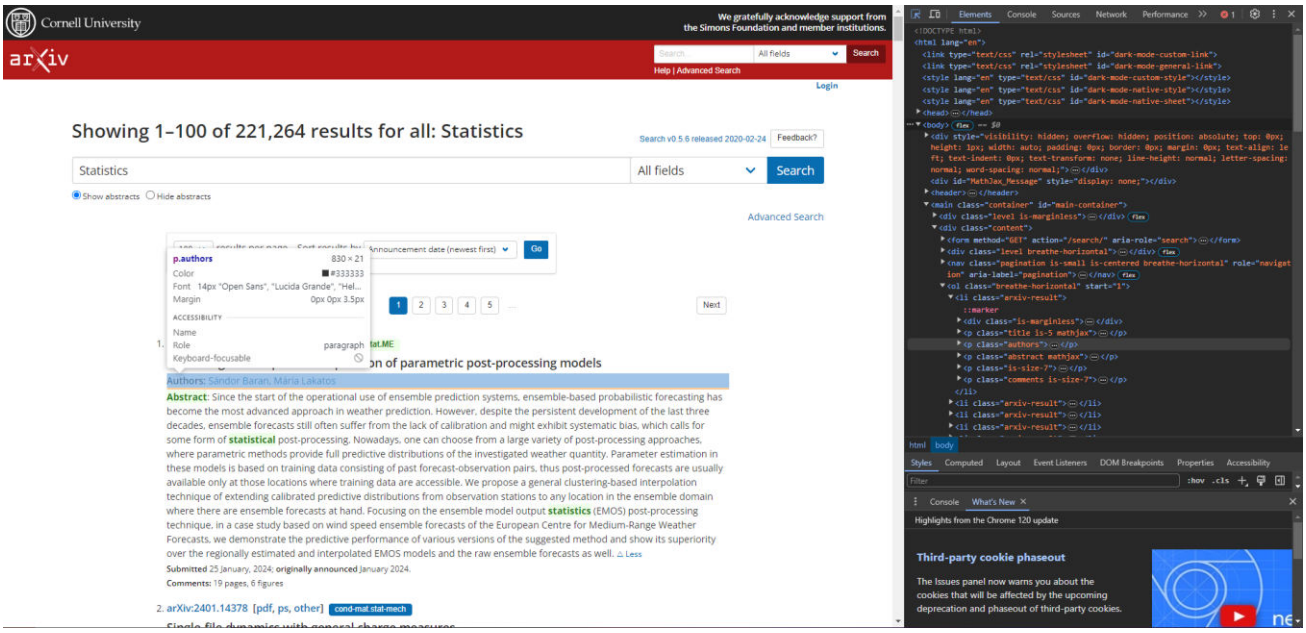


Figure 1.B.2 Retrieving the authors of the paper

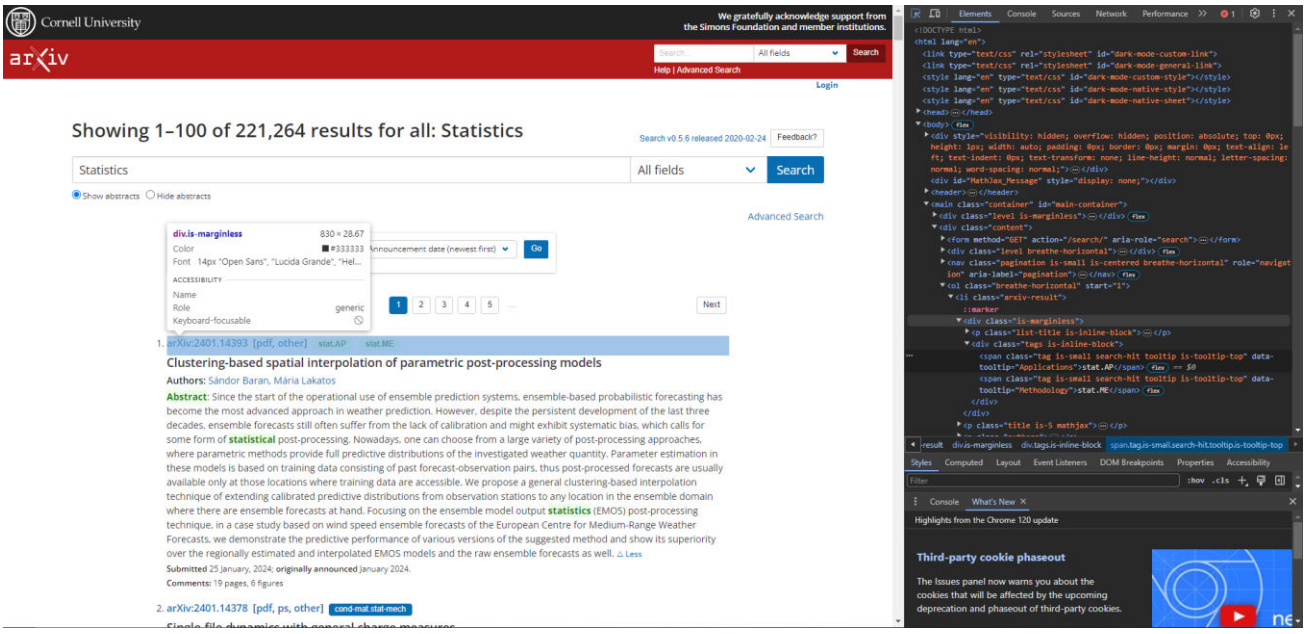


Figure 1.B.3 Retrieval of work-related courses

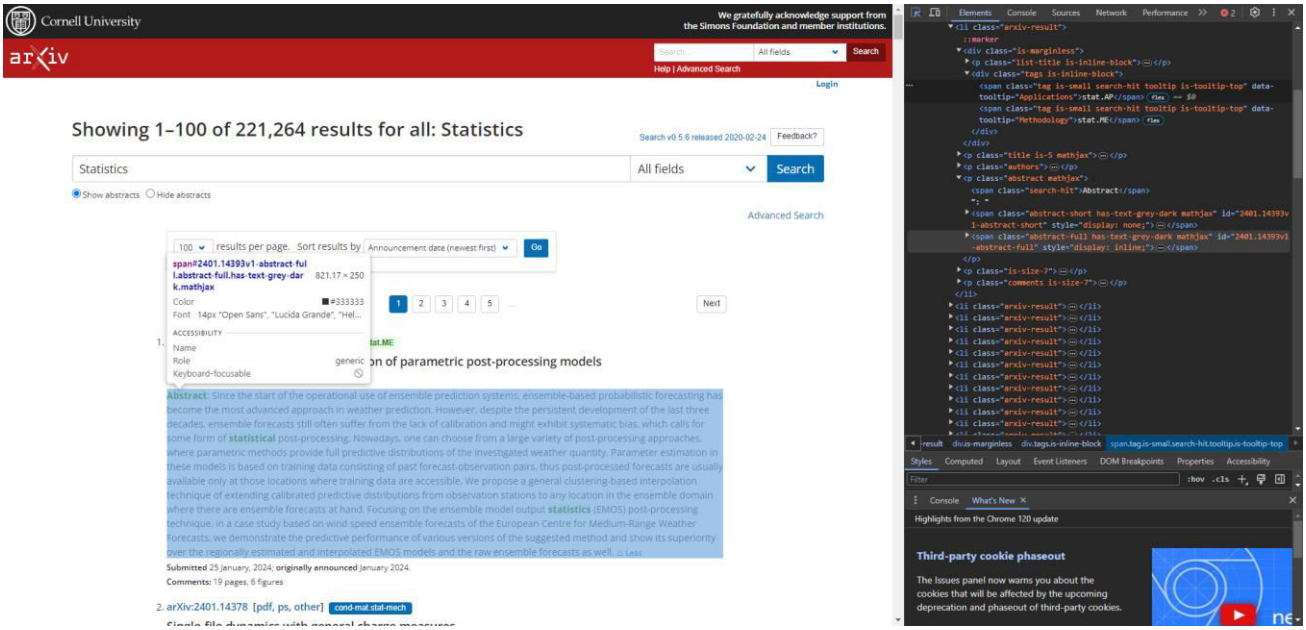


Figure 1.B.4 Retrieve the job summary

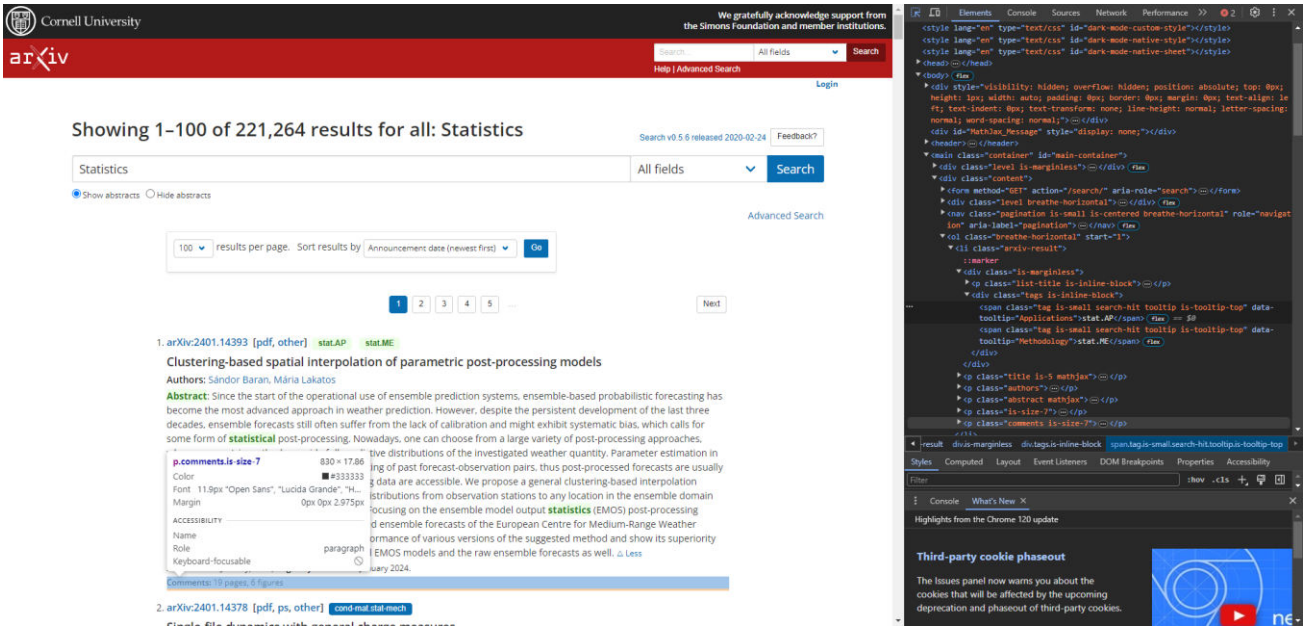


Figure 1.B.5 Retrieving the comments of the work

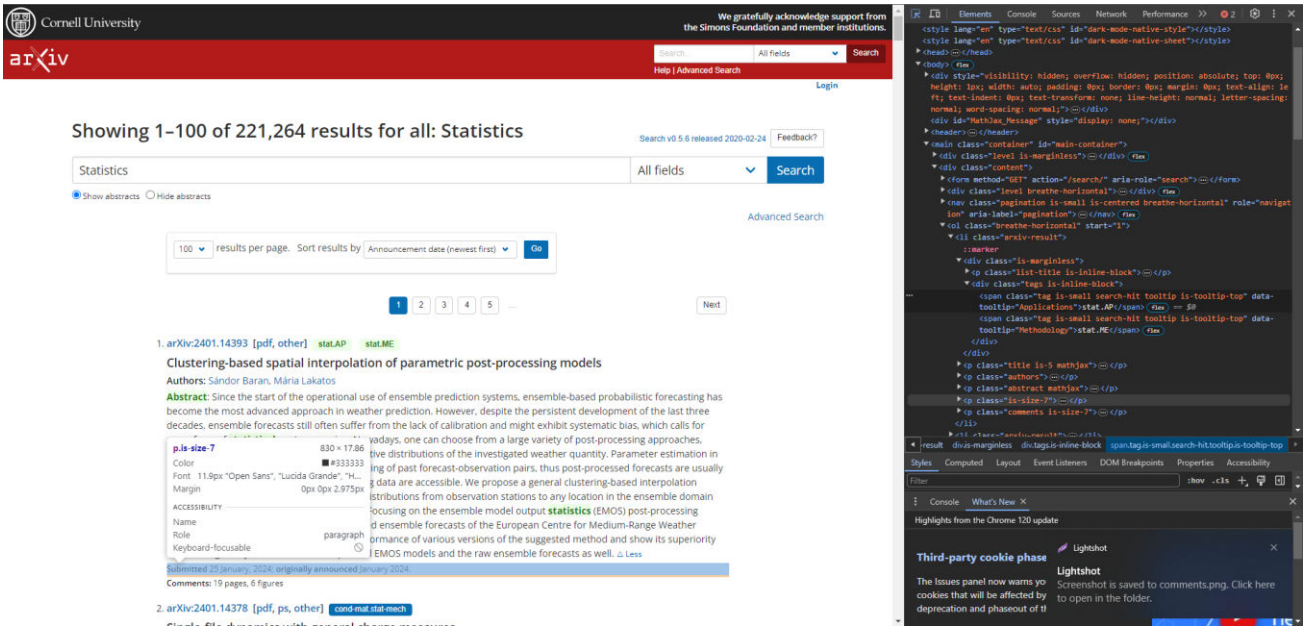


Figure 1.B.6 Retrieving the publication date of the work

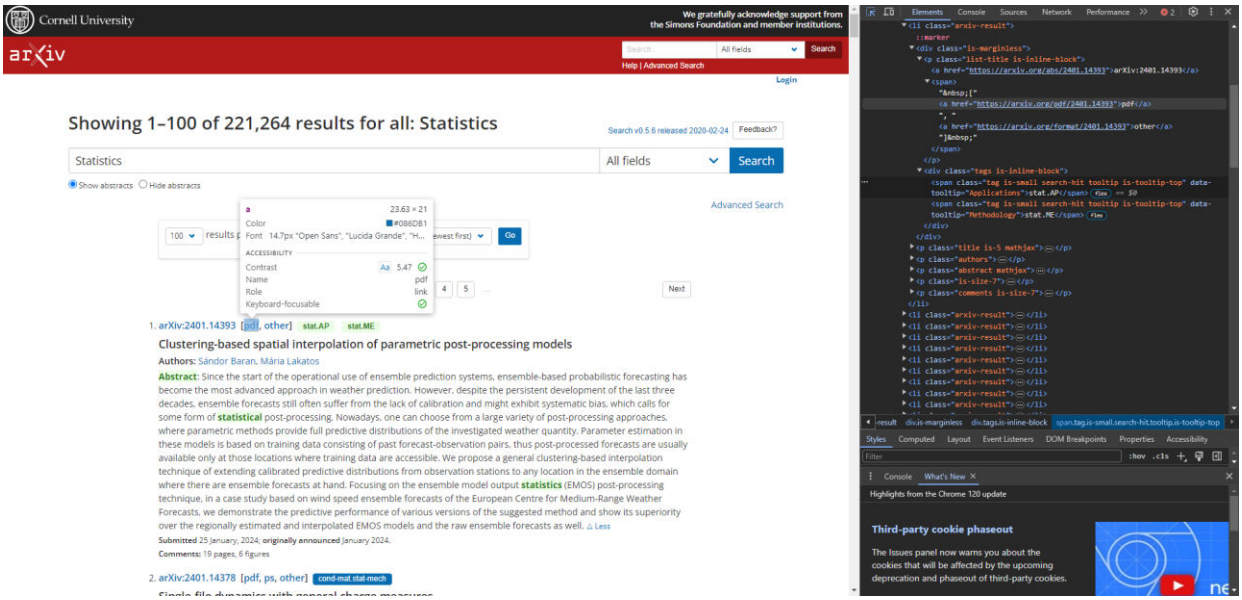


Figure 1.B.7 Retrieving the link to download the work in pdf

1.b.3 Evaluation

The original implementation of the scraper was more complicated, as, for each paper, the collection of metadata (web scrape) took place on the separate pages of each paper with a link arxiv.org/abs/XXX.XXXX (eg https:/ /arxiv.org/abs/2401.14393). This created significant delays in the harvester's performance, as collecting data from 200-800 jobs meant sending HTTP-GET requests to 200-800 job links to display each job's respective data pages in unstructured HTML . The final implementation reduces page retrieval from 200-800 links to 2-8 links, where job data is collected as-is as embedded in unstructured HTML and stored cleanly according to the necessary techniques, in a data structure, achieving this way to reliable dataset size and gleaner performance.

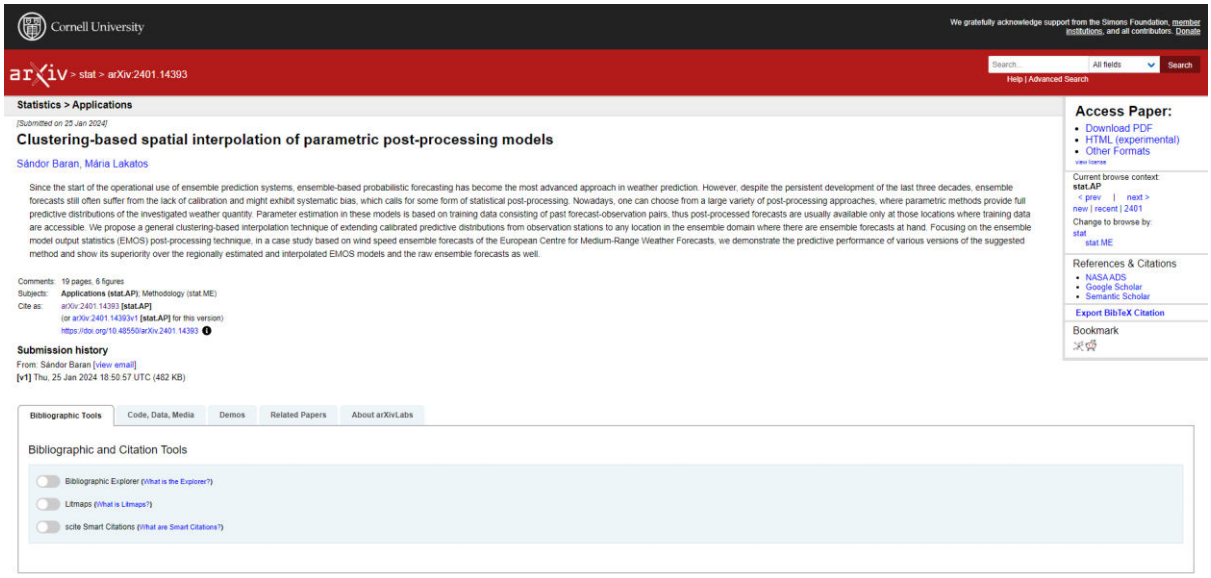


Figure 1.B.8 The job page with all the data

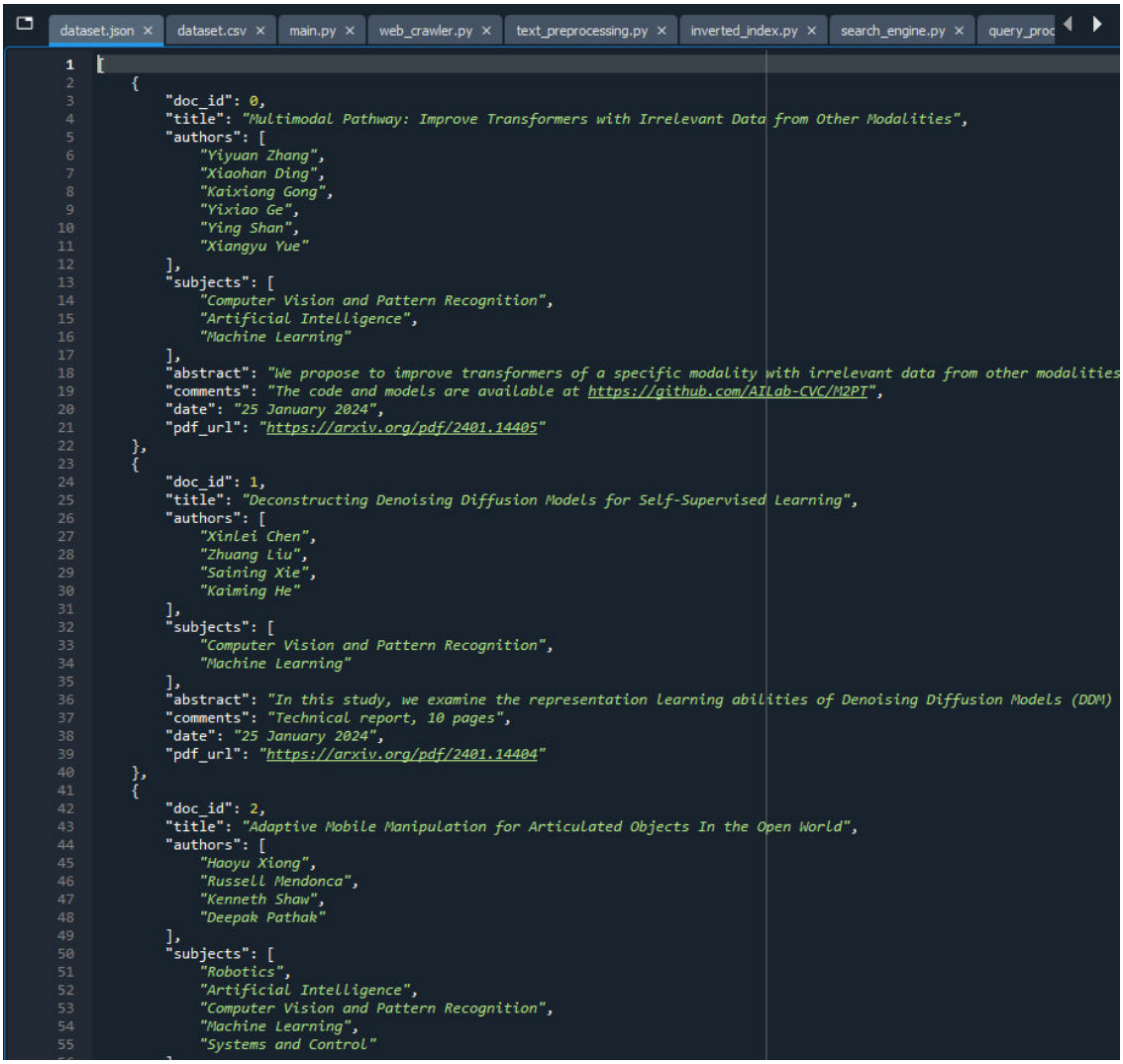
1.c. Store data in a structured format

1.c.1 Planning

The design adopts the structured JSON format instead of the CSV format to store the task data, and by performing this step, the creation of the local search engine dataset is completed

1.c.2 Implementation

The collection of task metadata from the unstructured HTML format is completed in the implementation of step 1.b. Web crawler implementation, so what remains is to write the data in a .json file and the repository (dataset) of the local search engine is this. The implementation takes place in the main program (main.py), where the results returned by the gleaner in dictionary format are stored in JSON format.

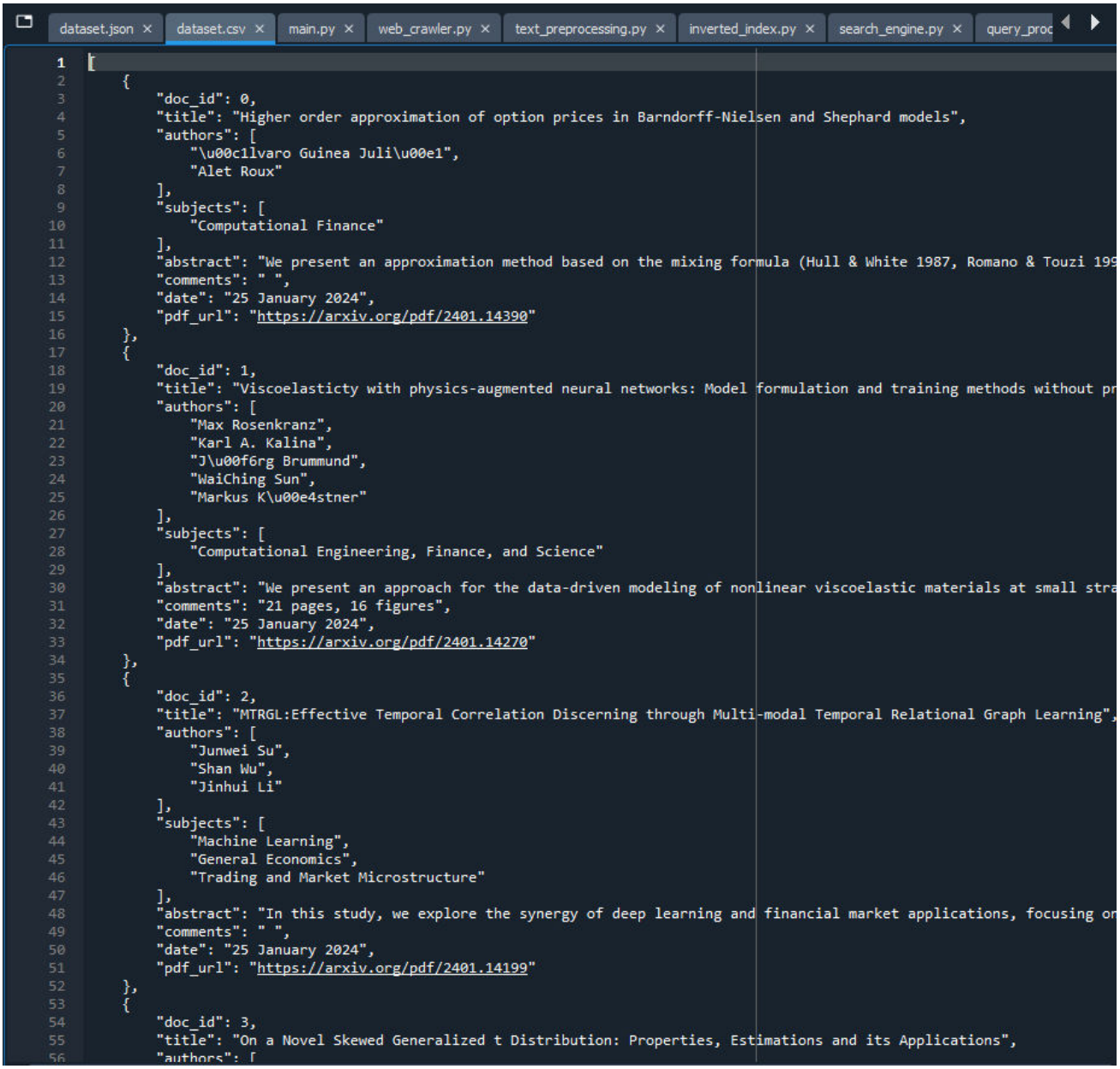



```
[
    {
        "doc_id": 0,
        "title": "Multimodal Pathway: Improve Transformers with Irrelevant Data from Other Modalities",
        "authors": [
            "Yiyuan Zhang",
            "Xiaohan Ding",
            "Kaixiong Gong",
            "Yixiao Ge",
            "Ying Shan",
            "Xiangyu Yue"
        ],
        "subjects": [
            "Computer Vision and Pattern Recognition",
            "Artificial Intelligence",
            "Machine Learning"
        ],
        "abstract": "We propose to improve transformers of a specific modality with irrelevant data from other modalities
        "comments": "The code and models are available at https://github.com/AILab-CVC/M2PT",
        "date": "25 January 2024",
        "pdf_url": "https://arxiv.org/pdf/2401.14405"
    },
    {
        "doc_id": 1,
        "title": "Deconstructing Denoising Diffusion Models for Self-Supervised Learning",
        "authors": [
            "Xinlei Chen",
            "Zhuang Liu",
            "Saining Xie",
            "Kaiming He"
        ],
        "subjects": [
            "Computer Vision and Pattern Recognition",
            "Machine Learning"
        ],
        "abstract": "In this study, we examine the representation learning abilities of Denoising Diffusion Models (DDM)
        "comments": "Technical report, 10 pages",
        "date": "25 January 2024",
        "pdf_url": "https://arxiv.org/pdf/2401.14404"
    },
    {
        "doc_id": 2,
        "title": "Adaptive Mobile Manipulation for Articulated Objects In the Open World",
        "authors": [
            "Haoyu Xiong",
            "Russell Mendonca",
            "Kenneth Shaw",
            "Deepak Pathak"
        ],
        "subjects": [
            "Robotics",
            "Artificial Intelligence",
            "Computer Vision and Pattern Recognition",
            "Machine Learning",
            "Systems and Control"
```


Figure 1.C.1 The repository in JSON format

1.c.3 Evaluation

The choice of JSON format instead of CSV to store task data in a structured format is justified by the fact that there are data (authors, courses) that are more readable in a structure (e.g. list) that supports JSON rather than a single comma-separated text (CSV). The volume of information ranges from 200 to 800 tasks, where each task includes 8 fields of data (documentation in the implementation of step 1.b. Web-crawler implementation). Therefore, leveraging large amounts of data is more easily manageable in JSON format than in simple CSV format.




```
[
    {
        "doc_id": 0,
        "title": "Higher order approximation of option prices in Barndorff-Nielsen and Shephard models",
        "authors": [
            "\u00c1lvaro Guinea Juli\u00e1",
            "Alet Roux"
        ],
        "subjects": [
            "Computational Finance"
        ],
        "abstract": "We present an approximation method based on the mixing formula (Hull & White 1987, Romano & Touzi 199
        "comments": " ",
        "date": "25 January 2024",
        "pdf_url": "https://arxiv.org/pdf/2401.14390"
    },
    {
        "doc_id": 1,
        "title": "Viscoelasticty with physics-augmented neural networks: Model formulation and training methods without pr
        "authors": [
            "Max Rosenkranz",
            "Karl A. Kalina",
            "J\u00f6rg Brummund",
            "WaiChing Sun",
            "Markus K\u00e4stner"
        ],
        "subjects": [
            "Computational Engineering, Finance, and Science"
        ],
        "abstract": "We present an approach for the data-driven modeling of nonlinear viscoelastic materials at small stra
        "comments": "21 pages, 16 figures",
        "date": "25 January 2024",
        "pdf_url": "https://arxiv.org/pdf/2401.14270"
    },
    {
        "doc_id": 2,
        "title": "MTRGL:Effective Temporal Correlation Discerning through Multi-modal Temporal Relational Graph Learning",
        "authors": [
            "Junwei Su",
            "Shan Wu",
            "Jinhui Li"
        ],
        "subjects": [
            "Machine Learning",
            "General Economics",
            "Trading and Market Microstructure"
        ],
        "abstract": "In this study, we explore the synergy of deep learning and financial market applications, focusing on
        "comments": " ",
        "date": "25 January 2024",
        "pdf_url": "https://arxiv.org/pdf/2401.14199"
    },
    {
        "doc_id": 3,
        "title": "On a Novel Skewed Generalized t Distribution: Properties, Estimations and its Applications",
        "authors": [
```


Figure 1.C.2 The repository in CSV format

2. Text Processing

2.a. Planning

The design follows the preprocessing of the textual content (Abstract field) of the repository created in step 1 , with various text preprocessing techniques. Text preprocessing is one of the most basic natural language processing (NLP) techniques, where it achieves the matching of data terms with the search query, so that the ambiguity that is apparently present in the wording of the user query, does not affect the search result .

Preprocessing was designed with the ultimate goal of grouping together different wording versions of a user query so that there is not much variance in search results when the same desired result is sought but with different wording. The design follows various text preprocessing techniques to preprocess the repository job data. The techniques designed for implementation are as follows:

- Tokenization: The separation of text into verbal units
- Punctuation characters removal: Removal of all punctuation marks
- Special characters removal: Removal of all special characters
- Normalization: The replacement of uppercase letters with their lowercase counterparts
- Stopwords removal: Removal of "forbidden" words
- Stemming: Removing endings and keeping the main stem of the term

2.b. Implementation

The implementations of the text preprocessing techniques take place in a separate module (text_preprocessing.py) and are as follows:

- Tokenization: Use nltk.word_tokenize() to split text into word units and separate punctuation
- Punctuation characters removal: Removal of all terms belonging to the set string.punctuation, that is, punctuation marks.
- Special characters removal: Removal of all terms that are not numbers or letters
- Normalization: Replace uppercase letters with their lowercase counterparts using the .lower() routine
- Stopwords removal: Remove all terms belonging to the set stopwords.words('english') from nltk.corpus. Usually, these are terms that appear frequently in texts such as articles, pronouns, adverbs, etc.
- Stemming: Remove endings and keep the main term stem using nltk.stem's PorterStemmer() algorithm

Given that a search query (query) will also be needed to search for jobs, this should also be pre-processed (details in step 4 Search engine). In search queries that include Boolean operations, the pre-processing technique "Stopwords removal" is not performed, as this will also remove the logical operators AND, NOT, OR from the query (details in step 4.b.1 Boolean Retrieval).

2.c. Evaluation

The pre-processing is done in the order presented in the design and implementation . The evaluation is based on the choice of text preprocessing techniques, as well as the order in which they take place. The evaluations of the functions of each technique are as follows:

- Tokenization: The separation of the text into verbal units contributes to the management of words and punctuation marks and is therefore applied 1 ^{n}.
- Punctuation characters removal: The removal of punctuation marks helps to clean the text from unnecessary information and is applied 2nd ^{for} this reason. However, this technique may affect the understanding of a text that uses punctuation to distinguish terms (eg by removing the punctuation on a date 19/03/2001, the preprocessed date is displayed as 3 numbers 19 03 2001 with resulting in a loss of information)

- Special characters removal: Special characters removal also helps to clean the text of unnecessary information. However, as with the punctuation characters removal technique, it may affect the understanding of the text (e.g. by removing the special characters in an email jondoe@gmail.com, the preprocessed email is rendered as jondoe gmail com, resulting in loss of information)
- Normalization: Normalization helps bring terms into a common form by converting all uppercase letters to lowercase.
- Stopwords removal: Stopwords removal cleans the text of words that appear too often and do not provide important information. However, it may affect the understanding of the text in terms accompanied by forbidden words (eg, by removing the forbidden words in King of Denmark, the preprocessed text is rendered as king denmark, thereby losing information).
- Stemming: The process of 'stemming' removes the endings from words and keeps the main stem of the word, thus succeeding in grouping different versions of a word into a common stem. However, here too there is a risk of information loss (e.g. in the boolean search queries operational AND research, operating AND system, operative AND dentistry, the stemming process groups the terms operational, operating, operative into the common stem oper, resulting in a loss information)

The lemmatization technique offers more accurate results, as it takes into account the grammar of the word to convert it to its basic form in contrast to the stemming technique which simply removes endings from words and presents a potential problem of information loss. However, given that the dataset is small compared to large search engine repositories (consisting of millions of documents), the stemming technique offers simplicity and speed in search engine performance, and this is why it was preferred over the lemmatization technique.

2.d. Application

For the needs of the text preprocessing application, the "abstract" field of the work was used with the following data:

{

"doc_id": 0,

"title": "Nucleosynthesis in magnetorotational supernovae: impact of the magnetic field configuration",

"authors": [

"M. Reichert",

"M. Bugli",

"J. Guilet",

"M. Obergaulinger",

"M. \u00c1. Aloy",

"A. Arcones"

],

"subjects": [

"High Energy Astrophysical Phenomena"

],

"abstract": "The production of heavy elements is one of the main by-products of the explosive end of massive stars. A long sought goal is finding differentiated patterns in the nucleosynthesis yields, which could permit identifying a number of properties of the explosive core. Among them, the traces of the magnetic field topology are particularly important for \\emph{extreme} supernova explosions. We investigate the nucleosynthesis of five state-of-the-art magnetohydrodynamic models that have been previously calculated in full 3D and that involve an accurate neutrino transport (M1). One of the models does not contain any magnetic field and synthesizes elements around the iron group, in agreement with other CC-SNe models in literature models host a strong magnetic field of the same intensity, but with different topology.For the first time, we investigate the nucleosynthesis of MR-SNe models with a quadrupolar magnetic field and a 90 degree tilted dipole. We obtain a large variety of ejecta compositions reaching from iron nuclei to nuclei up to the third r-process peak. We assess the robustness of our results by considering the impact of different nuclear physics uncertainties such as different nuclear masses, $\u03b2^{-}$-decays and $\u03b2^{-}$-delayed neutron emission probabilities, neutrino reactions, fission, and a feedback of nuclear energy on the temperature. We find that the qualitative results do not change with different nuclear physics input. The properties of the explosion dynamics and the magnetic field configuration are the dominant factors determining the ejecta composition.",

"comments": " ",

"date": "25 January 2024",

"pdf_url": https://arxiv.org/pdf/2401.14402

}

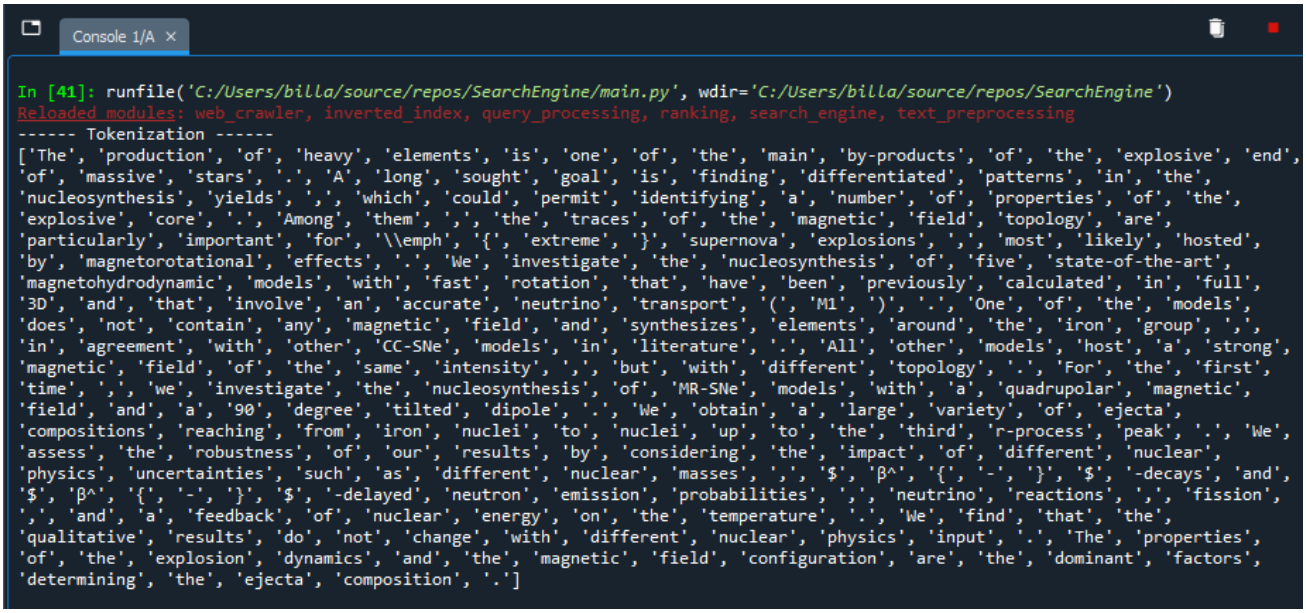


Figure 2.D.1 Tokenization

Figure 2.D.2 Punctuation characters removal

Figure 2.D.3 Special characters removal

Figure 2.D.4 Normalization

Figure 2.D.5 Stop-words removal

Figure 2.D.6 Stemming (instead of Lemmatization)

Figure 2.D.7 Lemmatization (instead of Stemming)

3. Indexing

3.a. Creating the inverted index data structure

3.a.1 Planning

The design of the inverted index data structure consists of the collection of tasks for indexing (step 1), the conversion of the textual content of each task into a list of elements (tokenization) as well as their preprocessing (step 2). Finally, it consists of the indexing of the works containing the terms, that is, it matches the terms (tokens) that have been collected with the specific texts in which they have been found. Virtually every word corresponds to a number which in turn shows the text in which the word is found. For example, if a word corresponding to the numbers 1,3,7 we know that it is in the texts with doc_id 1,3,7.

3.a.2 Implementation

To implement the inverted index data structure it was necessary to initialize an empty inverted index, then to separate the words from the summaries for each summation separately into word units (tokenization). All preprocessed words are then added to the index, and if a word already exists in it, the id of the task in which the word is found is simply added to the index keyword, thus avoiding duplicate concepts within the index. Finally, the keys of the dictionary and its elements are sorted.

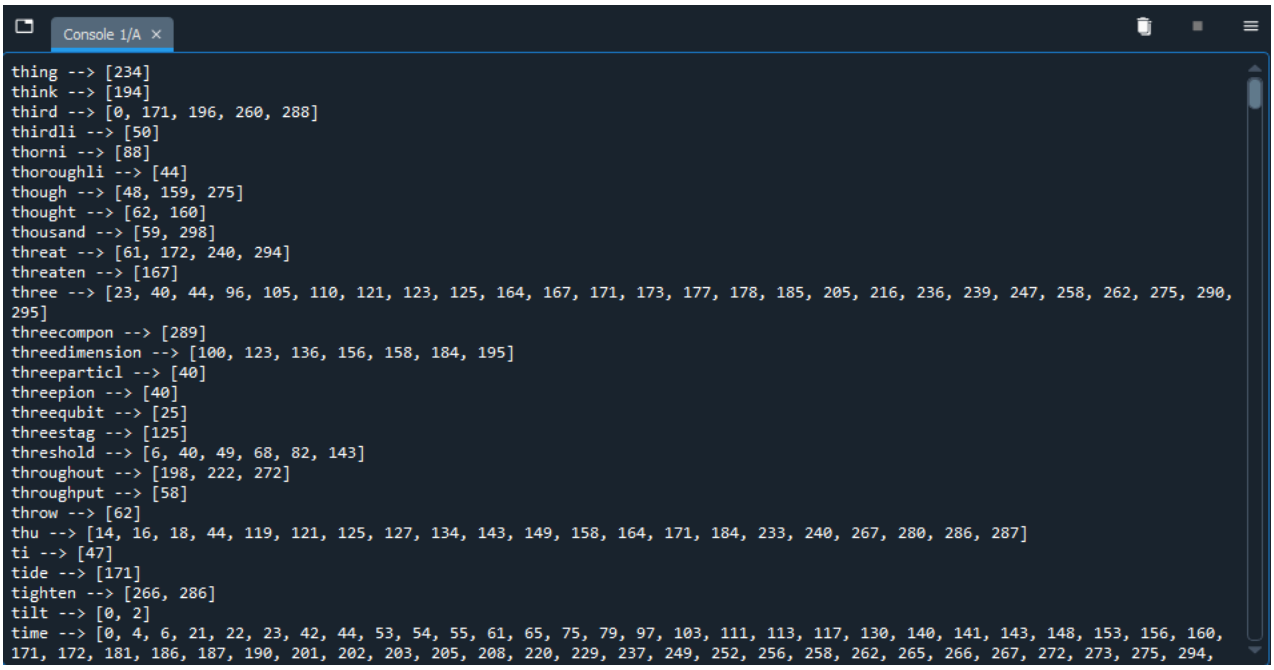


Figure 3.A.1 The inverted index data structure

3.a.3 Evaluation

The inverted index data structure greatly improves search terms in a text and is therefore a powerful tool for search engines. Matching the terms to the texts in which they appear creates an important picture of the frequency of the terms in the collection with the papers, which reinforces how important information a term is to the collection. The optimal indexing in terms of efficiency would be to list the position(s) of the term within the text, so that searches are more efficient and accurate, as in this way more information is provided to the user. However, this methodology was not chosen when implementing the local search engine index, as the repository is small and does not need such an implementation that spends memory space and search time due to its large size.

3.b. Store the index in a data structure

3.b.1 Planning

The creation of the inverted index data structure has been completed in step 3.a , so what remains is to store it in a Python language data structure that will be efficient in terms of searching. The design model boils down to the choice of dictionary structure to store the index.

3.b.2 Implementation

The structure of the dictionary includes two fields, the keys (keys) and the contents (values). In the local search engine implementation, keywords are all the unique terms of the collection of tasks (terms), while contents are lists (posting lists) with the unique integers (doc_id) that denote the texts in which the term appears-key.

3.b.3 Evaluation

The choice of the dictionary structure for storing the index is evaluated by the fact that it yields search efficiency by matching key terms to the lists of texts in which they appear. However, there may be significant memory consumption due to the large index size resulting from the number of unique keywords. However, the requirements of the local search engine are more about the accuracy and efficiency of the search than about saving space. By choosing another structure (eg table, list, tuple, etc.) there would be a significant improvement in space, but there would be a significant impact on the search result. In closing, the repository is small so memory consumption would not be so much that it would cause a significant problem in search engine performance.

4. Search Engine

4.a. Develop user interface for job search

4.a.1 Planning

In order to make the search engine more user friendly a user interface (GUI) was designed to search the tasks. More specifically, this interface contains:

- User interface window
- User query input field
- Retrieval Algorithm Selection Field (Boolean Retrieval, Vector Space Model, Probabilistic Retrieval Model)
- Search recovery algorithm results button
- Filter criteria input field
- Filter selection field

- Search filter results button

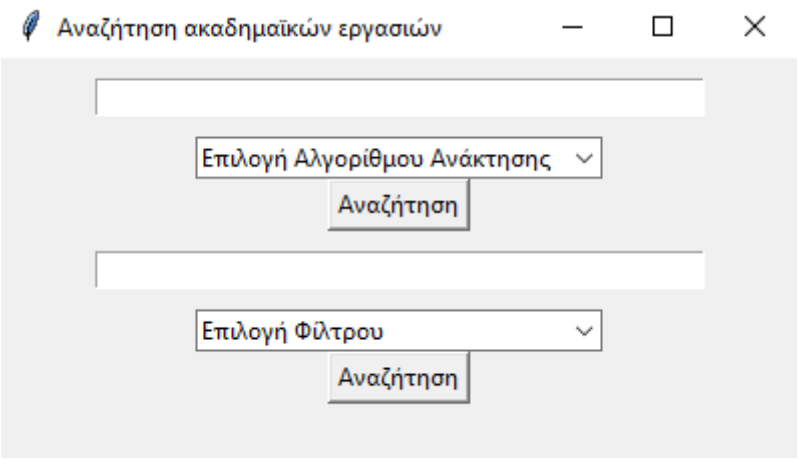


Figure 4.A.1 The user interface

4.a.2 Implementation

The implementation of the user interface was defined in a separate module (search_engine.py) and specifically the tkinter library was used. The interface serves the two basic functions of searching for works based on a user query and filtering results based on specific criteria, such as author or publication date. The interface and task search functionality are implemented in a class with properties the repository with preprocessed fields "abstract" (step 2) and the inverted index (step 3).

The implementations of the interface components mentioned in the design are as follows:

- User query input field: The user enters the query
- Retrieval Algorithm Selection Field: The user selects from a menu one of the 3 retrieval algorithms (Boolean Retrieval, Vector Space Model, Probabilistic Retrieval Model) to be applied to retrieve tasks
- Search Retrieval Algorithm Results Button: Clicking the Search button displays the query results in the terminal window, based on the retrieval algorithm selected by the user
- Filter criterion selection field: The user selects from a menu (if desired) one of the 2 result filtering criteria (Author, Publication date)
- Filter criteria input field: User enters the filter (author name or publication date)
- Filter Results Search Button: Pressing the "Search" button displays the filtering results in the terminal window, based on the filter criteria selected by the user

4.a.3 Evaluation

The interface, although simple, meets the criteria so that the user can easily search for jobs based on certain words that he sets as a search query. It just as easily allows filtering based on specific criteria (published date, author), so there's nothing else in the interface that isn't there as a user-facing function. The interface takes care of the correct entry of data into the search engine, data processing and possible filtering. The usefulness of this class is catalytic, since closing the interface also terminates the program. Perhaps, it would be preferable

not to have so many functions concentrated in the interface class purely as a matter of code flexibility and readability .

4.b. Implementation of recovery algorithms

4.b.1 Planning

Boolean Retrieval

The Boolean Retrieval algorithm implemented in the search engine is based on Boolean algebra and set theory, using three basic keywords: OR, AND and NOT. It follows the process of Query Processing . These functions affect the search results according to their position in the query.

- The keyword NOT, when placed before a word, searches for texts that do not contain that word.
- The AND keyword, when placed between two words, searches for texts that contain both words.
- The OR keyword, when placed between two words, searches for text that contains either word.

In addition, the algorithm takes into account the priority of operations in complex Boolean user queries, performing the operations in this order of priority:

1. Operations with the logical NOT operators in parentheses (from left to right)
2. Operations with the AND or OR operators inside parentheses (from left to right)
3. Operations with the logical NOT operators (from left to right)
4. Operations with the AND or OR operators (from left to right)

In this way, the algorithm can be adapted to respond to complex search queries.

Vector Space Model

The Vector Space Model algorithm implemented in the search engine represents texts and user queries as vectors in an n-dimensional space. From the ranking algorithms it follows TF-IDF and Cosine Similarity to calculate the similarity between the user query and the collection texts. Due to the problem with large Euclidean distances of vectors, calculating the cosine of the angle that the query vector forms with the vector of some text in the collection is more efficient. The closer the vectors are, the closer the query is to the text, that is, the cosine of the angle tends to 1, where 1 means identical.

Probabilistic Retrieval Model

The Probabilistic Retrieval Model algorithm implemented in the search engine searches for jobs based on the probabilistic model. It essentially calculates the probability that a text is relevant to the user query. From the ranking algorithms it follows TF-IDF and Okapi BM25 to calculate the probabilities.

4.b.2 Implementation

Boolean Retrieval

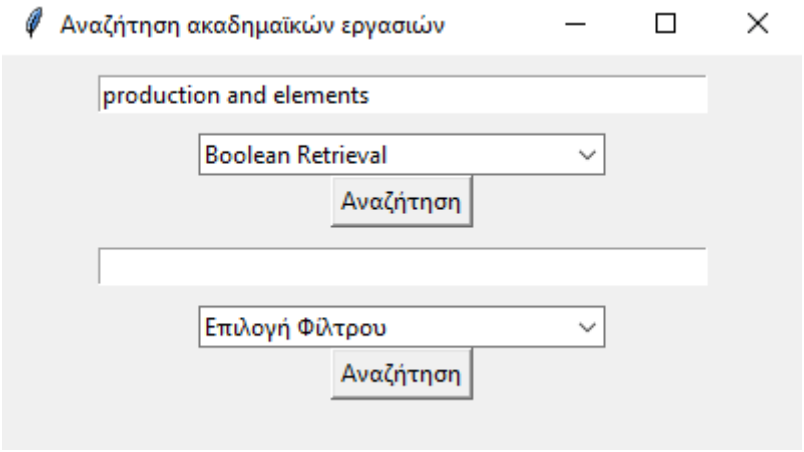
The implementation of Boolean Retrieval was carried out in the same module (search_engine.py) as the user interface and specifically in the same class. As input, the algorithm takes the following data:

- The Boolean query that the user will provide to perform the search
- The inverted index that will do the necessary matching between words and texts

The function of the algorithm is to create a stack of simple Boolean sub-queries of the complex or simple Boolean user query. The position of each sub-query in the stack is determined by the priority order of the operations mentioned in the design . For each sub-query, the operation from the query processing process is performed . The first concern of the algorithm is whether there are parentheses. In the case of parentheses, the algorithm allocates the logical operations within the parentheses to the stack first, and then any remaining operations. If there are no parentheses in the user's query, the search is performed normally from left to right.

Each word in the query must be properly examined to check if it corresponds to a keyword (and, or, not) or a word found within the index. For this reason, the terms of the query must become tokens, through which it will be possible to identify possible keywords that will allow the correct matching between words of the query and words found in the index.

Some illustrative case studies from user queries are as follows:

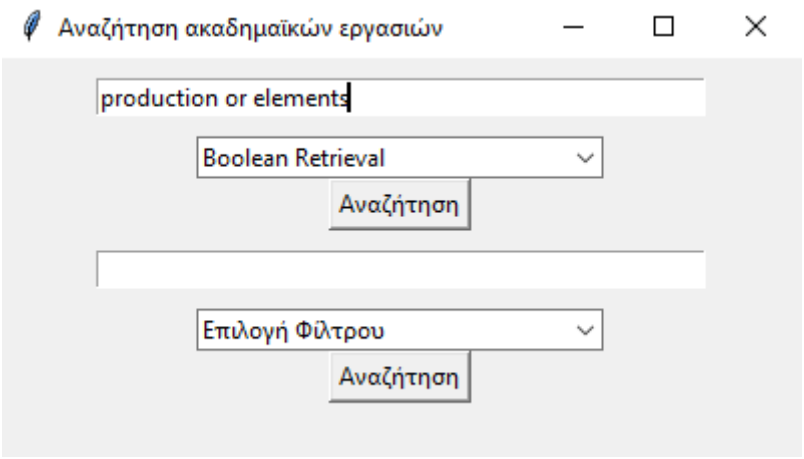


```
In [17]: runfile('C:/Users/billa/source/repos/SearchEngine/main.py', wdir='C:/Users/billa/source/repos/SearchEngine')
Reloaded modules: web_crawler, text_preprocessing, inverted_index, query_processing, ranking, search_engine


============ Ερώτημα αναζήτησης   : production and elements ============
============ Αλγόριθμος ανάκτησης : Boolean Retrieval ============
--------------------------------------------------
#1
--------------------------------------------------
Document ID : 0
Title       : Nucleosynthesis in magnetorotational supernovae: impact of the magnetic field configuration
Authors     : M. Reichert, M. Bugli, J. Guilet, M. Obergaulinger, M. Á. Aloy, A. Arcones
Subjects    : High Energy Astrophysical Phenomena
Abstract    : The production of heavy elements is one of the main by-products of the explosive end of massive stars. A long
sought goal is finding differentiated patterns in the nucleosynthesis yields, which could permit identifying a number of
properties of the explosive core. Among them, the traces of the magnetic field topology are particularly important for
\emph{extreme} supernova explosions, most likely hosted by magnetorotational effects. We investigate the nucleosynthesis of
five state-of-the-art magnetohydrodynamic models with fast rotation that have been previously calculated in full 3D and that
involve an accurate neutrino transport (M1). One of the models does not contain any magnetic field and synthesizes elements
around the iron group, in agreement with other CC-SNe models in literature. All other models host a strong magnetic field of
the same intensity, but with different topology. For the first time, we investigate the nucleosynthesis of MR-SNe models with a
quadrupolar magnetic field and a 90 degree tilted dipole. We obtain a large variety of ejecta compositions reaching from iron
nuclei to nuclei up to the third r-process peak. We assess the robustness of our results by considering the impact of different
nuclear physics uncertainties such as different nuclear masses, $β^{-}$-decays and $β^{-}$-delayed neutron emission
probabilities, neutrino reactions, fission, and a feedback of nuclear energy on the temperature. We find that the qualitative
results do not change with different nuclear physics input. The properties of the explosion dynamics and the magnetic field
configuration are the dominant factors determining the ejecta composition.
Comments    :
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14402
```

Figure 4.B.1 Simple Boolean query with AND

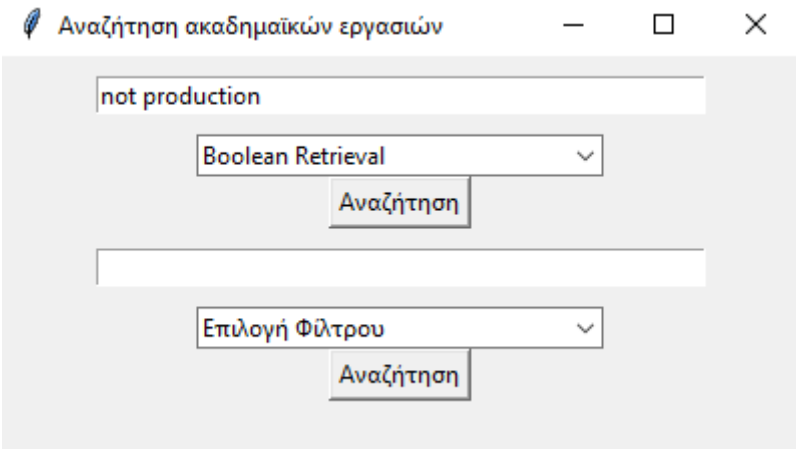
RECOVERY OF INFORMATION




```
============ Ερώτημα αναζήτησης   : production or elements ============
============ Αλγόριθμος ανάκτησης : Boolean Retrieval ============
--------------------------------------------------
#1
--------------------------------------------------
Document ID : 0
Title       : Nucleosynthesis in magnetorotational supernovae: impact of the magnetic field configuration
Authors     : M. Reichert, M. Bugli, J. Guilet, M. Obergaulinger, M. Á. Aloy, A. Arcones
Subjects    : High Energy Astrophysical Phenomena
Abstract    : The production of heavy elements is one of the main by-products of the explosive end of massive stars. A long
sought goal is finding differentiated patterns in the nucleosynthesis yields, which could permit identifying a number of
properties of the explosive core. Among them, the traces of the magnetic field topology are particularly important for
\emph{extreme} supernova explosions, most likely hosted by magnetorotational effects. We investigate the nucleosynthesis of
five state-of-the-art magnetohydrodynamic models with fast rotation that have been previously calculated in full 3D and that
involve an accurate neutrino transport (M1). One of the models does not contain any magnetic field and synthesizes elements
around the iron group, in agreement with other CC-SNe models in literature. All other models host a strong magnetic field of
the same intensity, but with different topology. For the first time, we investigate the nucleosynthesis of MR-SNe models with a
quadrupolar magnetic field and a 90 degree tilted dipole. We obtain a large variety of ejecta compositions reaching from iron
nuclei to nuclei up to the third r-process peak. We assess the robustness of our results by considering the impact of different
nuclear physics uncertainties such as different nuclear masses, $β^{-}$-decays and $β^{-}$-delayed neutron emission
probabilities, neutrino reactions, fission, and a feedback of nuclear energy on the temperature. We find that the qualitative
results do not change with different nuclear physics input. The properties of the explosion dynamics and the magnetic field
configuration are the dominant factors determining the ejecta composition.
Comments    :
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14402
--------------------------------------------------
#2
--------------------------------------------------
Document ID : 42
Title       : How far can we see back in time in high-energy collisions using charm quarks?
Authors     : Laszlo Gyulai, Gabor Biro, Robert Vertesi, Gergely Gabor Barnafoldi
Subjects    : High Energy Physics - Phenomenology, Nuclear Theory
Abstract    : We use open charm production to estimate how far we can see back in time in high-energy hadron-hadron collisions.
We analyze the transverse momentum distributions of the identified D mesons from pp, p-Pb and A-A collisions at the ALICE and
STAR experiments covering the energy range from $\sqrt{s_{\rm NN}} = 200$ GeV up to 7 TeV. Within a non-extensive statistical
framework, the common Tsallis parameters for D mesons represent higher temperature and more degrees of freedom than that of
light-flavour hadrons. The production of D mesons corresponds to a significantly earlier proper time, $τ_{\rm D} = (0.18 \pm
0.06) τ_{\rm LF}$.
Comments    : 18 pages, 6 figures, 1 table
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14282
--------------------------------------------------
#3
--------------------------------------------------
Document ID : 45
Title       : Phenomenology of TMD parton distributions in Drell-Yan and $Z^0$ boson production in a hadron structure oriented
```

```
taken from the 2014 Global Adult Population Survey of the Global Entrepreneurship Monitor project. The propensity for
innovation amongst tourism entrepreneurs has a statistically significant relationship to gender, age, level of education and
informal investments in previous businesses.
Comments    : Journal ref:        Sustainability 12:5003 (2020)
Date        : 5 December 2023
PDF_URL     : https://arxiv.org/pdf/2401.13679
--------------------------------------------------
#19
--------------------------------------------------
Document ID : 236
Title       : Realized Stochastic Volatility Model with Skew-t Distributions for Improved Volatility and Quantile Forecasting
Authors     : Makoto Takahashi, Yuta Yamauchi, Toshiaki Watanabe, Yasuhiro Omori
Subjects    : Econometrics
Abstract    : Forecasting volatility and quantiles of financial returns is essential for accurately measuring financial tail
risks, such as value-at-risk and expected shortfall. The critical elements in these forecasts involve understanding the
distribution of financial returns and accurately estimating volatility. This paper introduces an advancement to the traditional
stochastic volatility model, termed the realized stochastic volatility model, which integrates realized volatility as a precise
estimator of volatility. To capture the well-known characteristics of return distribution, namely skewness and heavy tails, we
incorporate three types of skew-t distributions. Among these, two distributions include the skew-normal feature, offering
enhanced flexibility in modeling the return distribution. We employ a Bayesian estimation approach using the Markov chain Monte
Carlo method and apply it to major stock indices. Our empirical analysis, utilizing data from US and Japanese stock indices,
indicates that the inclusion of both skewness and heavy tails in daily returns significantly improves the accuracy of
volatility and quantile forecasts.
Comments    :
Date        : 23 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.13179
--------------------------------------------------
#20
--------------------------------------------------
Document ID : 248
Title       : Optimal design of a local renewable electricity supply system for power-intensive production processes with
demand response
Authors     : Sonja H. M. Germscheid, Benedikt Nilges, Niklas von der Assen, Alexander Mitsos, Manuel Dahmen
Subjects    : Optimization and Control
Abstract    : This work studies synergies arising from combining industrial demand response and local renewable electricity
supply. To this end, we optimize the design of a local electricity generation and storage system with an integrated demand
response scheduling of a continuous power-intensive production process in a multi-stage problem. We optimize both total
annualized cost and global warming impact and consider local photovoltaic and wind electricity generation, an electric battery,
and electricity trading on day-ahead and intraday market. We find that installing a battery can reduce emissions and enable
large trading volumes on the electricity markets, but significantly increases cost. Economic and ecologic process and battery
operation are driven primarily by the electricity price and grid emission factor, respectively, rather than locally generated
electricity. A parameter study reveals that economic savings from the local system and flexibilizing the process behave almost
additive.
Comments    : manuscript (32 pages, 9 figures, 6 tables), supporting materials (11 pages, 9 figures, 2 tables)
Date        : 23 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.12759
```

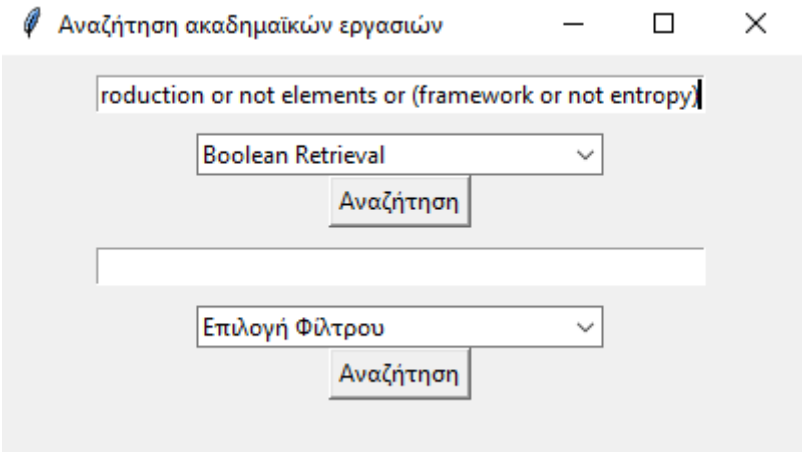
Figure 4.B.2 Simple Boolean query with OR



```
============ Ερώτημα αναζήτησης   : not production ============
============ Αλγόριθμος ανάκτησης : Boolean Retrieval ============
--------------------------------------------------
#1
--------------------------------------------------
Document ID : 1
Title       : pix2gestalt: Amodal Segmentation by Synthesizing Wholes
Authors     : Ege Ozguroglu, Ruoshi Liu, Dídac Surís, Dian Chen, Achal Dave, Pavel Tokmakov, Carl Vondrick
Subjects    : Computer Vision and Pattern Recognition, Machine Learning
Abstract    : We introduce pix2gestalt, a framework for zero-shot amodal segmentation, which learns to estimate the shape and
appearance of whole objects that are only partially visible behind occlusions. By capitalizing on large-scale diffusion models
and transferring their representations to this task, we learn a conditional diffusion model for reconstructing whole objects in
challenging zero-shot cases, including examples that break natural and physical priors, such as art. As training data, we use a
synthetically curated dataset containing occluded objects paired with their whole counterparts. Experiments show that our
approach outperforms supervised baselines on established benchmarks. Our model can furthermore be used to significantly improve
the performance of existing object recognition and 3D reconstruction methods in the presence of occlusions.
Comments    : Website: https://gestalt.cs.columbia.edu/
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14398
--------------------------------------------------
#2
--------------------------------------------------
Document ID : 2
Title       : Entanglement entropy and deconfined criticality: emergent SO(5) symmetry and proper lattice bipartition
Authors     : Jonathan D'Emidio, Anders W. Sandvik
Subjects    : Strongly Correlated Electrons, High Energy Physics - Lattice
Abstract    : We study the Rényi entanglement entropy (EE) of the two-dimensional $J$-$Q$ model, the emblematic quantum spin
model of deconfined criticality at the phase transition between antiferromagnetic and valence-bond-solid ground states. Quantum
Monte Carlo simulations with an improved EE scheme reveal critical corner contributions that scale logarithmically with the
system size, with a coefficient in remarkable agreement with the form expected from a large-$N$ conformal field theory with
SO($N=5$) symmetry. However, details of the bipartition of the lattice are crucial in order to observe this behavior. If the
subsystem for the reduced density matrix does not properly accommodate valence-bond fluctuations, logarithmic contributions
appear even for corner-less bipartitions. We here use a $45^\circ$ tilted cut on the square lattice. Beyond supporting an
SO($5$) deconfined quantum critical point, our results for both the regular and tilted cuts demonstrate important microscopic
aspects of the EE that are not captured by conformal field theory.
Comments    : 5 pages, 3 figures
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14396
--------------------------------------------------
#3
--------------------------------------------------
Document ID : 3
Title       : Summing up perturbation series around superintegrable point
Authors     : A. Mironov, A. Morozov, A. Popolitov, Sh. Shakirov
Subjects    : High Energy Physics - Theory, Mathematical Physics
Abstract    : We work out explicit formulas for correlators in the Gaussian matrix model perturbed by a logarithmic potential,
i.e. by inserting Miwa variables. In this paper, we concentrate on the example of a single Miwa variable. The ordinary Gaussian
```

```
--------------------------------------------------
#19
--------------------------------------------------
Document ID : 19
Title       : Uncovering Heterogeneity of Solar Flare Mechanism With Mixture Models
Authors     : Bach Viet Do, Yang Chen, XuanLong Nguyen, Ward Manchester
Subjects    : Solar and Stellar Astrophysics, Applications, Methodology
Abstract    : The physics of solar flares occurring on the Sun is highly complex and far from fully understood. However,
observations show that solar eruptions are associated with the intense kilogauss fields of active regions, where free energies
are stored with field-aligned electric currents. With the advent of high-quality data sources such as the Geostationary
Operational Environmental Satellites (GOES) and Solar Dynamics Observatory (SDO)/Helioseismic and Magnetic Imager (HMI), recent
works on solar flare forecasting have been focusing on data-driven methods. In particular, black box machine learning and deep
learning models are increasingly adopted in which underlying data structures are not modeled explicitly. If the active regions
indeed follow the same laws of physics, there should be similar patterns shared among them, reflected by the observations. Yet,
these black box models currently used in the literature do not explicitly characterize the heterogeneous nature of the solar
flare data, within and between active regions. In this paper, we propose two finite mixture models designed to capture the
heterogeneous patterns of active regions and their associated solar flare events. With extensive numerical studies, we
demonstrate the usefulness of our proposed method for both resolving the sample imbalance issue and modeling the heterogeneity
for rare energetic solar flare events.
Comments    :
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14345
--------------------------------------------------
#20
--------------------------------------------------
Document ID : 20
Title       : From the Choi Formalism in Infinite Dimensions to Unique Decompositions of Generators of Completely Positive
Dynamical Semigroups
Authors     : Frederik vom Ende
Subjects    : Functional Analysis, Mathematical Physics, Quantum Physics
Abstract    : Given any separable complex Hilbert space, any trace-class operator $B$ which does not have purely imaginary
trace, and any generator $L$ of a norm-continuous one-parameter semigroup of completely positive maps we prove that there
exists a unique bounded operator $K$ and a unique completely positive map $\Phi$ such that (i) $L=K(\cdot)+(\cdot)K^*+\Phi$, (ii) the
superoperator $\Phi(B^*(\cdot)B)$ is trace class and has vanishing trace, and (iii) ${\rm tr}(B^*K)$ is a real number. Central to
our proof is a modified version of the Choi formalism which relates completely positive maps to positive semi-definite
operators. We characterize when this correspondence is injective and surjective, respectively, which in turn explains why the
proof idea of our main result cannot extend to non-separable Hilbert spaces. In particular, we find examples of positive semi-
definite operators which have empty pre-image under the Choi formalism as soon as the underlying Hilbert space is infinite-
dimensional.
Comments    : 25+3 pages. Generalizes arXiv:2310.04037 to infinite dimensions. To be submitted to J. Funct. Anal
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14344
```

Figure 4.B.3 Simple Boolean query with NOT




```
============ Ερώτημα αναζήτησης   : production or not elements or (framework or not entropy) ============
============ Αλγόριθμος ανάκτησης : Boolean Retrieval ============
--------------------------------------------------
#1
--------------------------------------------------
Document ID : 0
Title       : Nucleosynthesis in magnetorotational supernovae: impact of the magnetic field configuration
Authors     : M. Reichert, M. Bugli, J. Guilet, M. Obergaulinger, M. Á. Aloy, A. Arcones
Subjects    : High Energy Astrophysical Phenomena
Abstract    : The production of heavy elements is one of the main by-products of the explosive end of massive stars. A long
sought goal is finding differentiated patterns in the nucleosynthesis yields, which could permit identifying a number of
properties of the explosive core. Among them, the traces of the magnetic field topology are particularly important for
\emph{extreme} supernova explosions, most likely hosted by magnetorotational effects. We investigate the nucleosynthesis of
five state-of-the-art magnetohydrodynamic models with fast rotation that have been previously calculated in full 3D and that
involve an accurate neutrino transport (M1). One of the models does not contain any magnetic field and synthesizes elements
around the iron group, in agreement with other CC-SNe models in literature. All other models host a strong magnetic field of
the same intensity, but with different topology. For the first time, we investigate the nucleosynthesis of MR-SNe models with a
quadrupolar magnetic field and a 90 degree tilted dipole. We obtain a large variety of ejecta compositions reaching from iron
nuclei to nuclei up to the third r-process peak. We assess the robustness of our results by considering the impact of different
nuclear physics uncertainties such as different nuclear masses, $β^{-}$-decays and $β^{-}$-delayed neutron emission
probabilities, neutrino reactions, fission, and a feedback of nuclear energy on the temperature. We find that the qualitative
results do not change with different nuclear physics input. The properties of the explosion dynamics and the magnetic field
configuration are the dominant factors determining the ejecta composition.
Comments    :
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14402
--------------------------------------------------
#2
--------------------------------------------------
Document ID : 1
Title       : pix2gestalt: Amodal Segmentation by Synthesizing Wholes
Authors     : Ege Ozguroglu, Ruoshi Liu, Dídac Surís, Dian Chen, Achal Dave, Pavel Tokmakov, Carl Vondrick
Subjects    : Computer Vision and Pattern Recognition, Machine Learning
Abstract    : We introduce pix2gestalt, a framework for zero-shot amodal segmentation, which learns to estimate the shape and
appearance of whole objects that are only partially visible behind occlusions. By capitalizing on large-scale diffusion models
and transferring their representations to this task, we learn a conditional diffusion model for reconstructing whole objects in
challenging zero-shot cases, including examples that break natural and physical priors, such as art. As training data, we use a
synthetically curated dataset containing occluded objects paired with their whole counterparts. Experiments show that our
approach outperforms supervised baselines on established benchmarks. Our model can furthermore be used to significantly improve
the performance of existing object recognition and 3D reconstruction methods in the presence of occlusions.
Comments    : Website: https://gestalt.cs.columbia.edu/
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14398
--------------------------------------------------
#3
--------------------------------------------------
Document ID : 2
Title       : Entanglement entropy and deconfined criticality: emergent SO(5) symmetry and proper lattice bipartition
```

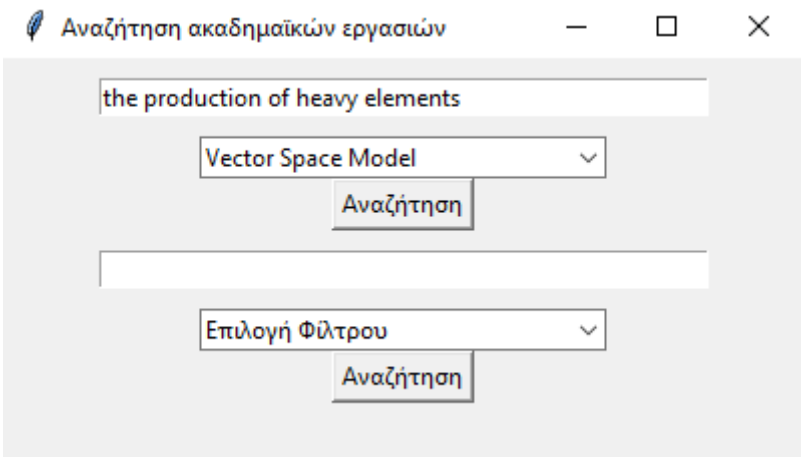
Figure 4.B.4 Complex Boolean query

Vector Space Model

The implementation of the Vector Space Model was carried out in the same module (search_engine.py) as the user interface and specifically in the same class. As input, the algorithm takes the user query and the implementations that take place are as follows:

- Text Preprocessing: Step 2
- TF-IDF calculation of text terms: Ranking of results (Ranking)
- TF-IDF calculation of query terms: Ranking of results (Ranking)
- Calculation of the cosines between the texts and the query: Ranking of results (Ranking)
- Ranking of results (Ranking)

Some illustrative case studies from user queries are as follows:




```
Console 1/A ×
============ Ερώτημα αναζήτησης   : the production of heavy elements ============
============ Αλγόριθμος ανάκτησης : Vector Space Model ============
--------------------------------------------------
#1 Cosine Similarity: 0.1566
--------------------------------------------------
Document ID : 0
Title       : Nucleosynthesis in magnetorotational supernovae: impact of the magnetic field configuration
Authors     : M. Reichert, M. Bugli, J. Guilet, M. Obergaulinger, M. Á. Aloy, A. Arcones
Subjects    : High Energy Astrophysical Phenomena
Abstract    : The production of heavy elements is one of the main by-products of the explosive end of massive stars. A long
sought goal is finding differentiated patterns in the nucleosynthesis yields, which could permit identifying a number of
properties of the explosive core. Among them, the traces of the magnetic field topology are particularly important for
\emph{extreme} supernova explosions, most likely hosted by magnetorotational effects. We investigate the nucleosynthesis of
five state-of-the-art magnetohydrodynamic models with fast rotation that have been previously calculated in full 3D and that
involve an accurate neutrino transport (M1). One of the models does not contain any magnetic field and synthesizes elements
around the iron group, in agreement with other CC-SNe models in literature. All other models host a strong magnetic field of
the same intensity, but with different topology. For the first time, we investigate the nucleosynthesis of MR-SNe models with
a quadrupolar magnetic field and a 90 degree tilted dipole. We obtain a large variety of ejecta compositions reaching from
iron nuclei to nuclei up to the third r-process peak. We assess the robustness of our results by considering the impact of
different nuclear physics uncertainties such as different nuclear masses, $β^{-}$-decays and $β^{-}$-delayed neutron emission
probabilities, neutrino reactions, fission, and a feedback of nuclear energy on the temperature. We find that the qualitative
results do not change with different nuclear physics input. The properties of the explosion dynamics and the magnetic field
configuration are the dominant factors determining the ejecta composition.
Comments    :
Date        : 25 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.14402
--------------------------------------------------
#2 Cosine Similarity: 0.1539
--------------------------------------------------
Document ID : 236
Title       : Realized Stochastic Volatility Model with Skew-t Distributions for Improved Volatility and Quantile Forecasting
Authors     : Makoto Takahashi, Yuta Yamauchi, Toshiaki Watanabe, Yasuhiro Omori
Subjects    : Econometrics
Abstract    : Forecasting volatility and quantiles of financial returns is essential for accurately measuring financial tail
risks, such as value-at-risk and expected shortfall. The critical elements in these forecasts involve understanding the
distribution of financial returns and accurately estimating volatility. This paper introduces an advancement to the
traditional stochastic volatility model, termed the realized stochastic volatility model, which integrates realized volatility
as a precise estimator of volatility. To capture the well-known characteristics of return distribution, namely skewness and
heavy tails, we incorporate three types of skew-t distributions. Among these, two distributions include the skew-normal
feature, offering enhanced flexibility in modeling the return distribution. We employ a Bayesian estimation approach using the
Markov chain Monte Carlo method and apply it to major stock indices. Our empirical analysis, utilizing data from US and
Japanese stock indices, indicates that the inclusion of both skewness and heavy tails in daily returns significantly improves
the accuracy of volatility and quantile forecasts.
Comments    :
Date        : 23 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.13179
--------------------------------------------------
#3 Cosine Similarity: 0.0992
```

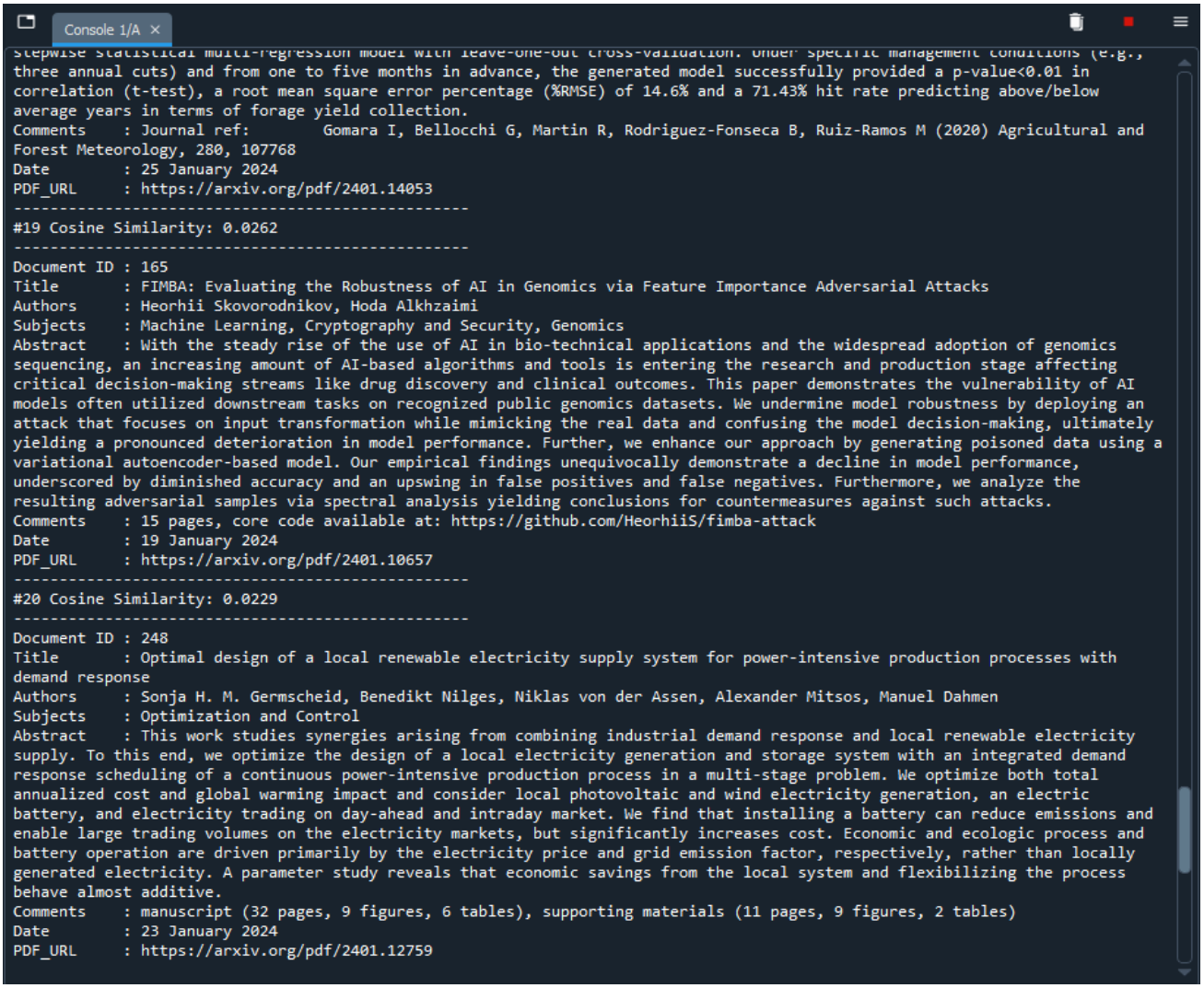
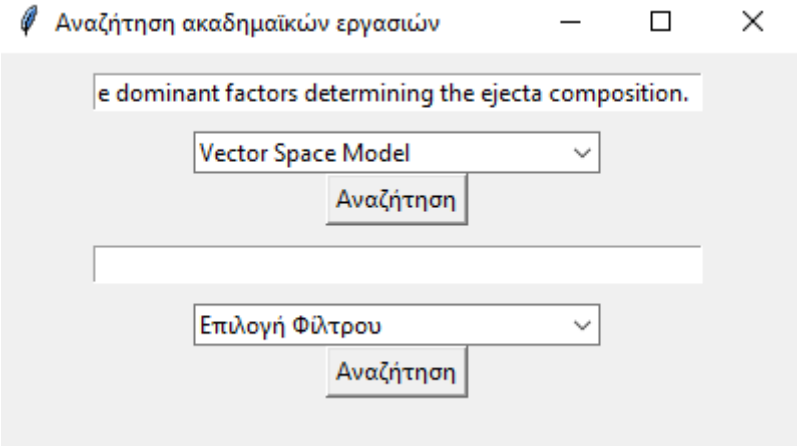


Figure 4.B.5 Small User Query (VSM)



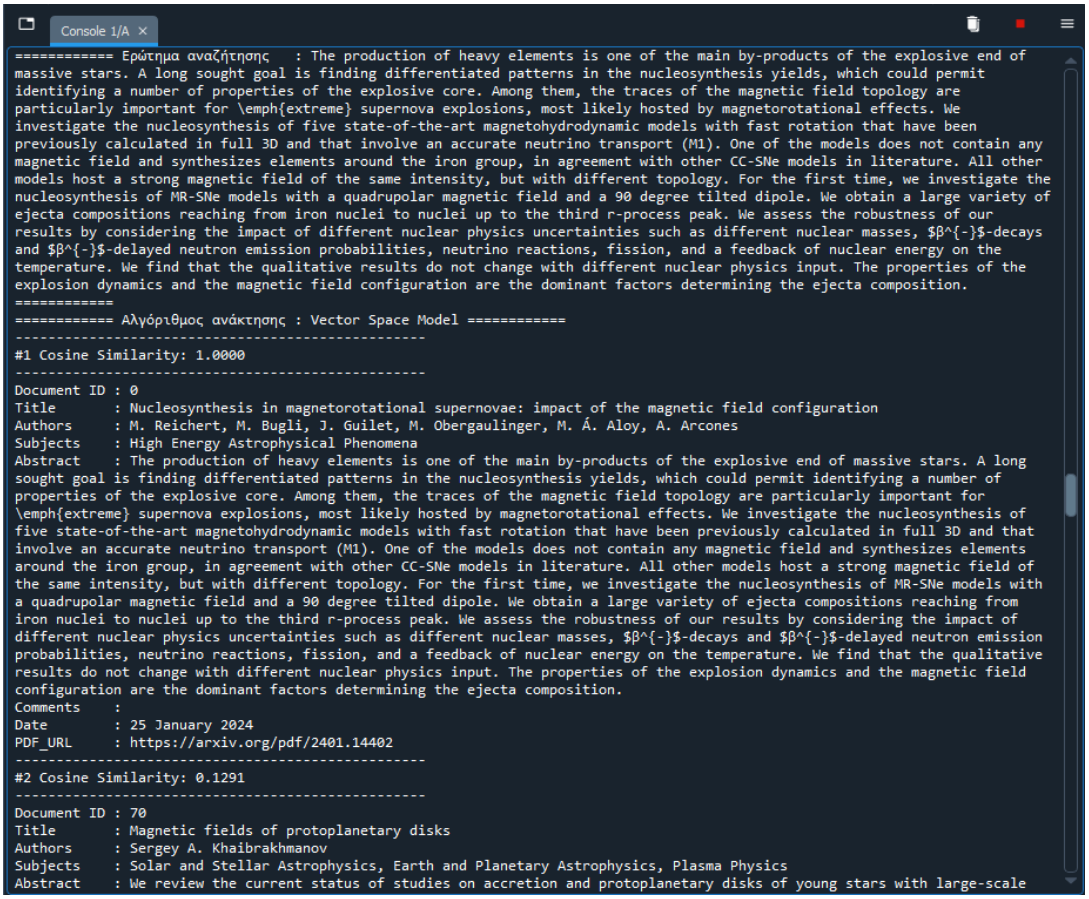


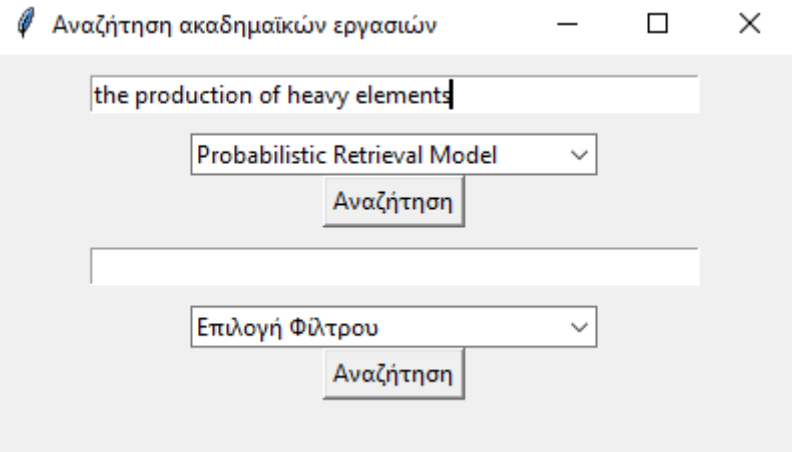
Figure 4.B.6 Identical user query with some text (VSM)

Probabilistic Retrieval Model

The implementation of the Probabilistic Retrieval Model was carried out in the same module (search_engine.py) as the user interface and specifically in the same class. As input, the algorithm takes the user query and the inverted index, and the implementations that take place are as follows:

- Text Preprocessing: Step 2
- TF-IDF calculation of text terms: Ranking of results (Ranking)
- TF-IDF calculation of query terms: Ranking of results (Ranking)
- Calculation of the Okapi BM25 score between the texts and the query: Ranking of results (Ranking)
- Ranking of results (Ranking)

Some illustrative case studies from user queries are as follows:



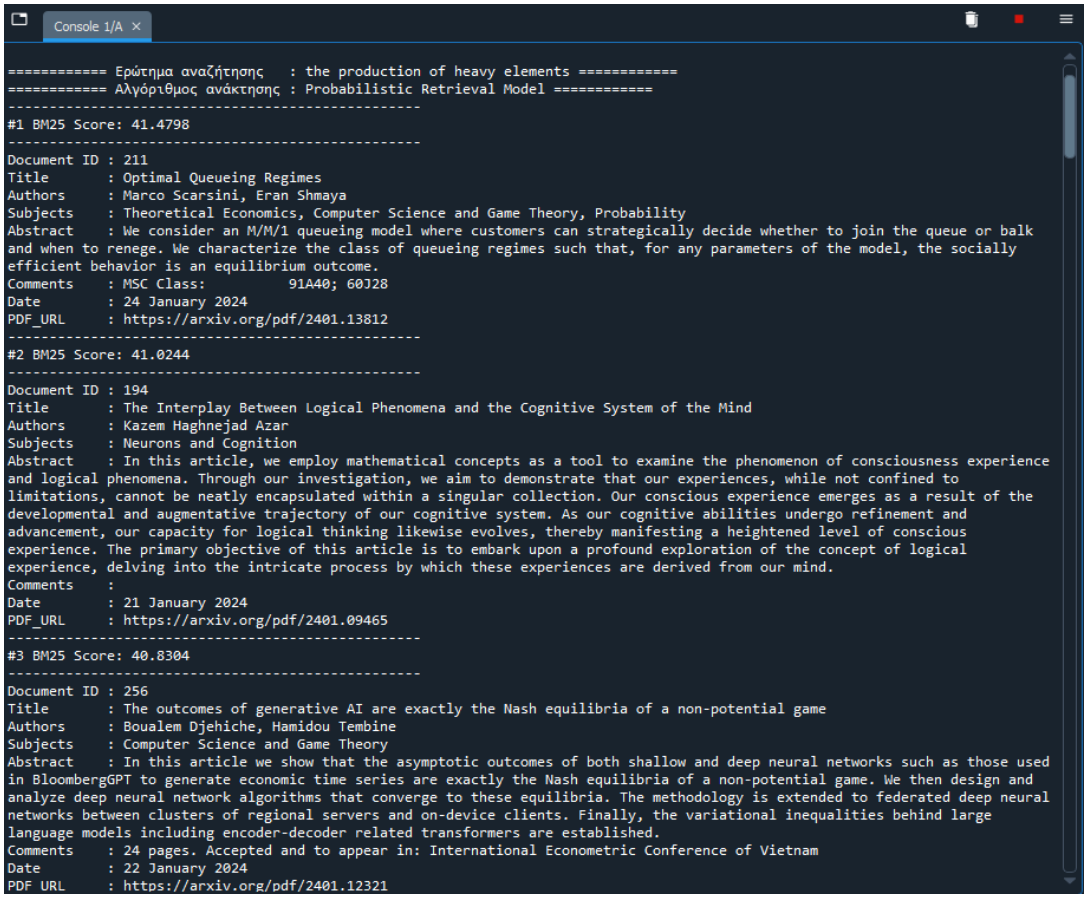
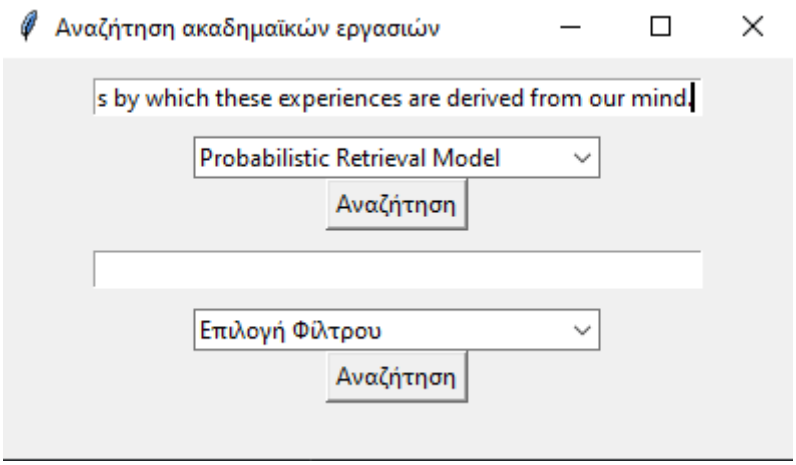


Figure 4.B.7 Small User Query (PRM)

RECOVERY OF INFORMATION




```
Console 1/A

============ Ερώτημα αναζήτησης   : In this article, we employ mathematical concepts as a tool to examine the phenomenon of
consciousness experience and logical phenomena. Through our investigation, we aim to demonstrate that our experiences, while
not confined to limitations, cannot be neatly encapsulated within a singular collection. Our conscious experience emerges as a
result of the developmental and augmentative trajectory of our cognitive system. As our cognitive abilities undergo refinement
and advancement, our capacity for logical thinking likewise evolves, thereby manifesting a heightened level of conscious
experience. The primary objective of this article is to embark upon a profound exploration of the concept of logical
experience, delving into the intricate process by which these experiences are derived from our mind. ============
============ Αλγόριθμος ανάκτησης : Probabilistic Retrieval Model ============
--------------------------------------------------
#1 BM25 Score: 719.5806
--------------------------------------------------
Document ID : 194
Title       : The Interplay Between Logical Phenomena and the Cognitive System of the Mind
Authors     : Kazem Haghnejad Azar
Subjects    : Neurons and Cognition
Abstract    : In this article, we employ mathematical concepts as a tool to examine the phenomenon of consciousness experience
and logical phenomena. Through our investigation, we aim to demonstrate that our experiences, while not confined to
limitations, cannot be neatly encapsulated within a singular collection. Our conscious experience emerges as a result of the
developmental and augmentative trajectory of our cognitive system. As our cognitive abilities undergo refinement and
advancement, our capacity for logical thinking likewise evolves, thereby manifesting a heightened level of conscious
experience. The primary objective of this article is to embark upon a profound exploration of the concept of logical
experience, delving into the intricate process by which these experiences are derived from our mind.
Comments    :
Date        : 21 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.09465
--------------------------------------------------
#2 BM25 Score: 711.1157
--------------------------------------------------
Document ID : 144
Title       : Accelerating Seed Location Filtering in DNA Read Mapping Using a Commercial Compute-in-SRAM Architecture
Authors     : Courtney Golden, Dan Ilan, Nicholas Cebry, Christopher Batten
Subjects    : Hardware Architecture, Genomics
Abstract    : DNA sequence alignment is an important workload in computational genomics. Reference-guided DNA assembly
involves aligning many read sequences against candidate locations in a long reference genome. To reduce the computational load
of this alignment, candidate locations can be pre-filtered using simpler alignment algorithms like edit distance. Prior work
has explored accelerating filtering on simulated compute-in-DRAM, due to the massive parallelism of compute-in-memory
architectures. In this paper, we present work-in-progress on accelerating filtering using a commercial compute-in-SRAM
accelerator. We leverage the recently released Gemini accelerator platform from GSI Technology, which is the first, to our
knowledge, commercial-scale compute-in-SRAM system. We accelerate the Myers' bit-parallel edit distance algorithm, producing
average speedups of 14.1x over single-core CPU performance. Individual query/candidate alignments produce speedups of up to
24.1x. These early results suggest this novel architecture is well-suited to accelerating the filtering step of sequence-to-
sequence DNA alignment.
Comments    : Journal ref:        5th Workshop on Accelerator Architecture in Computational Biology and Bioinformatics
(AACBB), June 2023
Date        : 21 January 2024
PDF_URL     : https://arxiv.org/pdf/2401.11685
```

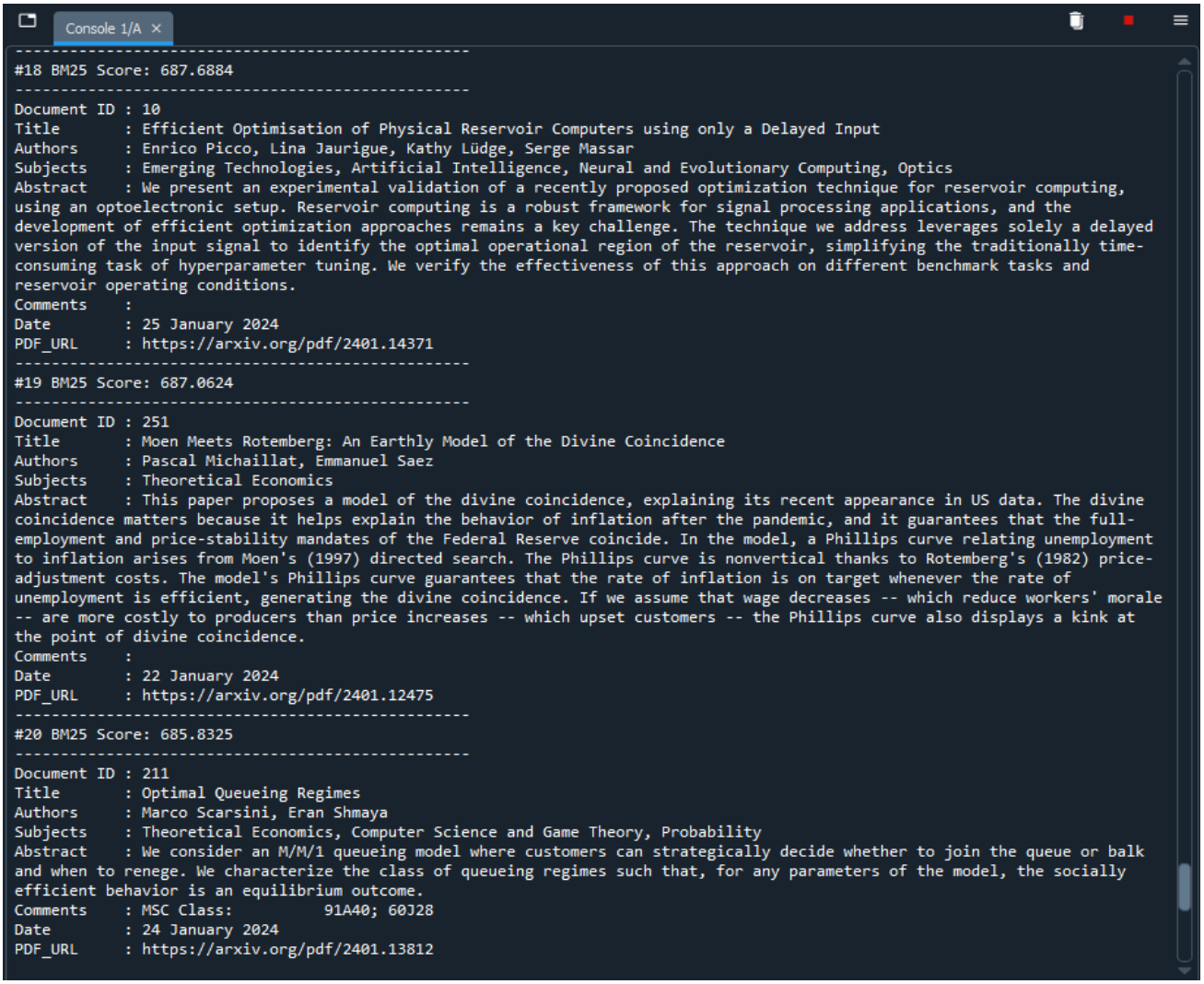


Figure 4.B.8 Identical user query with some text (PRM)

4.b.3 Evaluation

Boolean Retrieval

The Boolean Retrieval code is implemented in such a way that it can search with both simple and complex expressions that will be given to it by the user through a search query. However, the specific keywords necessary for the operation of the algorithm (and, or, not) must be spelled correctly, because whether the letters entered by the user are in lowercase or uppercase does not affect in search of something. In the event that a word given through a query by the user does not exist in the index, the code will handle the error as in other search engines if a word is not in the indexes, there are no relevant results.

Vector Space Model

The Vector Space Model code performs for the small search engine repository and is an important tool for retrieving tasks. The cosine calculation is an advanced ranking technique, so the retrieval results could be limited to the simple TF-IDF ranking algorithm. It is worth noting that the Python library, scikit-learn, which provides ready-made methods that generate the vectors and calculate the cosines, was not used.

Probabilistic Retrieval Model

The Probabilistic Retrieval Model code also performs for large search engine repositories and is an important tool for retrieving tasks. The calculation of the bm25 score is an advanced ranking technique, where it offers efficiency in the recovery of tasks, removing with the constant saturation coefficients b and k, the terms that appear very often in the collection and are usually of little importance information.

4.c. Filter search results by various criteria

4.c.1 Planning

Filtering the results is an option that the user has after entering the query he wants to search for and the algorithm he chooses to do the search. Essentially it offers the user the choice of the results he got to filter them either based on their author, or based on the date they were published. This is achieved through a second query in which the user must enter an author's name (in the case that the "Authors" field has been selected as filtering) so that only those in which he/she is included in the authors are displayed from the results obtained who asks Accordingly, he must do the same if he wants to filter by date, but the field must be filtered by "Date".

4.c.2 Implementation

To implement the filtering, we needed access to all the data we have in json format as well as the algorithm used for the search. Knowing the algorithm that was used, the filtering function knows from which table it should get the results (boolean_results if boolean retrieval was used etc). These results are the doc_ids of the texts printed by each algorithm. The function that deals with filtering takes the filter selected by the user (date, author) and also a query that contains what the user wants to search for in the results. Later, the necessary check is done with the author or date fields and the texts that have the same information as the user's query are printed.

Some illustrative case studies from user queries are as follows:

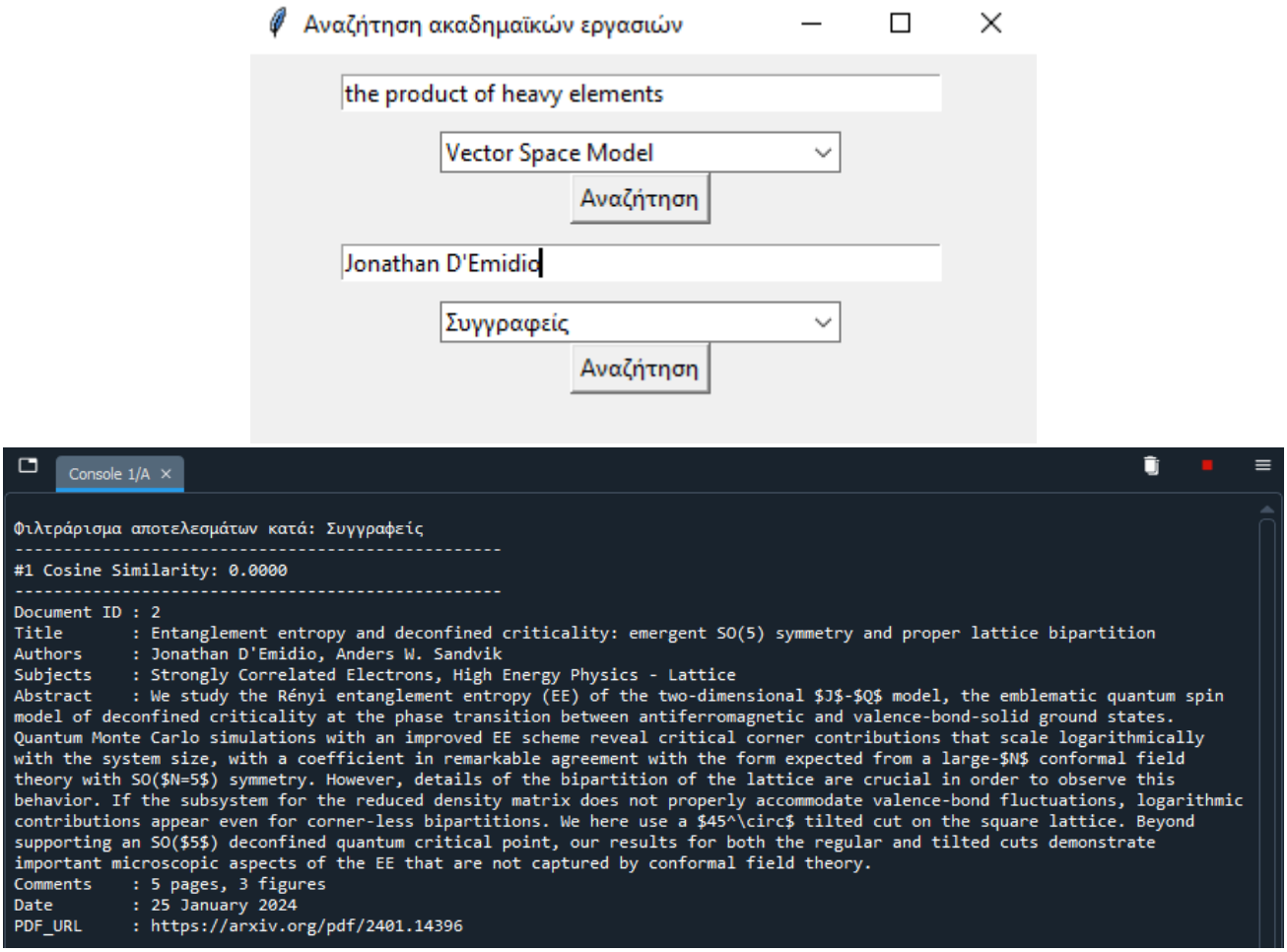
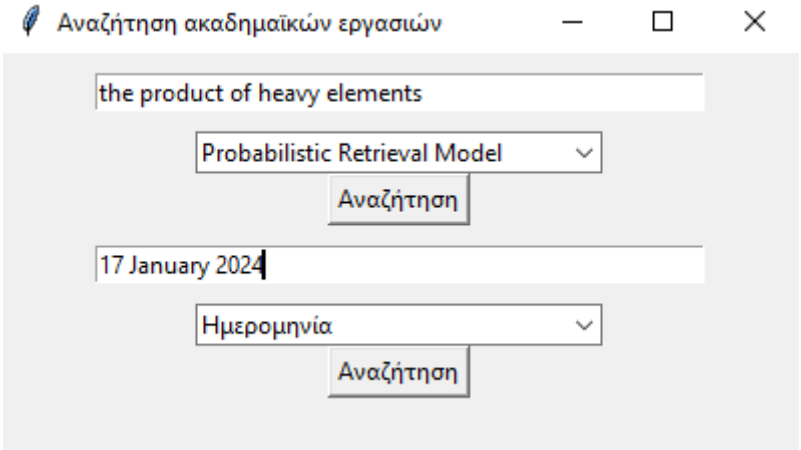


Figure 4.C.1 Filtering by Author



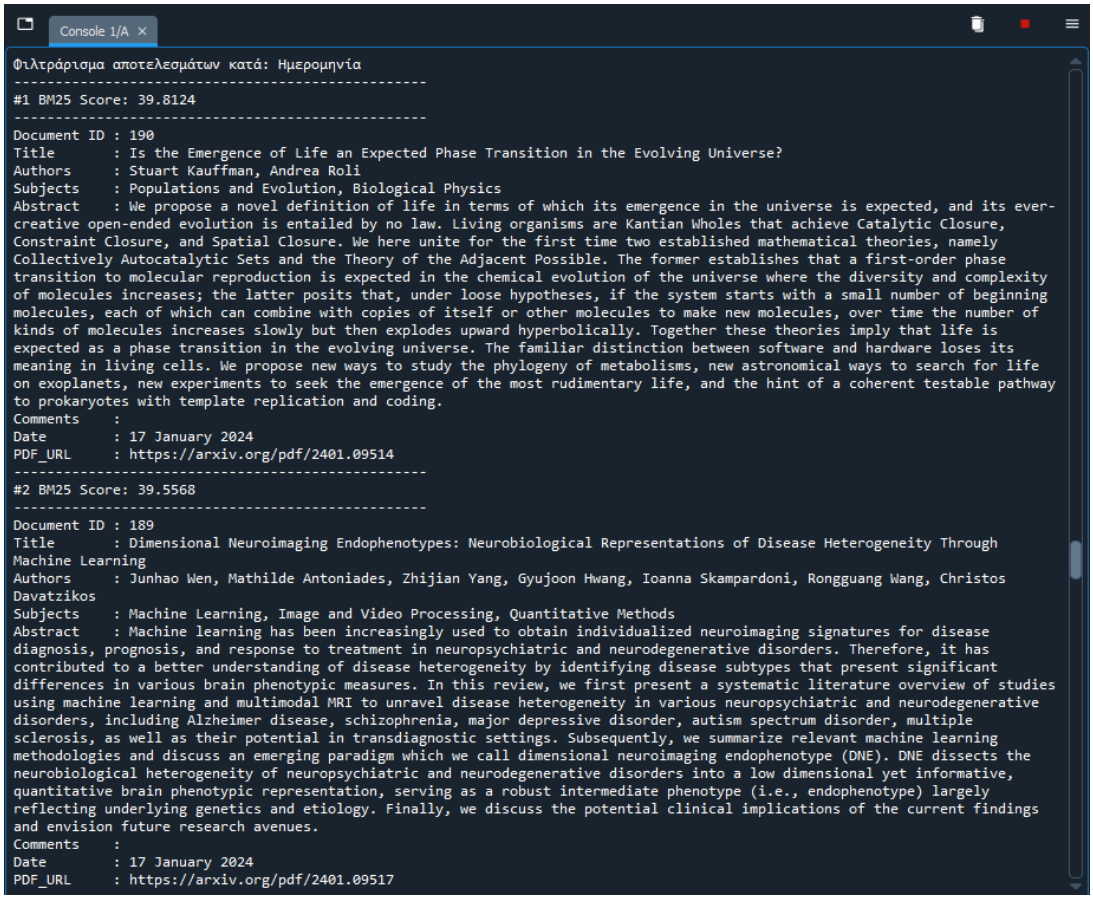


Figure 4.C.2 Filtering by Publication Date

4.c.3 Evaluation

The function to filter the results works well, however it is a rather simplistic function that requires the user to precisely enter the items they are interested in looking for in the algorithm results. The only case where the function does not work is when the user enters more than 1 author as the input to the filter. However, for an author the function satisfies the filter logic.

Query Processing

Planning

The process of query processing (Query Processing) followed by the Boolean Retrieval algorithm is based on Boolean algebra and set theory, using three basic Boolean operations: OR, AND and NOT. These functions affect the search results according to their position in the query received by the user. The relevant documents are retrieved using the inverted index. The theoretical/mathematical background of the design model is as follows:

AND

$$AND\_RES = Q_0 \cap Q_1 = \{doc_{id} \mid doc_{id} \in Q_0 \; and \; doc_{id} \in Q_1\}$$

AND_RES The texts containing both terms Q_0, Q_1

Q_0 The texts containing the term Q_0

Q_1 The texts containing the term Q_1

OR

$$OR\_RES = Q_0 \cup Q_1 = \{doc_{id} \mid doc_{id} \in Q_0 \; or \; doc_{id} \in Q_1\}$$

RES The texts containing either the term Q_0, or the term Q_1 or both

Q_0 The texts containing the term Q_0

Q_1 The texts containing the term Q_1

NOT

$$NOT\_RES = \neg Q_0 = \{doc_{id} \mid doc_{id} \in D \setminus Q_0\}$$

RES Texts that do not contain the term Q_0

D The texts of the collection

Q_0 The texts containing the term Q_0

The process is carried out for simple Boolean user queries e.g. black OR white, NOT black, black AND WHITE. For complex queries, the Boolean Retrieval algorithm determines the priority order of operations, where it breaks the complex query into simple sub-queries and sends them in order of priority for processing.

Implementation

The query processing implementation was performed in a separate module (query_processing.py). Initially, the terms of the Boolean query are separated into verbal units and using the inverted index from a routine, all the terms of the Boolean query are replaced with the posting lists with the texts (doc_id) in which they appear. Boolean operations are performed on these lists. If the term does not exist in the index, then the routine handles the error.

Retrieving the relevant documents for each logical operator is implemented as follows:

AND

If the current term is the AND operator, the repository doc_ids that belong to the list of doc_ids displayed by the previous term and the corresponding list of the next term are retrieved.

OR

If the current term is the OR operator, the repository doc_ids that belong to the list of doc_ids displayed by the previous term and the doc_ids that belong to the list of the next term are retrieved.

NOT

If the current term is the 'NOT' operator, the repository doc_ids that are not in the list of doc_ids that the next term appears in are retrieved

The implementation focuses on handling the Boolean functional operators AND, OR, NOT for query processing, using sets for the set operations.

Evaluation

Boolean Retrieval algorithm . The philosophy "belongs to x text, does not belong to y text" provides an efficient way to retrieve texts based on the query posed by the user. The design presents the theoretical background based on set theory, while the implementation presents the practical background based on Boolean algebra. The replacement of the term-words with the lists of the texts of the collection they belong to, according to the inverted index, speeds up the process of retrieving the texts that the user poses as a query. Representing the process with set theory and Boolean algebra make text retrieval efficient.

Ranking of results (Ranking)

Planning

The design model of ranking results (Ranking) firstly follows the application of the simple TF-IDF (Term Frequency-Inverse Document Frequency) ranking algorithm and then the application of advanced Cosine Similarity ranking techniques followed by the Vector Space Model and Okapi BM25 recovery algorithm that the Probabilistic Retrieval Model retrieval algorithm follows . The aforementioned applications contribute to the estimation of the semantic importance of the terms in a set of texts, where, based on a user query, the cosine similarity coefficients or the BM25 coefficients (BM25 Score) are calculated between the terms of the user query and the terms of the texts in the collection. The TF-IDF model follows the calculation of the TF × IDF product for each term of the collection and for each term of the user query, while the Cosine Similarity and Okapi BM25 models follow the calculation of the cosine similarity coefficient and a score respectively based on these two events. Finally, the search results are sorted by the highest coefficient. The theoretical/mathematical background of the design model is as follows:

TF-IDF

$$TF_{t,d} = \frac{N_{t,d}}{\sum_k \quad N_{k,d}}$$

$TF_{t,d}$ The frequency of occurrence of the term t in text d

$N_{t,d}$ The number of occurrences of the term t in text d

$\sum_k \quad N_{k,d}$ The number of terms in the text d

$$DF_t = |D_t|$$

DF_t The frequency of occurrence of texts where the term t appears

$|D_t|$ The number of texts where the term t appears

$$IDF_t = log\ log\ \frac{|D|}{DF_t}$$

IDF_t The inverse frequency of texts where the term t occurs. Terms with a high frequency of occurrence in texts are of little importance information and this is due to the fact that the logarithm tends to 0.

$|D|$ The number of tasks collection

DF_t The frequency of occurrence of texts where the term t appears

Cosine Similarity

$$cos\ cos\ (q,d) = \frac{q \times d}{\|q\| \times \|d\|} \rightarrow cos\ cos\ (q,d) = \frac{\sum_{t=1}^{n} \quad (q_t \times d_t)}{\sqrt{\sum_{t=1}^{n} \quad (q_t)^2} \times \sqrt{\sum_{t=1}^{n} \quad (d_t)^2}}$$

$\sum_{t=1}^{n} \quad (q_t \times d_t)$ The inner product of the user query vector with the vector of each text in the collection

$\sqrt{\sum_{t=1}^{n} \quad (q_t)^2}$ The Euclidean distance of the user query vector

$\sqrt{\sum_{t=1}^{n} \quad (d_t)^2}$ The Euclidean distance of the vector of each text in the collection

Okapi BM25

$$(q,d) = \sum_{t=1}^{n} IDF(q_t) \times \frac{TF(q_t,d) \times (k+1)}{TF(q_t,d) + k \times (1 - b + b \times \frac{|d|}{avg_{dlen}})}$$

$\sum_{t=1}^{n}$ $IDF(q_t)$ The inverse frequency of occurrence of the texts where the query terms appear

$TF(q_t,d)$ The frequency of occurrence of the query terms in the text d

k Positive parameter (k = 1.2) which controls the terms that appear very often in a text (TF) and usually, this is information of little importance for this text

b Positive parameter (b = 0.75) which controls the terms that appear very frequently in texts (DF) and usually, this is information of little importance for the collection

$|d|$ The size of the text d

avg_{dlen} The average size of the collection

Implementation

The ranking algorithms are implemented in a separate module (ranking.py) and include the following routines:

TF-IDF

- TF-IDF calculation of the terms of the tasks:
  For each task collection term (from the abstract fields):
  1. Text preprocessing (step 2) and separation into verbal units (tokenization)
  2. Calculation of the frequency of each term in the text (Term Frequency)
  3. Calculation of the frequency of occurrence of the texts in which each term appears (Document Frequency)
  4. Calculation of the inverse frequency of the texts that each term appears (Inverse Document Frequency)
  5. Calculation of the frequency of each term in the text (TF × IDF)
- Compute TF-IDF of user query terms:
  For each term of the user query (query):
  1. Preprocessing of the query (routine of step 2) and separation into verbal units (tokenization)
  2. Calculation of the frequency of each query term in the query (Term Frequency)

3. Calculation of the inverse frequency of occurrence of the texts in which each term of the text appears (Inverse Document Frequency)
4. Calculation of the frequency of each query term in each text (TF × IDF)

Cosine Similarity

- Calculation of the cosine similarity (Vector Space Model) between the user query and the tasks (Cosine Similarity):
  For each text in the collection, its similarity to the user query is calculated as follows:
  1. Compute the inner product of the user query vectors with the vector of each text in the collection
  2. Computing the Euclidean distance of the user query vector
  3. Calculation of the Euclidean distance of the vector of each text in the collection
  4. Calculate the similarity between the user query and the text
- Ranking tasks based on cosine similarity

Okapi BM25

- Calculation of the BM25 Score (Probabilistic retrieval model) coefficient between the user query and the tasks:
  For each text in the collection, its similarity to the user query is calculated as follows:
  1. Query preprocessing (step 2 routine)
  2. Calculation of the size of the task
  3. Calculate average task collection size
  4. Calculation of the frequency of occurrence of the texts in which each term appears (Document Frequency)
  5. Calculation of the inverse frequency of the texts that each term appears (Inverse Document Frequency)
  6. Calculation of frequency of appearance of the query term in each text (Term Frequency)
  7. Calculation of BM25 coefficient for each term of the query
- Ranking of documents based on the BM25 Score

Evaluation

Ranking algorithms are important tools for search engines in document retrieval efficiency. The simple TF-IDF ranking algorithm calculates how important a term is in the collection, while the advanced Cosine Similarity and Okapi BM25 algorithms focus on the similarity between the user query and the texts in the collection. The philosophy of "how close some texts are to the user's query" provides an efficient way to retrieve texts based on the user's query.

5. System evaluation

5.a. Datasets

The data is extracted from arxiv.org and with various scientific fields such as Physics, Mathematics, etc. As a repository of scientific works, it provides a large number of documents and words for the index.

5.b. Evaluation Scenarios

The user can through the search engine search to make any search he wants in contents from a large index, however the final results that will be displayed are limited in size since they fit in the console of each IDE.

5.c. Python libraries

The libraries that were used to create the search engine are several, some of them are json which allows data to be stored in a specific format, request which allows retrieving a page through an HTTP-GET request and bs4 which returns the page in unstructured HTML format (parsing)

5. d. Applications of metrics

The search engine to start in an average test takes about 12 seconds and fetching the relevant data from arxiv.org takes about 12 seconds on average. The papers it collects range from 200 to 800 because as mentioned in the implementation of the web crawler, it addresses 2-8 corresponding links to retrieve the data that will be stored in the repository (dataset). Indicative assessment measures are:

Precision is the percentage of retrieved texts that are relevant

$$P = \#relevant\ recovered\ texts/\#recovered\ texts$$

Recall is the percentage of relevant texts that have been retrieved

$$R = \#relevant\ recovered\ texts/\#recovered\ texts$$

F1 Score is the percentage balancing precision and recall

$$1/F = 1/2*(1/P + 1/R)$$

5.e. Analysis and Improvements

The search engine cannot filter multiple authors and this is something that would make it even easier for the user. The GUI is also quite simple and could be more user-friendly and visually nicer in terms of its layout. Certainly, the efficiency of the algorithms is not the best possible for everyone, but this does not significantly affect the accuracy, let alone the use of the application.

RECOVERY OF INFORMATION

RECOVERY OF INFORMATION

Thank you for your attention.