A Gender Equality Observatory on Scientific Research

Styliani Kyrama[*]
kyrastyl@csd.auth.gr
Aristotle University of Thessaloniki
Greece

Vasileios Moschopoulos[*]
vmoschop@csd.auth.gr
Aristotle University of Thessaloniki
Greece

Dimitrios Tourgaidis[*]
tourgaidi@csd.auth.gr
Aristotle University of Thessaloniki
Greece

Georgios Michoulis[*]
gmichoul@csd.auth.gr
Aristotle University of Thessaloniki
Greece

Abstract

This work is dedicated to studying the gender balance and differences between men and women in the academic sector. The term “gender balance” refers to an equitable distribution of life’s opportunities and resources between women and men, and/or the equal representation of women and men. In science and academia inequalities exist in different career stages from postgraduate level to top-tier academia. This work aims to create an observatory of this phenomenon by exploring curated data extracted from Google Scholar. The data is formatted in an interaction graph representation between different academic authors. We explored a number of different graph metrics regarding gender, but also the interactions between different academic authors. All the aforementioned are presented and visualized on a website we built, which also provides a way of exploring the data of authors from the 6 different universities studied.

Keywords: gender equality, diversity, academia, graph analysis, graph database

ACM Reference Format:
Styliani Kyrama, Vasileios Moschopoulos, Dimitrios Tourgaidis, and Georgios Michoulis. 2021. A Gender Equality Observatory on Scientific Research. In *Proceedings of ACM Conference (Conference’17).* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

The term “gender balance” refers to an equitable distribution of life’s opportunities and resources between women and men, and/or the equal representation of women and men. In science and academia inequalities exist in different career stages from postgraduate level to top-tier academia.

Women in universities constitute about 1/3 of the total staff (only 28% of faculty members). It has also been found that the higher the position in the academic hierarchy, the lower the participation rate of women, which makes the distribution of their presence “pyramidal”, with the majority of female teaching and research staff at the lower level of development [1]. Women in higher education hold fewer senior management and managerial positions, tend to be more represented in particular fields, such as the Humanities and Economics, and show, compared to male university professors, lower productivity, less and fewer educational permits. Women are promoted at a slower pace than men, especially when they have family responsibilities, as there is not enough government care with adequate measures to help “reconcile” family and employment. Research on the position of women in academia confirms the “horizontal” and “vertical” discrimination observed in the academic staff of universities internationally [1].

A look at the positions occupied by women academics, we discover that women are mostly concentrated at the “entry” level, and therefore occupy the lower positions of the academic community [2]. The inequalities between women and men’s positions remain regarding a scientific and academic career. Women’s academic careers remain remarkable characterized by strong vertical segregation. According to She Figures (2013), women accounted for only 44% of academic grade C staff, 37% of academic staff grade B, and 20% of academic staff grade A [2].

Especially for Greek women, although there have been many steps forward, they still seem to be at disadvantage. Regarding the EU countries, Greece has the lowest ranking in the Gender Equality Index [3], with a score of 0.122. Similarly, the population of the country, data from 2019, is 44.17% women. In addition, rates of domestic violence appear to be rising, having risen by more than 30% in the last six years. On the contrary, Greek women have made great strides in academia. More than 50% of Greek citizens who obtain a university degree are women [4].

[*] All authors contributed equally to this research.

This paper is published under the Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International (CC-BY-NC-ND 4.0) license. Authors reserve their rights to disseminate the work on their personal and corporate Web sites with the appropriate attribution.

Conference’17, July 2017, Washington, DC, USA

© 2021 IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC-BY-NC-ND 4.0 License.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1.1 Our Contibution

This work aims to create a gender balance observatory on scientific research, by exploring curated data extracted from Google Scholar. More specifically, we created a web application that visualizes the analysis made on data collected for some faculty members, in order to be presented and emphasized the inequality of the two genders, in academia. Both the data collection and the architecture and functionality of the system are described in detail below in this technical report.

1.2 Limitations

Due to the domain's wideness that our projects regards, in order to able to handle it, we had to delimit it. We observed the gender equality in the the Academia field based on 6 universities; Universities of Porto, Bordeaux, Lodz, Bochum, Oulu and Aristotle (AUTh). Furthermore, we examined only the departments of those universities that concern the fields of computer science and electrical engineering. In order to implement the gender equality observation, we exclusively utilize data that we extract from the departments' faculty member (professors) and their coauthors in scientific papers they have published.

Based on the aforementioned details, the domain's entities are; **1)** the universities we examine **2)** the departments that belong to those universities **3)** the professors that work in the departments **4)** the professors' coauthors with whom they have cooperated with to write and publish a scientific paper.

1.3 Outline

The rest of the paper is structured as follows. In Section 2 we present the way we decided to model the domain that our project regards, as well as the database engineering process. In Section 3 we describe in detail the tool, steps and procedure taken in order to extract the data, which we store in the database, that are used to examine the "gender balance" in the Academia field. Section 4 is dedicated in presenting the overall architecture of the system we created, from the information extraction to the end-user data flow. We explain in detail the analysis done on the obtained information, from all different aspects (in 5.1 and 5.2), and present the results and provide insights about gender equality (in 5.3). In Section 6, we outline the components and functionality of the web application in which the aforementioned results are visualized. Finally, we discuss possible extensions to our project and future plans for extending the functionality of our web application, in Section 7.

2 Domain modeling

Based on the problem definition described in 1.2, we decided that modeling the domain into a graph representation would be the most appropriate way to model it. The reason is because the domain includes a number of distinct entity types (nodes), where some among them have a natural correlation (edges) with each other. Consequently, through modeling the domain as a graph, we can utilize established metrics of the *graph theory* field to produce *gender equality metrics* in order to examine the gender equality in the Academia field, which is our objective.

Academia Network. Since the Academia field is the domain that our project regards, we resolved to name the produced network as *Academia network*. The schema of the Academia network, namely all the nodes and edges types, are presented at Table 1.

Nodes of the Academia Network	
Real life entities	**Corresponding node**
University that we examine	University
Department that belongs to a university	Department
Professor that works at a department	Professor
Coauthor of a professor in a paper	Coauthor
Edges of the Academia Network	
Real life entities' correlations	**Corresponding edges**
Department belongs to university	belongsTo
Professors works at a department	worksAt
Professor/Coauthor cooperates with a Professor/Coauthor in a paper	cooperateWith

Table 1. The mapping of the domains' real life entities and their correlations to the corresponding nodes and edges in the Academia network

2.1 Database engineering

Due to the fact that we decided to model the field that our project regards as a graph, we concluded to use the **Neo4j** database in order to store the Academia network, which is a NoSQL graph database. The reasons is due to the manner that the data are stored and the queries are implemented in Neo4j. Both are implemented based on a graph's structure, consequently, the data handling and the information extraction comes easier compared to the case if the Academia network would have been stored in some other type database (e.g RDBMS, document-oriented, column-oriented, etc.).

As a graph database, Neo4j stores the data as nodes and edges, where different types of nodes and edges can be declared. Both can have attributes or not. Furthermore, an essential feature of Neo4j is that all the edges are directed. Consequently, Neo4j stores only directed graphs, but it provides the option to query patterns without taking the direction of an edge into consideration, i.e. handle a graph as undirected.

Now for our case, all the node and edge types of the Academia network presented in Table 1, are mapped to corresponding nodes and edges in Neo4j. Table 2 demonstrates the node and edge types of the graph (Academia network) stored in Neo4j, as well for each edge type the corresponding source and destination node type, will Figure 1 describes the schema of the stored graph in Neo4j based in Table 2.

Nodes and Edges in Neo4j Graph		
Edge Type	**src(node type)**	**dst(node type)**
belongsTo	Department	University
worksAt	Professor	Department
cooperateWith	Professor, Coauthor	Professor, Coauthor

Table 2. Node and edge types of the graph (Academia network) stored in Neo4j. Source and destination node type of each edge type

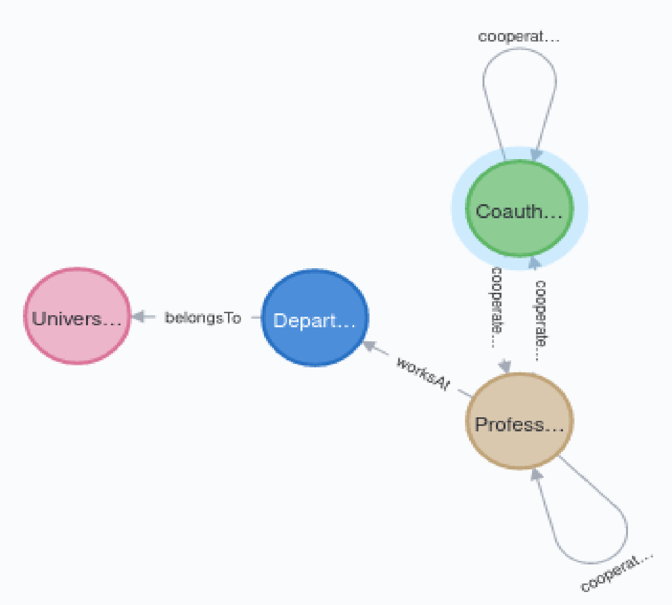


Figure 1. Neo4j graph schema

The interpretation of the graph-schema in Figure 1 is simple and does not require further explanation, besides the 2 edges that connect the Professor-type and Coauthor-type nodes. Normally, this correlation should have been expressed through a undirected edge, because when a Professor/Coauthor cooperates with another Professor/Coauthor, the latter cooperates with the former at the same time. However, as mentioned before, Neo4j does not support undirected edges, so this correlation can only be implemented through 2 directed edges.

3 Dataset

In this section we discuss the tool we used for scraping, the data collection procedure and the structure of the extracted data. All collected data described in this section was used afterwards in order to create the Academia network in Neo4j.

Generally speaking, we provide an overall outline of the data collection procedure. Data collection actually consisted of several different stages, due to changing project requirements, re-querying failed authors and merging results. Each stage used and produced certain data, with data from previous stages being passed on to the next ones.

The section is structured as follows: **(i)** we describe the tool we used to collect the data (scholarly Python API), **(ii)** we outline the pipeline regarding the data extraction, and **(iii)** we detail the structure of the final extracted data.

3.1 The scholarly Python API

scholarly[1] is a Python library built on top of the Google Scholar web API, utilizing the Selenium web driver to scrape information regarding different authors, as well as publications, from Google Scholar.

scholarly provides a number of possible information types for authors and publications. We used scholarly to get authors (professors of the universities and their coauthors) information only, according to our needs. Authors can be queried in terms of name, id, as well as keywords. Searching by name and keywords returns a Python generator of different authors, while searching by id returns a single author. In our tasks we used searching by name and id. When searching by name, we would simply pick the top item in the generator. This, however, required of us to provide the name and affiliation in an appropriate manner in order to retrieve the correct author.

Authors are structured as dictionaries containing a number of different objects and string and int values. Possible information that can be retrieved using scholarly include:

- Basics: This includes basic information such as the name of the author, affiliation and a list of scientific interests.
- Indices: Indices include h-index and i10-index indicators, total citations, as well as 5-year analogues for each of them.
- Counts: Counts comprise a list of numbers of citations for every year the author has been active for. The sum of all citations is roughly equal to the citations retrieved in the indices information.
- Coauthors: Coauthors contain a list of all coauthors with their basic information.
- Publications: Publications include a list of publications, structured in a way as publications would normally be retrieved if queried for with scholarly, and containing basic information.

[1] https://github.com/scholarly-python-package/scholarly

We used all types of information in our tasks.

It is worth mentioning that the search functionality returns only basic information for both authors and publications by default. A 'fill' method can be used to retrieve all other types of information, while providing '[]' as a parameter option, all types of information can be retrieved.

Due to the fact Google Scholar keeps track of IPs executing consecutive queries, it can identify them as bots and block them. A way to bypass this is and continue using scholarly is by using proxies. For such occasions, a ProxyGenerator object is provided by scholarly which can produce a proxy on the fly and swap it with the current one. Out of the 5 different proxy options, we chose to use the FreeProxies option, which uses the free-proxy Python library and automatically provides new proxies, without requiring of the user to specify any parameters. Other options included Tor proxies, Luminati proxies and single, user-defined proxies.

Finally, scholarly also provides a number of different methods for different uses, such as retrieving Bibtex entries and publications referencing other publications, providing custom URLs for retrieval, a pretty print function and author and publication parser modules. The aforementioned methods were not used in our tasks, and are only cited for the interested reader.

3.2 Data collection pipeline

In this subsection we outline the pipeline we developed to collected the data. As mentioned, the data collection procedure consisted of several distinct stages, but we will not elaborate any further regarding these stages.

The data collection pipeline was structured as follows: **(i)** firstly, we gathered a list of information for the faculty members of the universities we confined our search in, **(ii)** secondly, we successfully managed to extract data for a large portion of the faculty members using scholarly, and **(iii)** finally, we extracted similar data regarding the coauthors of the successfully queried faculty members, using scholarly once more.

3.2.1 Gathering a list of faculty members. Our first task in the pipeline was to collect an initial list of the faculty members of each of the 6 selected universities. As mentioned in Section 1.2, we limited our search to the faculty members of departments related to Computer Science and Electrical Engineering. The list contained for each member:

- The first and last name
- The university affiliation
- The gender; male or female
- Their role in the affiliation, e.g. assistant professor, associate professor, etc.
- The department they belonged to, e.g. Computer Science, Information Technology, Computer Engineering, Electrical Engineering, etc.

The collection of this data was a collaborative effort, done manually through the website of each university. All data was stored in a sheet file, to be used for the next step in the pipeline, with the number of the collected faculty members being 509. We highlight here that the role and department were exclusive to the initially collected faculty members, since Google Scholar could not provide us with this kind of information.

3.2.2 Extracting data for faculty members. The second step in the pipeline was to extract data from scholarly for each of the faculty members. Our queries were done with the search by name functionality, using the full name of authors and the university's city as the search parameter.

In order to maximize our successful query rates, we had to make sure that the first and last name of the faculty members aligned with their profiles in Google Scholar. Other challenges we faced concerned the use of language specific characters, non-existent Google Scholar profiles, identically named authors and the fact that certain faculty members could not be retrieved using the university's city as a keyword. To combat these challenges we ensured that the names of faculty members that had a corresponding profile in Google Scholar were properly formatted and introduced an alternative keyword parameter, in place of the university's city.

Through re-queries, edits and the use of alternative keywords, we managed to achieve a query rate of ~66%, successfully extracting information for 337 out of the 509 faculty members. The data we collected with scholarly comprised the following:

- Google Scholar ID
- Name
- Picture URL. The picture URL was deemed useful for the profile construction of individual authors and coauthors for the web app.
- Indices. Indices included h-index, i10-index and total citations statistics, as well as last-5-year analogues.
- Number of publications
- A list of citations for each year the author has been active for. The sum of all citations nearly equaled the total citations statistic described above.
- A list of scientific interests
- A list of Google Scholar IDs of their coauthors

3.2.3 Extracting data for co-authors. The final step in the pipeline, in order to build the complete graph, was to extract data for each of the coauthors of the faculty members, using the lists of coauthor IDs described in the previous subsection.

The data we extracted for coauthors was similar to the data extracted for faculty members, with the exception of the list of citations per year and scientific interests. The total number of unique coauthors was 2320. Due to the fact that the Google Scholar API, and consequently scholarly, does not return the gender of authors, we had to collectively label the gender of coauthors manually.

We also kept a list of Google Scholar IDs for the coauthors of coauthors for possible future extensions of our Academia network. Important tasks of this pipeline step also included re-querying failed coauthors and ensuring that no faculty member duplicates existed in our list of coauthors.

3.3 Building the graph data

Having collected all of the necessary data, we needed to structure them in an appropriate format to be loaded into the graph database. The eventual graph data comprised 3 CSV files:

- Professors. The professors CSV contained extracted data from scholarly for 340 faculty members, plus manually collected gender, university affiliation, role and department.
- Coauthors. The coauthors CSV contained similarly extracted data of 2320 coauthors, plus their manually labeled gender.
- Relations. The relations CSV contained 6260 unique, undirected edges between all professors and coauthors, using their full names as identifiers.

4 System Architecture

Before moving on to the analysis made regarding gender equality, we present the architecture of the system we developed, which is very important as it provides an overall view of the physical deployment of the software system, but also of the data flow. The architecture is depicted in Figure 2.

4.1 From WWW to Neo4j

As described in section 3, the data for each faculty member and their coauthors we used for the analysis were extracted from Google Scholar, which is a freely accessible web search engine. To create the Neo4j database, as discussed in section 2, we implemented a basic component that utilizes and loads the 3 CSV files with the necessary data. This component, which is presented as a **Database Loader** in Figure 2, creates all the entities and the relationships among them, in order to achieve the construction of the ***"Academia Network"***. This network is used for the analysis described in more detail, in section 5.

4.2 End-User Data Flow

In order for a user to be able to interact with the system and retrieve information from the database, a web application was created, the navigation and functionality of which is described in more detail in the following section 6.

For the integration of the front end with the back end of the web application, the Flask service, or as it is otherwise characterized, micro-framework, was used. The Flask is essentially based on two other very useful tools, the Werkzeug, a comprehensive Web Server Gateway Interface (WSGI) web application library, and Jinja2 a fast, expressive, extensible templating engine.

When the user navigates through the pages of the application, sends HTTP requests to the Flask built-in web server. Flask invokes the appropriate python function that is responsible for returning an HTTP response. To retrieve the information requested from the database, Flask sends some queries written in "cypher" to the Neo4j system database. The result of the query is returned from the base to the Flask in JSON format and then it is being processed in order to get to the appropriate format so that they can be displayed, using jinja2, on HTML pages. After the pre-processing is done, Flask returns to the user the appropriate HTML and Javascript code.

5 Analysis

In this section, we present the analysis made using the academic network we built, both at a general level, regarding the elements of the graph, and at a more specific level, based on gender.

5.1 General Metrics

First, we examined the network generally, and obtained useful information about the graph as a whole, by applying some community detection algorithms. These algorithms are used to evaluate how groups of nodes are clustered or partitioned, as well as their tendency to strengthen or break apart.

We applied the Louvain Modularity algorithm; a hierarchical clustering algorithm, that recursively merges communities into a single node and executes the modularity clustering on the condensed graphs. This algorithm can easily detect communities in large networks, and maximize the modularity score for each community. Modularity as a metric quantifies the quality of an assignment of nodes to communities, i.e. if nodes are assigned to the correct cluster. This means evaluating how much more densely connected the nodes within a community are, compared to how connected they would be in a random network.

Besides the Louvain method, which uses the modularity in order to measure the quality of clustering, there is an algorithm called Modularity Optimization algorithm, which tries to detect communities in the graph based on their modularity. As already mentioned, modularity is a measure of the structure of a graph, measuring the density of connections within a module or community. Graphs with a high modularity score will have many connections within a community but only few pointing outwards to other communities. This algorithm will explore for every node if its modularity score

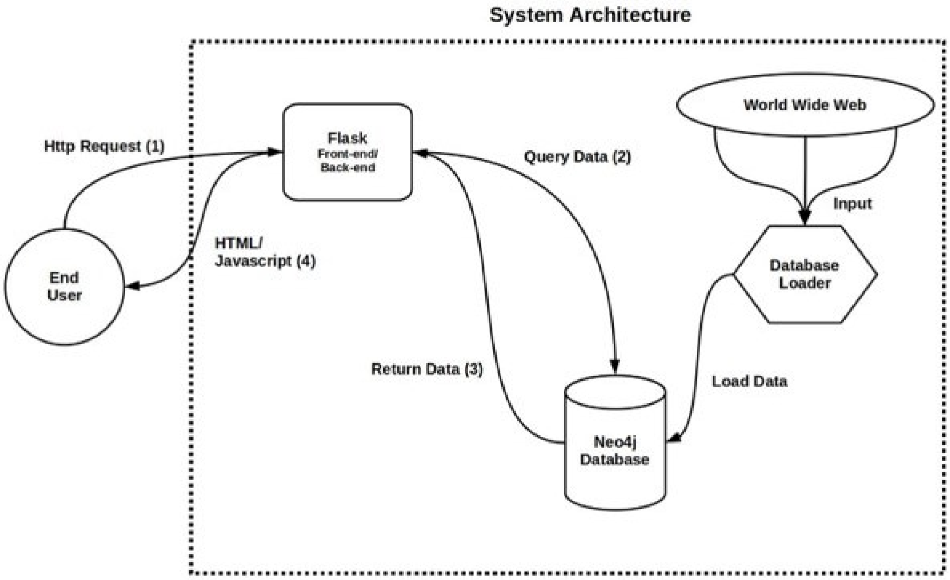


Figure 2. System architecture

might increase if it changes its community to one of its neighboring nodes.

Another algorithm that detects communities, or otherwise, connected components in the graph is the WCC algorithm. WCC is a very useful algorithm that is used in analysis to understand the structure of a graph, and discover the disconnected components.

The number of communities for each algorithm may vary from execution to execution, but in general, the three aforementioned community detection algorithms gave us completely different results, with **Louvain** obtaining about **115 communities**, **Modularity Optimization** returning about **530**, and **WCC** algorithm **81 connected components**.

We, also, applied a triangle count algorithm, in order to discover the number of the 3-cliques in our network. Triangle counting has gained popularity in social network analysis, where it is used to detect communities and measure the cohesiveness of those communities, but it can also be used to determine the stability of a graph. Our academic network consists of **4756 triangles**, or otherwise 3-cliques.

5.2 Gender Equality Metrics

In terms of gender, we turned the analysis in two main directions; traditional analytics which relies on statistics obtained from entities' information, and graph analytics, which exploits the structure of the graph to analyze relationships and correlations between different entities.

Traditional Analytics. As already mentioned in section 3, for each faculty member we gathered some basic information. We exploited this information in order to perform an aggregative analysis, by extracting the average values of each metric, per gender. The results of this analysis are show later in this section, in 5.3.

The metrics examined are the number of publications, citations, and co-authors, as well as the two indices obtained for each faculty member from their Google Scholar profile, h-index, and i10-index, which are briefly explained below.

H-index is a number **h** indicating that this individual has at least h publications with at least h citations.

I10-index is a number **i** indicating that this individual has at least i publications with at least 10 citations.

Graph Analytics. We apply graph analytics algorithms to the academic network we created in order to examine the difference between genders. More specifically, we used algorithms such as degree centrality, betweenness centrality, PageRank score, closeness centrality, and count of triangles. All these algorithms were performed taking into account the gender and derived a cumulative result, as the average value of each graph metric, per gender. Before analyzing the results, we provide an explanation of each metric we used and its applicability in our domain.

Degree Centrality can be considered as the number of linkages each faculty member has, that to our academic network are either entity type Professor or Co-author. In real-life terms, degree centrality represents the number of collaborators each person has, so individuals with high degree centrality are most important people in the network.

Betweenness Centrality is a way of detecting the amount of influence a node has over the flow of information in a graph and is often used to find nodes that serve as a bridge from one part of a graph to another. Given the above, a faculty member with high betweenness centrality will be an intermediary, for example, a Professor who has collaborators to many different Universities, and so becomes a "human bridge" between them.

PageRank Score in a graph measures the importance of a node in terms of i. number of connections and ii. the importance of each of those connections.

Closeness Centrality is a way of detecting nodes that are able to spread information very efficiently through the graph. In a graph, nodes with a high closeness score have the shortest distances to all other nodes, but in real-life we could say that individuals with high closeness centrality are in a favourable position to control and acquire vital information and resources within the organization.

5.3 Results

In Table 3 we present the results from the traditional analytics. As we can see, there is a clear imbalance between men and women. Men seem to be more active in research and academia in general. They have on average more co-authors and also a lot more publications than women. Regarding the h-index, we do not notice any significant difference, but when it comes to the i10-index, we can see that a man has on average at least 20 more posts than a woman, with at least 10 citations. Regarding the number of citations, we can observe that there is a significant difference, as a man has on average at least 1000 more citations than a woman.

Metric	Men	Women
Publications	184.64	127.73
Co-authors	12.75	9.26
Citations	3662.15	2203.53
h-index	23.31	18.82
i10-index	55.58	36.98

Table 3. Traditional Analytics Results

All the graph metrics discussed in section 5.2, along with the average number of triangles an individual participates in, have been calculated per gender and their average was visualized. For example, regarding the degree centrality, we calculate and present the average degree centrality per gender, i.e. the number of connections that a man would have on average, and the same for women.

The results, which are also depicted in Figure 7 but also Table 4, have shown that regarding all metrics, there is a difference between genders. Men tend to have more connections than women, be more important nodes in the academic network, more influential, and participate in more cliques. As for the closeness centrality, we observed that men and women have almost the same average value, meaning that there is no favoritism for men to be the ones who control and spread more efficiently the information inside the network.

Besides the above analytics, we performed a series of Top-K queries regarding mainly some of the graph metrics that already have been mentioned. We obtained from the database the top 10 faculty members with the i. most co-authors, ii. highest pagerank score, iii. highest betweenness centrality, iv. highest degree centrality, and v.highest closeness centrality. From the above results, we calculated the percentage of participation of each gender on those. These are presented in Table 5.

Metric	Men	Women
Degree Centrality	20.59	15.25
Betweenness Centrality	0.54	0.32
PageRank Score	2.09	1.73
Closeness Centrality	0.14	0.15
Triangle Count	11.89	6.78

Table 4. Graph Analytics Results

As we can see, in none of the above queries do we have an equal participation rate for men and women. In most of them, men are the vast majority of the results, such as in the "top 10 faculty members with the highest pagerank score" query, there is only one woman, while only two in the "top 10 faculty members with the highest closeness centrality" query.

The query "faculty members with higher betweenness centrality" is an extreme case, as there is not a single woman in the top 10 results. For the remaining two queries, we can see that the results are distributed almost equally between genders.

Metric	% Men	% Women
Most Co-authors	60	40
PageRank Score	90	10
Betweenness Centrality	100	0
Degree Centrality	60	40
Closeness Centrality	80	20

Table 5. Top-10 Results

6 The web application

In this section we will analyze the front-end of the project. In order to present the above analysis we had to create a web application, as described in Section 4, the code of which is publicly available at Github. In the front-end we chose to use Bootstrap and Jquery for the designing part and also we used Flask as back-end to manage design components and connect with the database. All in all we designed four web-pages / tools and we will describe the in detail in the following sub-sections.

Home page. All web applications start with a home page. Most home pages welcomes the people on the web app and inform them about the web app they just visited. We created our home page as a static web page accordance to user friendliness and also to inform the public about our Project. We followed the modern web design trends and we created a slider with the universities we researched and also we created buttons with the universities logo that navigates to their website. In the bottom of the home page we present out team 3.

Graph Information page. The first dynamic page we will describe is the one that present the general metrics and the graphs that shows the current state of the graph-database we created 4. The first thing we see on the web page is the 3 cards with information such as i) Number of Universities, ii) Number of Professors, and iii) Number of Co-Authors. Those information are retrieved in real time from our database and shows the number of the data-nodes we collected and discussed in previous sections. After that, we present three cards with different color 4, each one shows a different number which is a metric for the graph about professors and co-authors and we discussed those metrics in previous section. The last thing we can find on this page is the tab list with the dynamic graphs. The graphs are divided into three tabs:

- The Professors Department Universities tab. This graph is presented on figure 4 and shows the first step of our project, which is the collection of the professors. So, in this graph each professor-node connects with a department-node which connects with a university-node. The red color of the nodes are the male professors and the green one are the female professors. The colors are dynamically assigned.
- The All Co-Authors. This graph describes the second step of our project, which was to find all co-authors of the professors in Google Scholar and determine their gender. So, in this graph we show each co-author-node which is connected with professor-node. In this graph we see some nodes to be larger than others, this means that this professor is more cited. The difference in color means the professor or the co-author is either male or female.
- Professor-Gender-CoAuthor. In order to use this tab it is a must to first select one of the two option on the radio buttons "Professor Gender" and "CoAuthor Gender". After the selection of the gender radio buttons we can select this tab and based on the selection it will show the proper graph. The smaller graph is the one of the Female professors and co-authors, after that is the Female professors with the Male co-authors, and the Male professor with female co-author is larger than the previous two graphs. The larger graph is the one with the Male professors and female co-authors. It needs some time to show the graph and that because it is dynamically created and is very large to calculate it. This shows the difference and he inequality of genders in tertiary education. Each time we select different gender on radio buttons we have to click again the tab in order to show the new graph.

In order to create the above graphs and the one we will discuss after, we used one tool that addresses some specific goals of graph visualization. This tool is Neovis.js[2] and is used for creating JavaScript based graph visualizations that are embedded in a web app. It uses the JavaScript Neo4j driver to connect to and fetch data from Neo4j and a JavaScript library for visualization called vis.js for rendering graph visualizations. Neovis.js can also leverage the results of graph algorithms like PageRank and community detection for styling the visualization by binding property values to visual components.

Universities page. The other utility we created was the University web page. In this we can see the list of the universities we examined. By clicking one university we can navigate to its professors name-buttons divided by the department they belong to as depicted in figure 5. We examined only computer science and electrical engineering departments. So, after choosing one professor, we get redirected to the professor's profile, figure 6. In this page we can see the data we retrieved from scholar about the professor on colored cards. Moreover we can see on a diagram near the cards his/her citations he/she got per year. Below that, there is a personalised graph about the professor and shows his/her co-authors. Again here the colors of the nodes are stand out from their gender and their size by their citations. By clicking on each node we can find all the information we retrieved from Google Scholar for this person.

Gender Metrics page. The last tool we developed is the "Gender Metrics". In this web-page we can find more specific metrics about the genders and that depicts the gender inequality. More specifically, the first thing we see are six check-boxes, that generates the top-k professors with the highest metrics of the checked boxes on the bottom of the page. For example , if we fill the "specific number of results" field with a number and select the boxes "Most Co-authors" and "PageRank score", the system will immediately will calculate those metrics and will return data such as, the name of the professors, their gender and their score with descending order as shown in figure 7. In the title of each data-table is described the percentage of each gender that can be found on the table bellow. Between the check boxes and the data tables we can see ten cards with metrics about the professors divided by their gender. We discussed about these metrics in previous section and show the gender inequality more clearly. The average metrics is an alternative way to see the gender balance instead of using the check-boxes and retrieve all the data on he front-end.

[2] https://github.com/neo4j-contrib/neovis.js/

7 Future Work

The creation of this website aims to give someone the chance to easily observe the gender balance and the differences that exist between men and women in the academic sector, in terms of equal representation. However, in order for the website to give a general picture of the gender inequality phenomenon, we consider that there are some additions and expansions to be made.

First of all, we consider that the *academic network should be expanded*, regarding the number of entities representing professors, but also regarding the information we hold for each one of them. As discussed in section 1.2, because of the limitation of time, we examined only 6 universities, and from those only the departments of Computer Science and Electrical and Computer Engineering. However, we could also examine both different departments of these universities, and other universities in Europe. We could also get more co-authors, not only those who are one hop away from our initial professors, but also those who are two and three hops away.

Furthermore, for each one of the faculty members, we could obtain even more information and from different sources, besides Google Scholar, such as DBLP or Research Gate. Except for the expansion of the network, using *multiple* sources for collecting the data gives a variety of *indices*, in addition to h-index and i10-index, the analysis of could provide useful insights about the gender inequality.

Finally, the network expansion could include an automated gender prediction model, in order to classify the incoming data regards to gender, and a component that will update the information collected for each faculty member after a short period of time, so as to be always up to date on any changes.

References

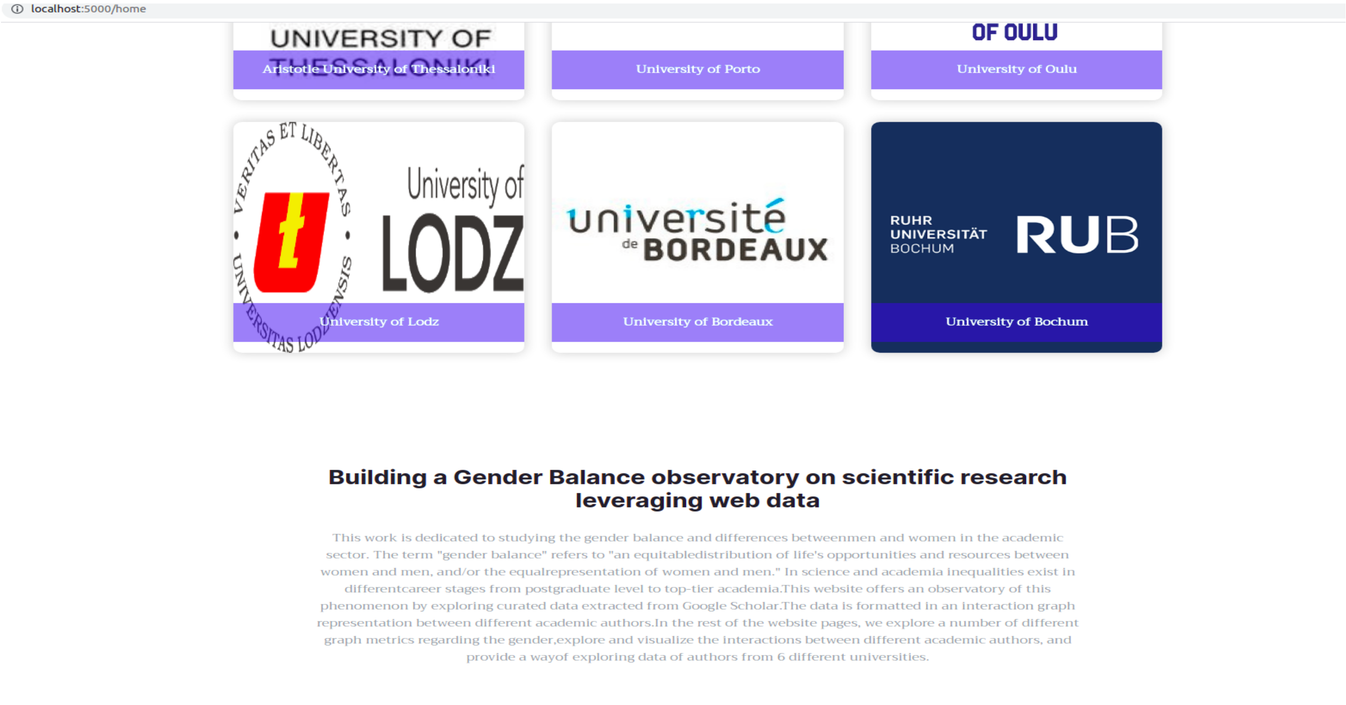
[1] Gender Equality Index 2019: Greece. page 6.

[2] She Figures 2013 - Gender in Research and Innovation - Data Europa EU.

[3] Gender inequality index.

[4] Gender comparisons.

Appendix: Figures



localhost:5000/home

Aristotle University of Thessaloniki

University of Porto

University of Oulu

University of Lodz

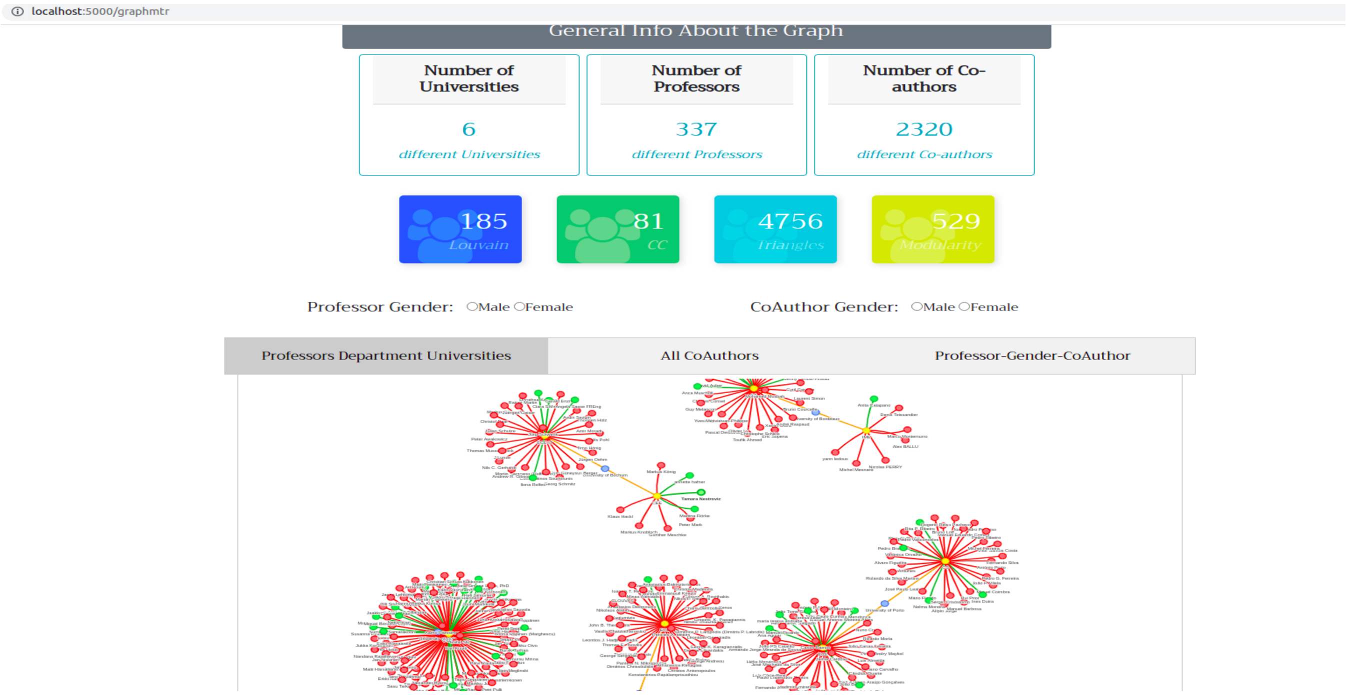
University of Bordeaux

University of Bochum

Building a Gender Balance observatory on scientific research leveraging web data

This work is dedicated to studying the gender balance and differences betweenmen and women in the academic sector. The term "gender balance" refers to "an equitabledistribution of life's opportunities and resources between women and men, and/or the equalrepresentation of women and men." In science and academia inequalities exist in differentcareer stages from postgraduate level to top-tier academia.This website offers an observatory of this phenomenon by exploring curated data extracted from Google Scholar.The data is formatted in an interaction graph representation between different academic authors.In the rest of the website pages, we explore a number of different graph metrics regarding the gender,explore and visualize the interactions between different academic authors, and provide a wayof exploring data of authors from 6 different universities.

Figure 3. Home Page



localhost:5000/graphmtr

General Info About the Graph

Number of Universities	Number of Professors	Number of Co-authors
6	337	2320
different Universities	*different Professors*	*different Co-authors*

185 *Louvain*

81 *CC*

4756 *Triangles*

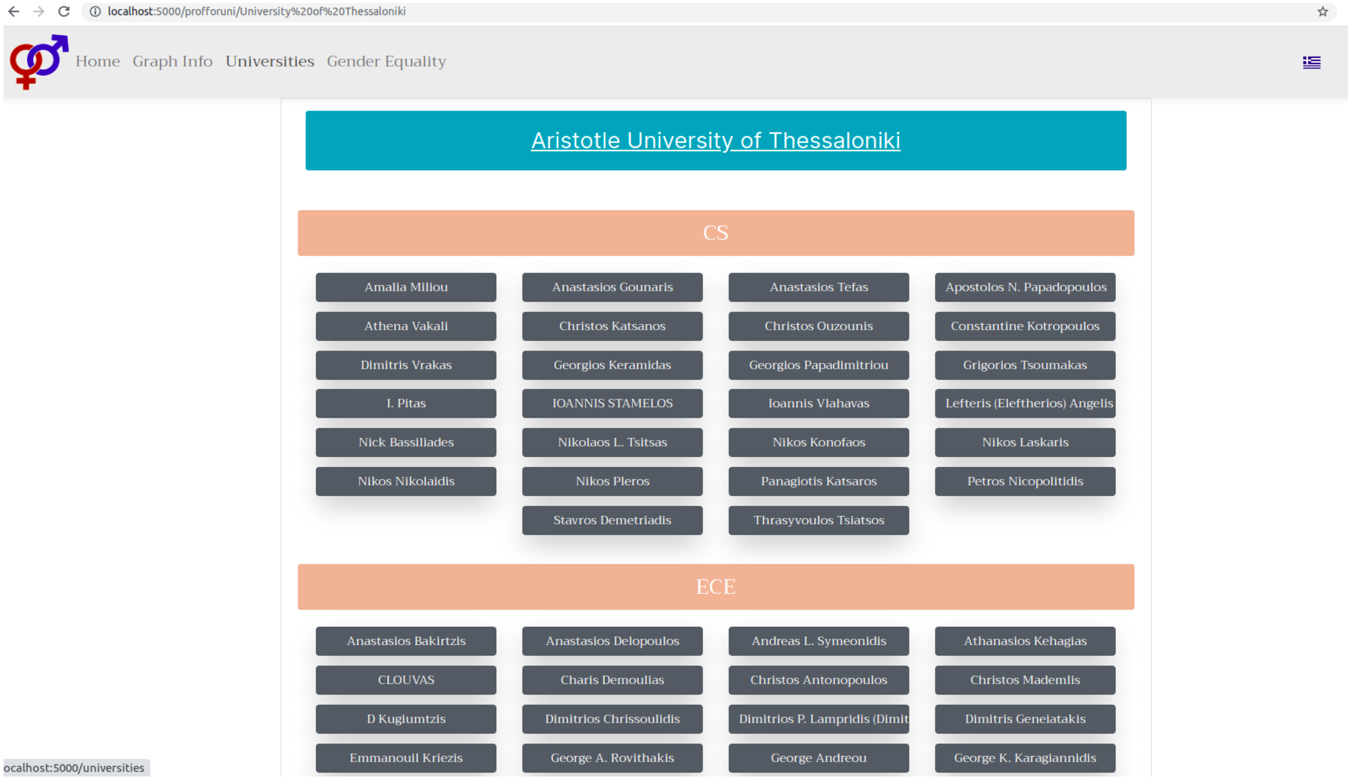
529 *Modularity*

Professor Gender: ○Male ○Female

CoAuthor Gender: ○Male ○Female

Professors Department Universities | All CoAuthors | Professor-Gender-CoAuthor

Figure 4. General Metrics & Graphs overview



localhost:5000/profforuni/University%20of%20Thessaloniki

Aristotle University of Thessaloniki

CS

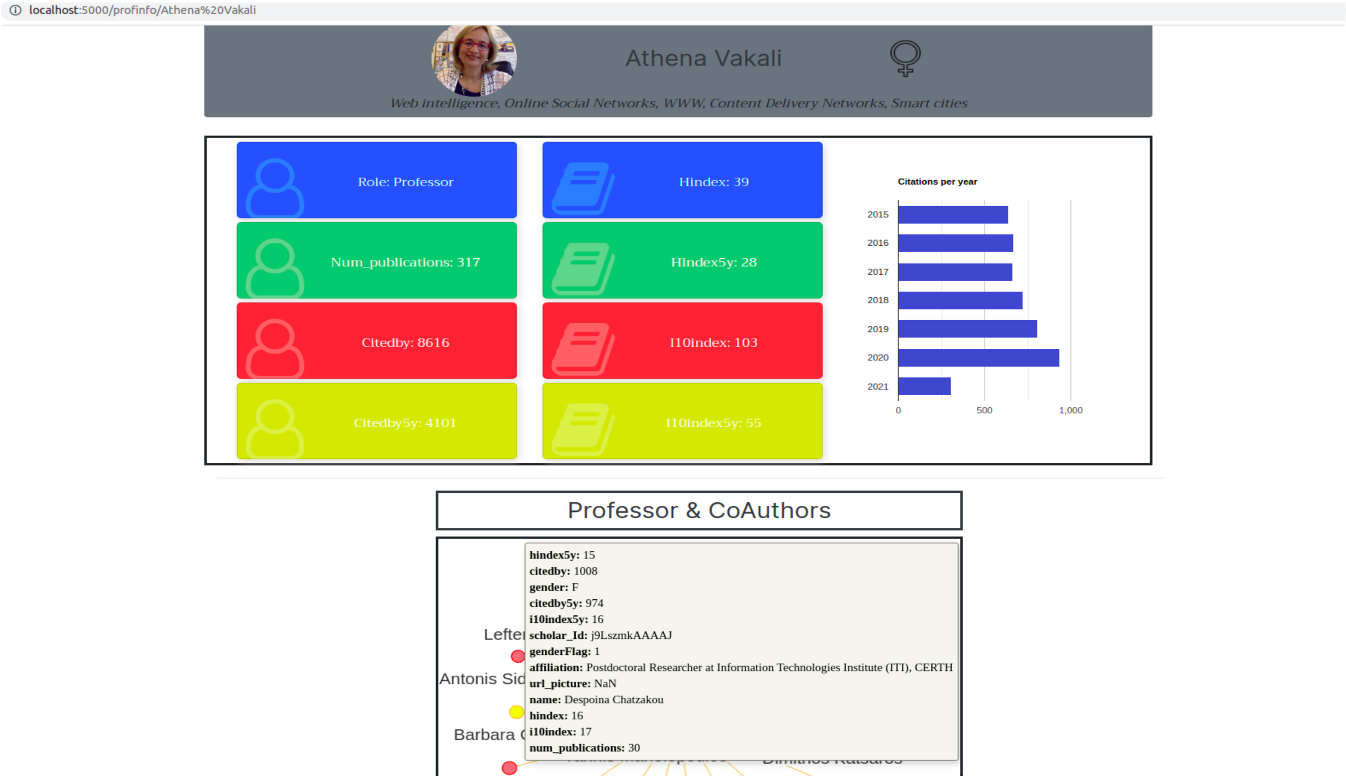
Amalia Miliou	Anastasios Gounaris	Anastasios Tefas	Apostolos N. Papadopoulos
Athena Vakali	Christos Katsanos	Christos Ouzounis	Constantine Kotropoulos
Dimitris Vrakas	Georgios Keramidas	Georgios Papadimitriou	Grigorios Tsoumakas
I. Pitas	IOANNIS STAMELOS	Ioannis Vlahavas	Lefteris (Eleftherios) Angelis
Nick Bassiliades	Nikolaos L. Tsitsas	Nikos Konofaos	Nikos Laskaris
Nikos Nikolaidis	Nikos Pleros	Panagiotis Katsaros	Petros Nicopolitidis
	Stavros Demetriadis	Thrasyvoulos Tsiatsos	

ECE

Anastasios Bakirtzis	Anastasios Delopoulos	Andreas L. Symeonidis	Athanasios Kehagias
CLOUVAS	Charis Demoulias	Christos Antonopoulos	Christos Mademlis
D Kugiumtzis	Dimitrios Chrissoulidis	Dimitrios P. Lampridis (Dimit	Dimitris Geneiatakis
Emmanouil Kriezis	George A. Rovithakis	George Andreou	George K. Karagiannidis

ocalhost:5000/universities

Figure 5. Departments of Aristotle University of Thessaloniki



localhost:5000/profinfo/Athena%20Vakali

Athena Vakali

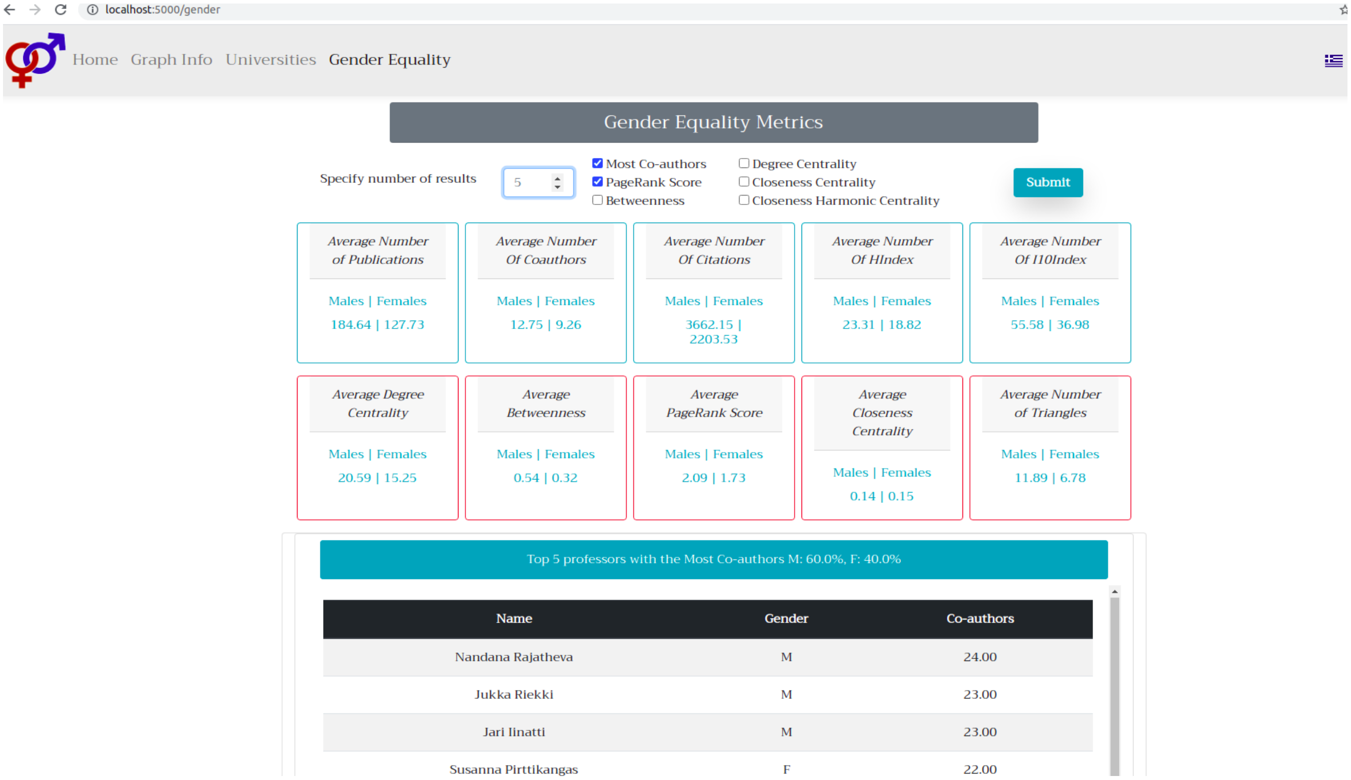
Web intelligence, Online Social Networks, WWW, Content Delivery Networks, Smart cities

Role: Professor	Hindex: 39
Num_publications: 317	Hindex5y: 28
Citedby: 8616	I10index: 103
Citedby5y: 4101	I10index5y: 55

Professor & CoAuthors

hindex5y: 15
citedby: 1008
gender: F
citedby5y: 974
i10index5y: 16
scholar_Id: j9LszmkAAAAJ
genderFlag: 1
affiliation: Postdoctoral Researcher at Information Technologies Institute (ITI), CERTH
url_picture: NaN
name: Despoina Chatzakou
hindex: 16
i10index: 17
num_publications: 30

Figure 6. Professor’s Profile



Gender Equality Metrics

Specify number of results 5

☑ Most Co-authors ☐ Degree Centrality
☑ PageRank Score ☐ Closeness Centrality
☐ Betweenness ☐ Closeness Harmonic Centrality

Submit

Average Number of Publications
Males | Females
184.64 | 127.73

Average Number Of Coauthors
Males | Females
12.75 | 9.26

Average Number Of Citations
Males | Females
3662.15 | 2203.53

Average Number Of HIndex
Males | Females
23.31 | 18.82

Average Number Of I10Index
Males | Females
55.58 | 36.98

Average Degree Centrality
Males | Females
20.59 | 15.25

Average Betweenness
Males | Females
0.54 | 0.32

Average PageRank Score
Males | Females
2.09 | 1.73

Average Closeness Centrality
Males | Females
0.14 | 0.15

Average Number of Triangles
Males | Females
11.89 | 6.78

Top 5 professors with the Most Co-authors M: 60.0%, F: 40.0%

Name	Gender	Co-authors
Nandana Rajatheva	M	24.00
Jukka Riekki	M	23.00
Jari Iinatti	M	23.00
Susanna Pirttikangas	F	22.00

Figure 7. Gender Metrics