International Journal of Progressive Sciences and Technologies (IJPSAT)
ISSN: 2509-0119.

http://ijpsat.ijsht-journals.org
Vol. 27 No. 2 July 2021, pp. 773-783

Development of a Decision Support System for Clustering Scientific Publications Using K-Means

Fauzan Bima Putra Kencana[1*], Rosihan Ari Yuana[1] and Nugroho Agung Pambudi[2]

[1,2]Pendidikan Teknik Informatika dan Komputer
Universitas Sebelas Maret
Jl. Ir. Sutami No.36, Kentingan, Kec. Jebres, Kota Surakarta, Jawa Tengah 57126

[2]Pendidikan Teknik Mesin
Universitas Sebelas Maret
Jl. Ir. Sutami No.36, Kentingan, Kec. Jebres, Kota Surakarta, Jawa Tengah 57126

Abstract— To improve the quality of departments within a university, a decision support system (DSS) that maps the number of scientific publications published is needed urgently today. This is because clusters and ranks for a university are determined by these scientific publications. In DSS, there is a crawler bot to recap publication data from Google Scholar and Scopus, as well as the K-means algorithm for clustering departments. DSS will produce 2 clusters with a silhouette score of 0.6245660527900866. A positive value indicates that the data in one cluster are similar. The sample used data from faculty members at the Faculty of Teacher Training and Education at Sebelas Maret University in Indonesia. The results showed that cluster one consisted of 18 departments. The average results were Scopus publication of 0.4669, Google scholar publications of 1.8955, Scopus citations of 0.8328, Google scholar citations of 18.3519. Cluster two consists of six departments., The average results were Scopus publication of 3.1727, Google Scholar publication of 4.2636, Scopus Citation of 9.0636, Google Scholar Citation of 67.6455.

Keywords— decision support system, bot crawler, K-means, google scholar, Scopus.

I. INTRODUCTION

Universities significantly contribute to the improvement of their human resources (HR) department [1]. Also, universities can improve the quality of their higher education by increasing the quality of scientific publications they release. In turn, this will aid in producing quality graduates. In Indonesia, the ministry of education and culture sorted universities into separate clusters. Furthermore, these Clusters and university rankings are used as indicators by international ranking organizations. The following indicators inform how the university will be ranked. These are the performance of inputs (20%) and processes (25%), outputs (25%), and outcomes (30%). The number of indexed scientific articles is 30% of the output indicators. Meanwhile, the number of citations is 20% of the outcome indicators.

The statistics for journals of higher education from Indonesia can be found on the SINTA website (http://sinta.ristekdikti.go.id/). The SINTA website sources its data from Scopus and Google Scholar. However, the data published on this website is often out of

Corresponding Author: Fauzan Bima Putra Kencana

date. To this day, the published data is compiled by the university, which is then carried out manually. Therefore, an automatic system in the form of a crawler bot is needed to facilitate the automatic collection of publications.

The use of a clustering algorithm can facilitate department mapping. One of the commonly used algorithms is K-means. The advantage of using a K-means algorithm is that it is easy to implement and relatively fast to learn. [2]. The next support tool that can be used is a decision support system (DSS). This is an information system that is used to support decision-making. It can provide information and provide various alternatives choices [3]. This will help in mapping study programs that are lacking in publications.

II. LITERATURE REVIEW

A. Decision Support System

A decision support system is an information system developed to support decision-making. Besides providing information, a DSS can provide various alternatives that can be taken in the decision-making process [3]. This system is only a supporting tool for making decisions. Therefore, it should not be used to replace decision-making.

According to Wibowo [4] the decision support system has the following criteria:

1. The Decision Support Systems are designed to assist decision-makers in solving semi-structured or unstructured problems by adding human wisdom and computerized information.
2. The decision support system combines the use of analytical models with conventional data entry techniques and information search/interrogation functions.
3. The Decision Support System is designed in such a way to be used/operated easily.
4. The Decision Support System is designed by emphasizing flexibility and high adaptability.

B. Beautiful Soup

Beautiful soup is a library that facilitates the collection of information on a web page. It supports HTML parsers, such as those provided in the python standard library and third-party parsers (i.g., lxml). [5]. Many methods can be used in this library (i.e., find() and find_all()).Furthermore, the search can also be carried out using the CSS attribute.

C. K-Means

The K-Means algorithm is one type of clustering in the unsupervised learning algorithm. It is popular because of its simplicity and low computational complexity [6]. This algorithm groups data based on the characteristics (i.e., same or different) of each group. This algorithm groups similar data samples into several parts to minimize the Euclidean distance between the data sample and the closest cluster. [7]

The k-means steps are as follows:

1. Specifies the number of clusters.
2. Determine the centroid or center point of each cluster.
3. Grouping data into clusters according to the nearest predetermined centroid distance.
4. Update the centroid point value and repeat step 3 until the centroid value does not change.

The following is one formula used to calculate the distance to the centroid, namely Minkowski [8]:

$$d(x_i, x_j) = \left(|x_{i1} - x_{j1}|^g + |x_{i2} - x_{j2}|^g + \ldots + |x_{ip} - x_{jp}|^g\right)^{1/g} \quad (1)$$

Where:

g = 1, to calculate the manhattan distance

g = 2, to calculate the euclidean distance

xi, xj = two data to be calculated the distance

p = dimensions of a data

K-means has the advantage of being easy to implement and efficient at data processing. On the other hand, a weakness of K-means is the curse of dimensionality. This weakness applies if the training data has high dimensions [9].

D. Crawling

Crawling is a technique of gathering information on the web [10]. This crawler tool is programmed with certain algorithm to gather information from the web. It works by scanning and reading the text on the web page. The purpose of crawling, or scraping, is to automize web data extraction [11]. Walther & Melsheimer [12] succeeded in conducting research and development in automatic processing of Scopus affiliate data at Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) using the crawling technique. Rose & Kitchin [13] also conducted research using pybliometrics which also uses the crawling method.

E. Scikit-learn

Scikit-learn is a machine learning library developed for the python programming language. It provides various well-known machine learning algorithms [14]. Furthermore, it can be used for supervised and unsupervised learning.

F. Scopus

Scopus is one of the standard citations, bibliometric, and abstract databases in the field of scientometrics and bibliometrics [13]. The database includes scientific journals, books, and conference proceedings. There are various types of research within Scopus such as science, technology, medicine, social sciences, and arts and humanities.

G. Google Scholar

The launch of Google Scholar in 2004 made it easy to search and access academic information [15]. With Google Scholar, various disciplines and sources of articles, books, theses, abstracts, online repositories, and others can be found.

III. Research Methods

This research uses published data from the Faculty of Teacher Training and Education, Sebelas Maret University in Indonesia. All of this data will be processed automatically by the bot crawler and presented to a website containing information on scientific publications and study program clustering.

A. Crawler Google Scholar

The Google Scholar crawling system uses the Scholarly Library with the Python programming language. In this research, the publication data of each faculty member were obtained with their Google Scholar ID. They were entered into the Scholarly function. The ID is obtained from the google scholar URL with the “get parameter user”, for example, https://scholar.google.co.id/citations?user=<ID>&hl=en. Scholarly produces output in the form of a list that will be stored in the database.

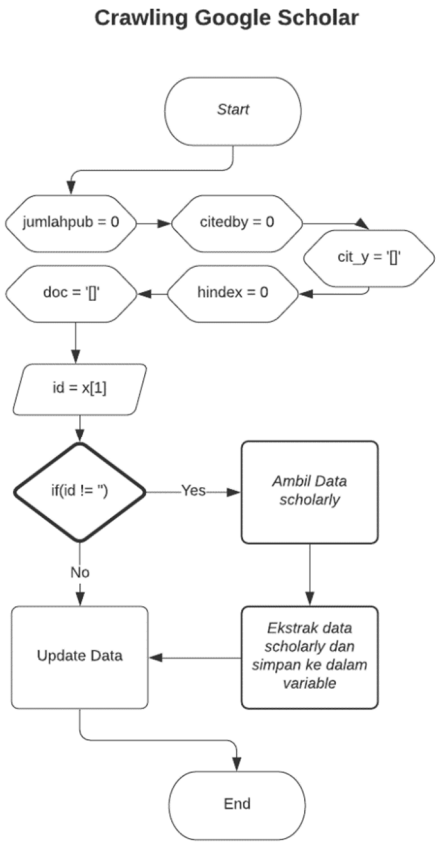

Fig. 1. Google scholar crawler flowchart

B. Crawler Scopus

Scopus has a special API that can be used to obtain data in its database. However, the API can only be accessed by accounts affiliated with Scopus. For purposes of this research, the researcher did not have this account.

Scopus website uses javascript to generate Html code. The use of python-requests cannot retrieve the Html code generated by the javascript. Therefore, it is necessary to use the Selenium Library to obtain the Html code. To find the data in the Html code, this research uses the beautifulsoup4 library based on the Html tags and attributes.

TABLE I. TAG TABLE AND SCOPUS ATTRIBUTE

Tags and Attributes	***Function***
id='warningMsgContainer'	Getting messages on the website
<h3 class='_1kYdpHTL20fs5Qk9XhXPVJ'>[0]	Getting the number of documents
<h3 class='_1kYdpHTL20fs5Qk9XhXPVJ'>[1]	Getting the number of citations
<h3 class='_1kYdpHTL20fs5Qk9XhXPVJ '>[2]	Getting Index

Tags and Attributes	*Function*
<class='ul--results'> <h5>	Getting document title
<class='ul--results'> <class='text-width-34'>[1]	Getting document year
<class='ul--results'> <class='text-width-34'>[0]	Getting source documents
<class='ul--results'> <class_='info-field__value'>	Getting the number of document citations

The system will retrieve data based on the Scopus ID of each lecturer. The ID is obtained from the URL with the "get parameter author" ID. Example https://www.scopus.com/authid/detail.uri?authorId=<ID>, <ID> is the ID used.

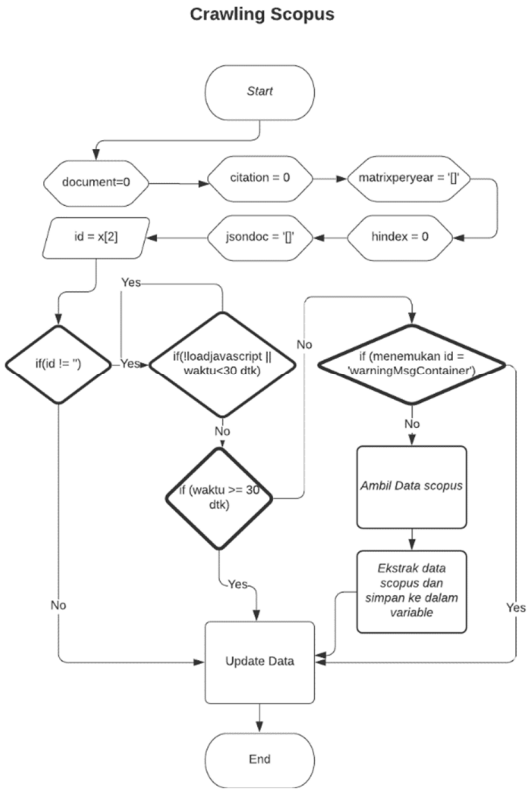

Fig. 2. Scopus crawling flowchart

In this system, the HTML code retrieval waits for the javascript code to load successfully. If the data does not load within 30 seconds, it will show an error message on the screen. If an error message does not appear, the website will be reloaded.

C. Department Clustering With K-means

The clustering process is carried out after the crawler has succeeded in retrieving all scientific publication data from each faculty members. This uses the K-means algorithm with the help of the python scikit-learn library. Furthermore, it produces 2 groups of study programs. The grouping is based on publication data and Google Scholar citations and the last year's Scopus.

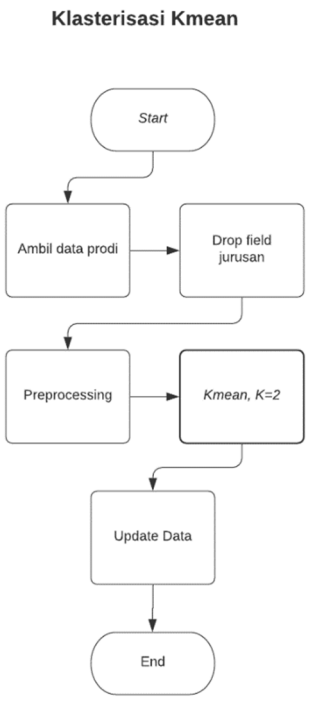

Fig. 3. Clustering Flowchart

The data used for clustering includes the average google scholar citation, the average google scholar publication, the average Scopus citation, and the average Scopus publication.

D. Entity Relational Diagram

The database consists of 6 tables, namely tbdosen, tbscholar, tbsopus, tb majors, tbfaculty, and the user.

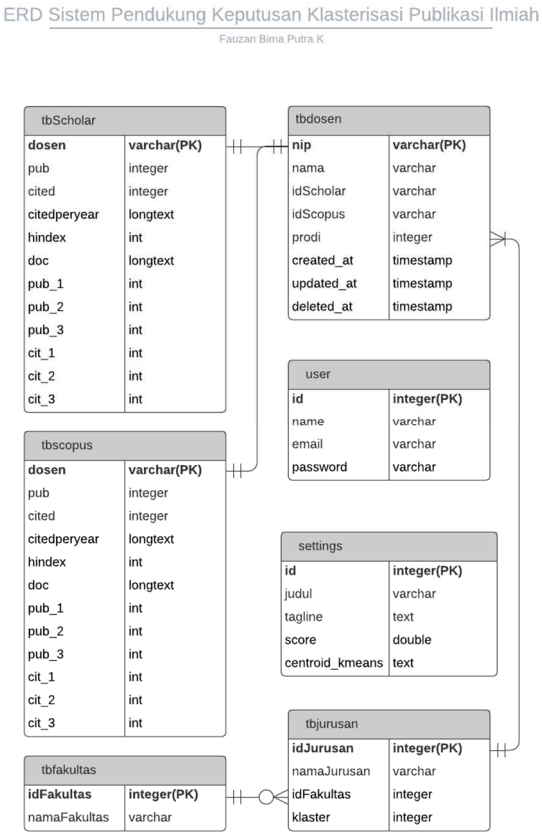

Fig. 4. ERD DSS Scientific Publications Cluster

IV. Results And Discussion

E. System Implementation

The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization). This template was designed for two affiliations.

F. Bot Crawler

The crawler bot is built using the Python 3.6 programming language. There are some changes in the scholarly library due to a bug found.

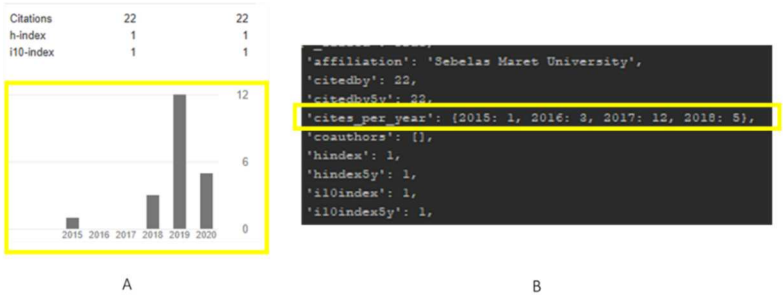

Fig. 5. Citation per year of google scholar(A) and scholarly(B) websites

Figure 5 shows that the 2018 citation data on the website is read as 2016 data on the Scholarly library. The library cannot read the value 0 in the 2016 and 2017 citations on the website.

```
years = [int(y.text) for y in soup.find_all('span', class_='gsc_g_t')]
cites = [int(c.text) for c in soup.find_all('span', class_='gsc_g_al')]
self.cites_per_year = dict(zip(years, cites))
```

Fig. 6. The original code for citations per year scholarly

To solve this problem, it is necessary to find the difference in the value of style right class=”gsc_g_t” with class=”gsc_g_a”. If the difference in value is equal to 5, the two values are pairs of years and citation values.

Fig. 7. Structure of html citations per year

```
yearsright = [int(re.search(r'\d+', y['style']).group()) for y in soup.find_all('span', class_='gsc_g_t')]

cites = [int(c.text) for c in soup.find_all('span', class_='gsc_g_al')]
citesright = [int(re.search(r'\d+', c['style']).group()) for c in soup.find_all('a', class_='gsc_g_a')]

citesperyear = []
for cy in yearsright:
    citesperyear.append(0)

i = 0
for c in citesright:
    j = 0
    for cy in yearsright:
        if c - cy == 5:
            citesperyear[j] = cites[i]
            break
        j += 1
    i += 1
```

Fig. 8. Citation code improvements per year

When retrieving data from Scopus, the program must wait for the javascript code to load successfully. . The process aims to wait for javascript to generate all the html code. If it fails to load within that time frame, the no id='warningMsgContainer', the process will be repeated. The process is successful if it successfully loads the class 'reactive-chart-container', '_2Y8rC6tZxrdhz4UUWoh3gQ', 'ul--results'. The class is sequentially a division containing the annual publication matrix, the overall publication matrix, and document data.

```
driver.get("https://www.scopus.com/authid/detail.uri?authorId=" + idscopus)

    try:
        WebDriverWait(driver, 30).until(
            EC.presence_of_element_located((By.CLASS_NAME, 'reactive-chart-container')) and
            EC.presence_of_element_located((By.CLASS_NAME, '_2Y8rC6tZxrdhz4UUWoh3gQ')) and
            EC.presence_of_element_located((By.CLASS_NAME, 'ul--results'))
        )
    except TimeoutException:
        print("Scopus Gagal : " + idscopus)
    html2 = driver.execute_script("return document.documentElement.innerHTML;")
```

Fig. 9. The process of loading javascript

Using the selenium library can help programs act like normal users because Scopus has protection against bot activity. In this research, selenium was run with the help of a web driver, namely the chrome driver. Web driver is created in headless mode therefore in the process no chrome window is open.

```
option = webdriver.ChromeOptions()
option.add_argument('--headless')
UserAgent = 'Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N)
             AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.116 Mobile Safari/537.36'
option.add_argument(f'user-agent={UserAgent}')
driver = webdriver.Chrome('chromedriver.exe', chrome_options=option)
```

Fig. 10. Setting headless selenium

The crawling system on Scopus has a weakness. If the Scopus website's structure changes, the tags and css selectors used may change. Therefore, it is necessary to make adjustments to the program to reflect the structure of the Scopus website. These problems can be solved by regularly monitoring the Scopus website.

G. Clustering

The clustering process is carried out after all lecturer publication data has been successfully updated. The program will retrieve data from the database in the form of a pandas data frame. The data taken from the database include Department ID, average publication, and average citation from Google Scholar and Scopus. Furthermore, preprocessing is carried out by transforming it into a range of 0 to 1. The purpose is to do preprocessing therefore the data can be processed easily.

```
02.
03. data = pd.DataFrame(q)
04. data.fillna(0, inplace=True)
05. data.head()
06. x = np.asarray(data)
07.
08. scaler = MinMaxScaler()
09. x_scaled = scaler.fit_transform(x)
10.
11. kmeans = KMeans(n_clusters=2)
12. kmeans.fit(x_scaled)
```

Fig. 11. Preprocessing code

After the clustering is complete then the results are updated to the database. The results are in the form of cluster classes and are displayed on a website.

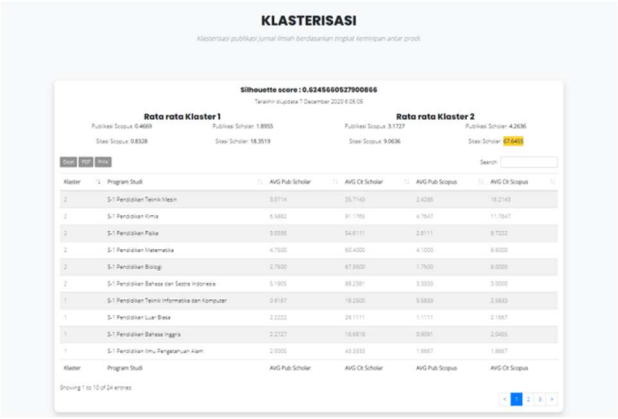

Fig. 12. DSS Website

H. System Trial

The trial was conducted using 397 sample faculty member data. Data from publications and Google Scholar citations were from 2020. Clustering uses data on the average number of Google Scholar publications, the average number of Scopus publications, the average Google Scholar citation, and the average Scopus citation in 2020. The level of clustering similarity is indicated by a silhouette score of 0.6245660527900866. The positive value in the silhouette score indicates that the data in one cluster have similarities. Cluster 1 has a Scopus publication average of 0.4669, Google scholar publications of 1.8955, Scopus citations of 0.8328, Google scholar citations of 18.3519. Meanwhile, cluster 2 has an average Scopus publication of 3.1727, Google scholar publications of 4.2636, Scopus citations of 9.0636, Google scholar citations of 67.6455. Based on these results, it can be concluded that cluster 2 is better than cluster 1. The results of clustering can be seen in table 2.

TABLE II. HASIL KLASTERISASI

Klaster	*Program Studi*
1	S-1 Bimbingan dan Konseling
1	S-1 Pendidikan Administrasi Perkantoran
1	S-1 Pendidikan Akuntansi
1	S-1 Pendidikan Bahasa Inggris
1	S-1 Pendidikan Bahasa Jawa
1	S-1 Pendidikan Ekonomi
1	S-1 Pendidikan Geografi
1	S-1 Pendidikan Guru Pendidikan Anak Usia Dini
1	S-1 Pendidikan Guru Sekolah Dasar
1	S-1 Pendidikan Guru Sekolah Dasar Kebumen
1	S-1 Pendidikan Ilmu Pengetahuan Alam
1	S-1 Pendidikan Luar Biasa
1	S-1 Pendidikan Pancasila dan Kewarganegaraan
1	S-1 Pendidikan Sejarah
1	S-1 Pendidikan Seni Rupa
1	S-1 Pendidikan Sosiologi Antropologi
1	S-1 Pendidikan Teknik Bangunan
1	S-1 Pendidikan Teknik Informatika dan Komputer
2	S-1 Pendidikan Bahasa dan Sastra Indonesia
2	S-1 Pendidikan Biologi
2	S-1 Pendidikan Fisika
2	S-1 Pendidikan Kimia
2	S-1 Pendidikan Matematika
2	S-1 Pendidikan Teknik Mesin

V. CONCLUSION

The conclusions that can be drawn from this research are as follows:

1. The Scholarly library can simplify the process of crawling Google Scholar data. Researchers should monitor any changes to the library to keep the citation data retrieval tool up-to-date. Meanwhile, the Scopus data crawling process can use the selenium and beautifulsoup libraries. The Scikit learn library can simplify the clustering process using the K-means algorithm.
2. Cluster 1 consists of 18 departments including Guidance and Counseling, Office Administration Education, Accounting Education, English Education, Javanese Language Education, Economic Education, Geography Education, Early

Childhood Teacher Education, Elementary School Teacher Education, Elementary School Teacher Education Kebumen, Natural Sciences Education, Special Education, Pancasila and Citizenship Education, History Education, Fine Arts Education, Sociology Anthropology Education, Building Engineering Education, and Informatics and Computer Engineering Education. Cluster 2 consists of 6 bachelor study programs, including Indonesian Language and Literature Education, Biology Education, Physics Education, Chemistry Education, Mathematics Education, and Mechanical Engineering Education. Cluster 1 has a Scopus publication average of 0.4669, Google scholar publications of 1.8955, Scopus citations of 0.8328, Google scholar citations of 18.3519. Meanwhile, cluster 2 has an average Scopus publication of 3.1727, Google scholar publications of 4.2636, Scopus citations of 9.0636, Google scholar citations of 67.6455. The K-means silhouette score is 0.6245660527900866. This positive value indicates that the data in one cluster have similarities.

ACKNOWLEDGMENT

The authors thanks LPPM UNS for research funding with the title project: Decision Support System for Clustering Scientific Publications Using K-Means at FKIP UNS. Contract number: 260/UN27.22/HK.07.00/2021

REFERENCES

[1] M. F. Soni Akhmad Nulhaqim, R. Dudy Heryadi, Ramadhan Pancasilawan, "Peranan Perguruan Tinggi Dalam Meningkatkan Kualitas Pendidikan Di Indonesia Untuk Menghadapi Asean Community 201533," Univ. Padjadjaran, vol. 6, p. 198, 2015.

[2] S. Russell and P. Norvig, Artificial Intelligence A Modern Approach Third Edition. 2010.

[3] Y. Widiastuti, S. W. Sihwi, and M. E. Sulistyo, "Decision Support System for House Purchasing Using Knn (K-Nearest Neighbor) Method," J. Itsmart, vol. 5, no. 1, pp. 43–49, 2016.

[4] Wibowo, Perancangan Sistem Pendukung Keputusan. Depok, 2011.

[5] L. Richardson, "Beautiful Soup Documentation Release 4.4.0," Media.Readthedocs.Org, pp. 1–72, 2019.

[6] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability, 1967, vol. 1, pp. 281–296.

[7] C. D. Nguyen and T. H. Duong, "K-means** - a fast and efficient K-means algorithms," Int. J. Intell. Inf. Database Syst., vol. 11, no. 1, p. 27, 2018.

[8] O. Maimon and L. Rokach, Data Mining and Knowledge Discovery Handbook. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[9] T. Wahyono, Fundamental Of Python For Machine Learning. Yogyakarta: Gava Media, 2018.

[10] A. Priyanto and M. R. Ma'arif, "Implementasi Web Scrapping dan Text Mining untuk Akuisisi dan Kategorisasi Informasi dari Internet (Studi Kasus: Tutorial Hidroponik)," Indones. J. Inf. Syst., vol. 1, no. 1, pp. 25–33, 2018.

[11] V. Singrodia, A. Mitra, and S. Paul, "A Review on Web Scrapping and its Applications," in 2019 International Conference on Computer Communication and Informatics, ICCCI 2019, 2019.

[12] M. Walther and B. Melsheimer, "Automated author affiliation processing using Scopus data," Procedia Comput. Sci., vol. 146, pp. 53–59, 2019.

[13] M. E. Rose and J. R. Kitchin, "pybliometrics: Scriptable bibliometrics using a Python interface to Scopus," SoftwareX, vol. 10, p. 100263, 2019.

[14] G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller, "Scikit-learn," GetMobile Mob. Comput. Commun., vol. 19, no. 1, pp. 29–33, 2015.

[15] A. Martín-Martín, E. Orduna-Malea, M. Thelwall, and E. Delgado López-Cózar, "Google Scholar, Web of Science, and Scopus: A systematic comparison of citations in 252 subject categories," J. Informetr., vol. 12, no. 4, pp. 1160–1177, 2018.