

Proyecto Final Primera Entrega

Lisseth Tatiana Quilindo Patiño
Fundación Universitaria Konrad Lorenz

Resumen—En el presente artículo explicamos los pasos a seguir para la búsqueda de gestión de documentos, partiendo de 38.427 resultados de búsqueda en la base de datos de Scopus, las referencias bibliográficas, graficas de las palabras mas relevantes encontradas y en Python reduciendo la búsqueda en 32 artículos de investigación el flujo grama de los mismos.

I. INTRODUCCIÓN

Para facilitar la búsqueda de artículos en la base de datos de Scopus se pueden usar herramientas como Python, en ella podemos reducir las búsquedas de muchos artículos como por ejemplo para el proyecto "Gestión de Documentos" con una inicial de 38.427 resultados, se pueden reducir a 32 resultados de búsqueda dejando los mas importantes para el proyecto de la investigación.

II. RESULTADOS

Como resultados se obtuvo que se pueden tener resultados mas rapidos con la estrategia de Python obteniendo resultados mas acordes a los temas como por ejemplo en la búsqueda de información para el proyecto "Gestión de Documentos." la hora de buscar un artículo para la base de datos muy extensas.

II-A. Flujo del Analisis

Los pasos para la búsqueda en Scopus se reducen a:

1. Buscar en el Scopus (Gestión de documentos) con 11 resultados.
2. Cambiar la búsqueda en inglés (Document management system) con 38.427 resultados.
3. Filtrar la búsqueda entre el año 2020 y el 2023, área en ingeniería con 1.666 resultados.
4. Analizar en Python y escoger las palabras mas relevantes.
5. Incluir en la búsqueda de Scopus las nuevas palabras recolectadas de Python con 32 resultados.
6. Seleccionar los artículos mas relevantes para la investigación.

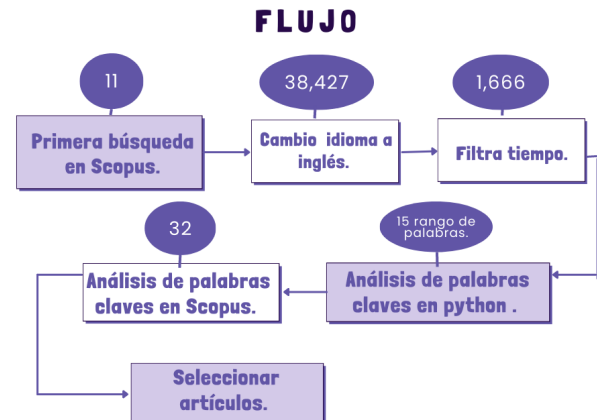


Figura 1. Diagrama de flujo búsqueda Scopus.

II-B. Query Exploratoria

Se inicia con una búsqueda en Scopus sobre el tema "gestión documental", escrito en inglés "Document management system" se filtran los temas de interés, fechas, área.

```

TITLE-ABS-KEY ( ( document AND management AND system ) ) AND ( LIMIT-TO ( PUBYEAR , 2023 ) OR LIMIT-TO ( PUBYEAR , 2022 ) OR LIMIT-TO ( PUBYEAR , 2021 ) OR LIMIT-TO ( PUBYEAR , 2020 ) ) AND ( LIMIT-TO ( SUBJAREA , "ENGI" ) )
  
```

Figura 2. Query Exploratoria Scopus primera búsqueda.

II-C. Analisis de Titulo

Se muestran las imágenes de las palabras mas relevantes encontradas en Python mediante google astrad por el "titles.ipynb" para el título mostrada en forma de grafica de barras de las "top", donde la mas relevante es management.

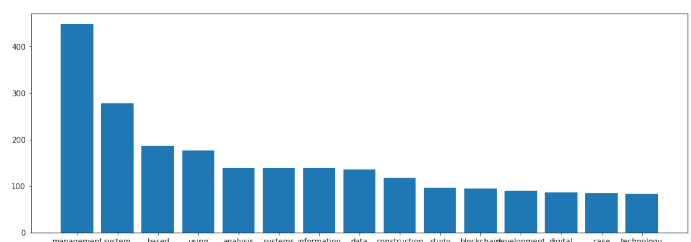


Figura 3. Palabras mas relevantes.

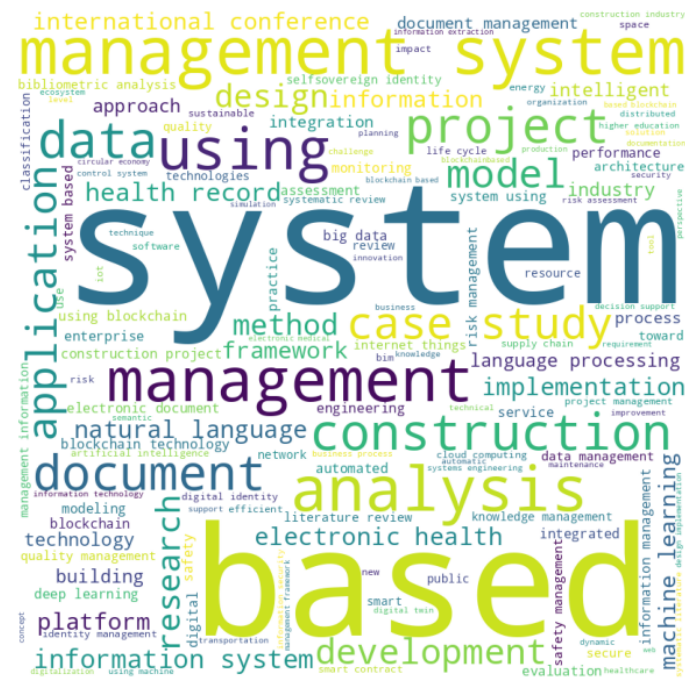


Figura 4. Las palabras van por tamaños entre mas grande sera mas relevantes.

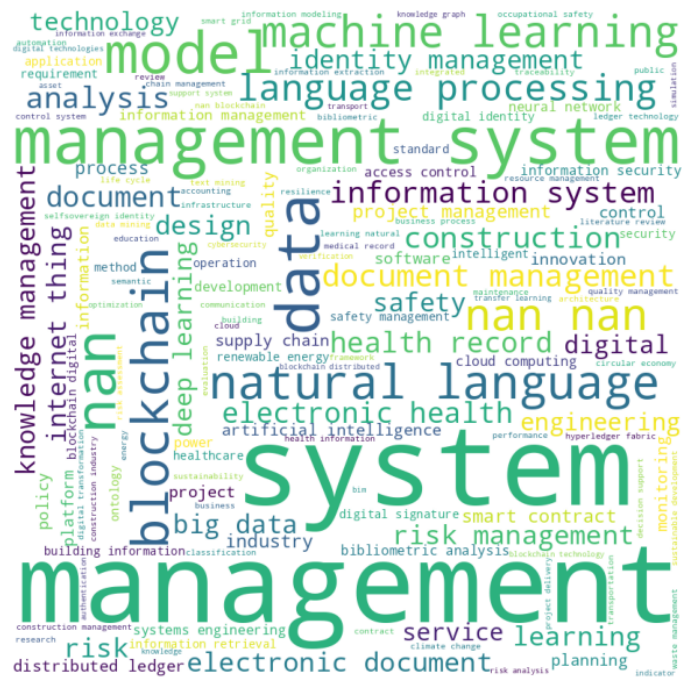


Figura 6. Muestra las palabras con más usadas en el Keyword de los artículos de scopus.

II-D. Analisis de Keyword

Este es el contenido del analisis del Keyword. se puede observar una grafica la cual tiene como objetivo mostrar el indice de mayor uso de una palabra en el Keyword de los articulos de Scopus.

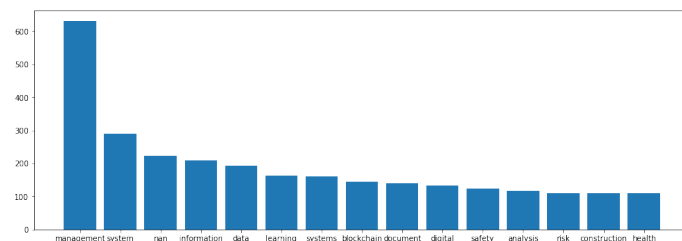


Figura 5. Descripción de la imagen.

II-E. Analisis de Abstract

Este es el contenido del analisis del Abstract se puede apreciar la grafica que muestra las palabra de los articulos de Scopus mas usadas.

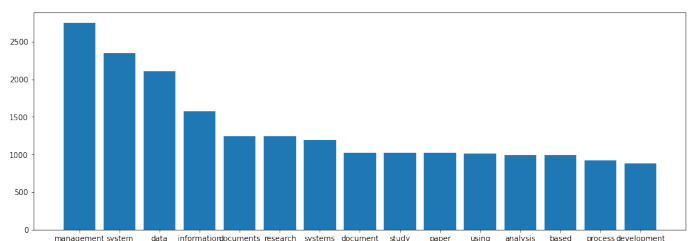


Figura 7. Palabras mas relevantes.



Figura 8. Las palabras van por tamaños entre mas grande sera mas relevantes.

II-H. RESULTADOS

Comparativa en espacios de tiempo.

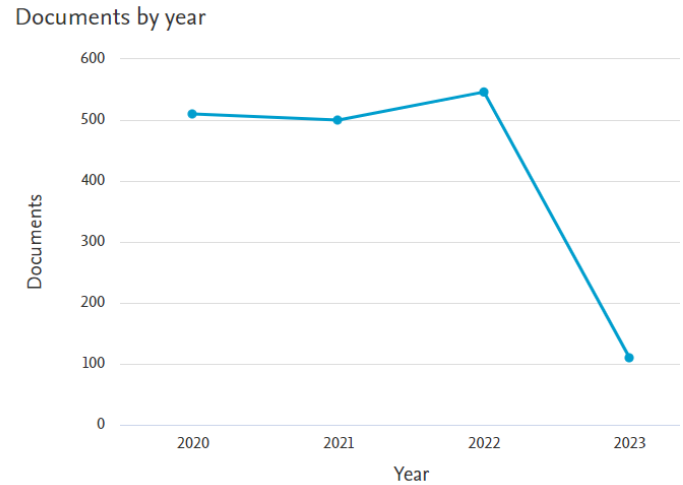


Figura 10. grafica con Query.

II-F. Ranking de palabras clave para la construction del nuevo filtro

Teniendo en cuenta las nuevas palabras encontradas en el titulo, Keyword y Abstract se añaden a el nuevo Query de busqueda en el Scopus obteniendo asi una un Query resultante mas completa para el proyecto "gestón de documentos".

II-G. Query Resultante

TITLE-ABS-KEY ((document AND management AND system)) AND (construction) AND (blockchain) AND (using) AND (development) AND (LIMIT-TO (SUBJAREA , "ENG")) AND (LIMIT-TO (PUBYEAR , 2023) OR LIMIT-TO (PUBYEAR , 2022) OR LIMIT-TO (PUBYEAR , 2021) OR LIMIT-TO (PUBYEAR , 2020))

Figura 9. Query Exploratoria Scopus busqueda final.

Este es el query resultante.

Este es el query resultante.

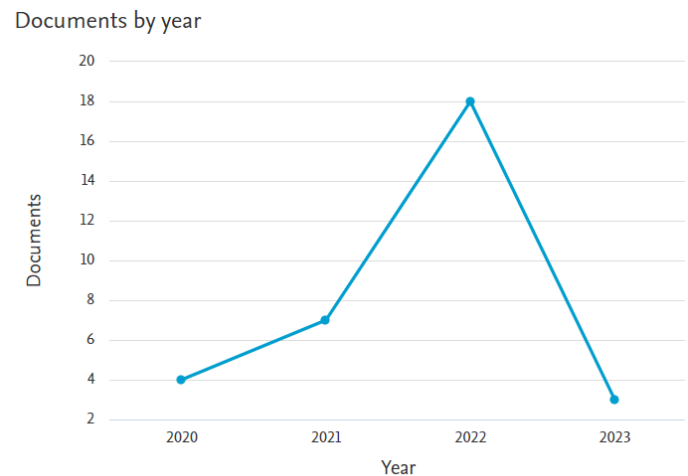


Figura 11. grafica con Query nuevo.

Este es el query resultante.

II-I. RESULTADOS

Comparativa en espacios de tiempo.

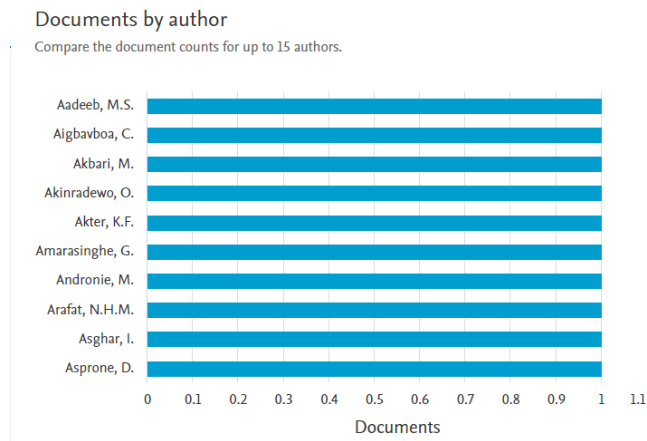


Figura 12. grafica de autores mas relevantes.

Este es el query resultante.

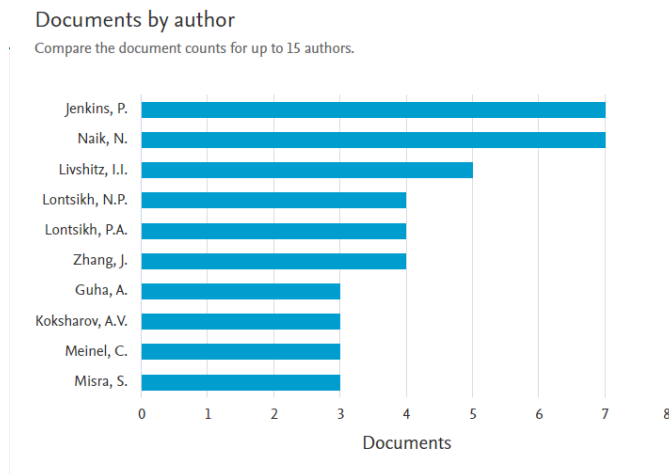


Figura 13. grafica autores mas relevantes nuevo Query.

II-J. RESULTADOS

Categorías Principales Encontradas

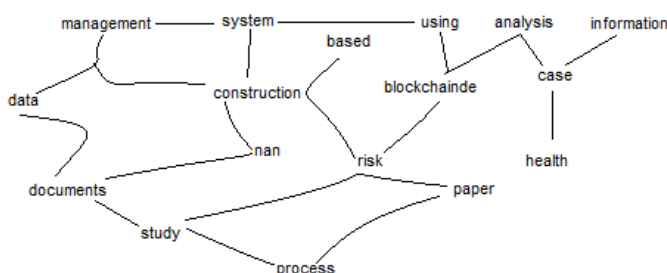


Figura 14. grafica palabras mas relevantes por Python.

III. CONCLUSIONES

Se puede concluir que el proceso para la búsqueda de palabras claves o relevantes fue mas conciso y relevante con Python ayudando a la clasificar los articulos con base a el nuevo Query de busqueda.

IV. REFERENCIAS

Cabrera, O., Franch, X., Marco, J. (2017). Ontology-based context modeling in service-oriented computing: A systematic mapping. *Data Knowledge Engineering*, 110, 24-53.

Petticrew, M., Roberts, H. (2008). *Systematic reviews in the social sciences: A practical guide*. John Wiley Sons.

Kitchenham, B., Brereton, P. (2013). A systematic review of systematic review process research in software engineering. *Information and software technology*, 55(12), 2049-2075.

OpenAI. (2021). OpenAI API. Recuperado el 23 de marzo de 2023, de <https://chat.openai.com/auth/login>

Mendez, O. (2023, 6 de marzo). Bibliography [Repositorio de GitLab]. Recuperado el 23 de marzo de 2023, de <https://gitlab.com/SoloTalleres/bibliography/-/tree/main>.