



Fundación Universitaria Konrad Lorenz  
Facultad de matemáticas e ingenierías  
Diseño de interfaces de usuario

---

## Taller de Bases de Datos Análisis Bibliográfico

---

*Alumno 1:*

Andrea Valentina Cubillos Pinto  
506231711

*Alumno 2:*

Ivonne Nathalia Sierra Ramírez  
506221064

*Alumno 3:*

Santiago Lozano Orrego  
506221033

*Profesor:*

Oscar Alexander Méndez Aguirre

6 de junio de 2024

# Índice

<b>1. Descripción del tema y justificación de la elección</b>	<b>4</b>
<b>2. Metodología aplicada</b>	<b>5</b>
2.1. Proceso de búsqueda . . . . .	5
2.2. Métodos de análisis . . . . .	9
<b>3. Resultados de análisis</b>	<b>10</b>
3.1. Análisis de Keywords . . . . .	10
3.2. Análisis de Abstract . . . . .	11
3.3. Análisis de Title . . . . .	11
3.4. Result Query . . . . .	12
3.5. Resultados: comparativas en espacios de tiempo . . . . .	13
3.6. Documentos por autor . . . . .	14
3.7. Categorías principales . . . . .	14
<b>4. Discusión de los resultados</b>	<b>15</b>
<b>5. Conclusiones y recomendaciones</b>	<b>16</b>
<b>Bibliografía</b>	<b>17</b>

## Índice de figuras

1.	Tema seleccionado . . . . .	4
2.	Ingreso a Scopus. . . . .	5
3.	Query avanzado. . . . .	5
4.	Refinar búsqueda por año. . . . .	6
5.	Refinar búsqueda por tipo de documento y área. . . . .	7
6.	Refinar búsqueda por idioma y palabra clave. . . . .	8
7.	Query exploratorio-parte 1. . . . .	9
8.	Query exploratorio-parte 2. . . . .	9
9.	Gráfica de análisis de Keywords. . . . .	10
10.	Nube del análisis de Keywords. . . . .	11
11.	Gráfica de análisis de abstract. . . . .	11
12.	Nube de palabras de análisis de abstract. . . . .	11
13.	Gráfica de análisis de title. . . . .	12
14.	Nube de palabras de análisis de title. . . . .	12
15.	Auge de categorías analizadas conceptuales al 2018. . . . .	13
16.	Auge de categorías analizadas conceptuales al 2025. . . . .	14
17.	Documentos por autor. . . . .	14
18.	Categorías principales. . . . .	15

# 1. Descripción del tema y justificación de la elección



Figura 1: Tema seleccionado

La arquitectura de software es un área de estudio y práctica que se enfoca en la estructuración, diseño y organización de los componentes y módulos que conforman un sistema de software. Su objetivo principal es definir una estructura sólida y coherente que permita el desarrollo, mantenimiento y evolución efectiva del sistema a lo largo de su ciclo de vida.

La importancia de este tema radica en su impacto directo en la calidad, escalabilidad, mantenibilidad y rendimiento de los sistemas de software desarrollados. Una arquitectura bien diseñada facilita la comunicación entre los diferentes equipos involucrados, promueve la reutilización de componentes, y guía las decisiones de diseño e implementación de manera coherente. Por tanto, se ha seleccionado la arquitectura de software como tema central de nuestro análisis bibliográfico debido a varias razones fundamentales:

- **Relevancia en la industria del software:** La arquitectura de software es un pilar esencial en el desarrollo de sistemas complejos y de alta calidad, lo que la convierte en un tema de suma importancia en la industria del software actual.
- **Impacto en el éxito de los proyectos:** Una arquitectura sólida y bien estructurada es un factor determinante para el éxito de los proyectos de software, ya que influye directamente en la capacidad de cumplir con los requisitos, la escalabilidad, el mantenimiento y la evolución futura del sistema.
- **Diversidad de enfoques y patrones:** La arquitectura de software abarca una amplia gama de enfoques, patrones y principios de diseño, lo que nos permite analizar y comparar diferentes perspectivas y prácticas en el campo.
- **Importancia en la formación de profesionales:** Comprender los fundamentos de la arquitectura de software es crucial para la formación de arquitectos de software, desarrolladores y otros profesionales involucrados en el ciclo de vida del desarrollo de sistemas.
- **Evolución constante:** A medida que surgen nuevas tecnologías, paradigmas y requisitos, la arquitectura de software debe adaptarse y evolucionar, lo que la convierte en un área de investigación y estudio continuo.

En sí, al realizar un análisis bibliográfico exhaustivo sobre la arquitectura de software, se puede comprender los conceptos fundamentales, los enfoques y patrones arquitectónicos más relevantes, las mejores prácticas, las tendencias actuales y las áreas de oportunidad para futuras

investigaciones y desarrollos. Esto nos brinda una base sólida para abordar de manera efectiva los desafíos y necesidades de la industria del software en relación con la arquitectura de los sistemas.

## 2. Metodología aplicada

### 2.1. Proceso de búsqueda

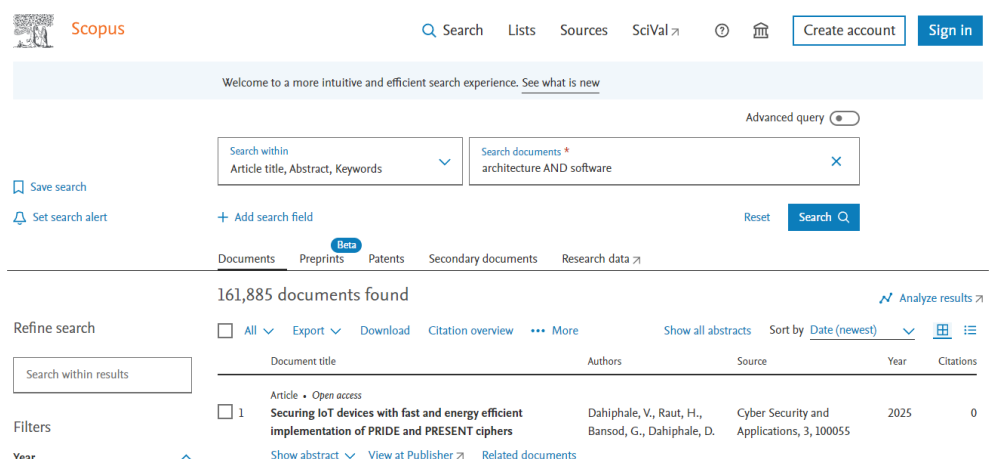


Figura 2: Ingreso a Scopus.

Se ingresa a la base de datos 'Scopus'.



Figura 3: Query avanzado.

En el apartado de búsqueda se escribe el tema seleccionado y, posteriormente, se da clic en la opción de ver la consulta avanzada.

Refine search

Search within results

Filters

Clear all

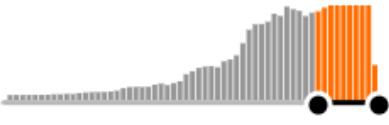
Year

Clear

^

☒ Range

☐ Individual



2015

–

2025

Subject area

Clear (3)

^

☒ Limited to Computer

48,279

Cancel

Exclude

Limit to

Figura 4: Refinar búsqueda por año.

Inicialmente, se limita la búsqueda por año.

The image shows a search refinement interface. At the top, there is a section titled 'Subject area' with a 'Clear (3)' link and an upward arrow. Below this, there is a list of subject areas with checkboxes and counts. The first three are selected: 'Computer Science' (48,279), 'Engineering' (29,025), and 'Mathematics' (12,508). The next three are not selected: 'Physics and Astronomy' (5,130), 'Decision Sciences' (4,804), and 'Materials Science' (3,655). The last one is 'Social Sciences' (2,844). Below the list is a 'Show all' link. Below the subject area section is a section titled 'Document type' with an upward arrow. Below this, there is a list of document types with checkboxes and counts. The first three are selected: 'Conference paper' (35,012), 'Article' (19,386), and 'Book chapter' (1,695). The last one is 'Conference review' (1,490). At the bottom of the interface are three buttons: 'Cancel', 'Exclude', and 'Limit to'.

Subject area	Count
<input checked="" type="checkbox"/> Limited to Computer Science	48,279
<input checked="" type="checkbox"/> Limited to Engineering	29,025
<input checked="" type="checkbox"/> Limited to Mathematics	12,508
<input type="checkbox"/> Physics and Astronomy	5,130
<input type="checkbox"/> Decision Sciences	4,804
<input type="checkbox"/> Materials Science	3,655
<input type="checkbox"/> Social Sciences	2,844

[Show all](#)

Document type	Count
<input checked="" type="checkbox"/> Conference paper	35,012
<input checked="" type="checkbox"/> Article	19,386
<input checked="" type="checkbox"/> Book chapter	1,695
<input type="checkbox"/> Conference review	1,490

[Cancel](#) [Exclude](#) [Limit to](#)

Figura 5: Refinar búsqueda por tipo de documento y área.

Se limita la búsqueda por 3 tipos de áreas(Ciencias de la computación, ingeniería y matemáticas) y por 3 tipos de documentos(Documento de sesión, artículo y capítulo del libro).

The image shows a search refinement interface with two sections: 'Language' and 'Keyword'. Each section has a list of items with checkboxes and a count. The 'Language' section lists English (57,366), Chinese (924), Spanish (217), Russian (133), and Portuguese (127). The 'Keyword' section lists Network Architecture (8,742), Computer Architecture (6,452), Application Programs (6,445), Software Architecture (6,272), and Software Design (4,527). The 'Application Programs' checkbox is highlighted with a red border. Below the sections are three buttons: 'Cancel', 'Exclude', and 'Limit to'.

Language	
<input checked="" type="checkbox"/> English	57,366
<input checked="" type="checkbox"/> Chinese	924
<input type="checkbox"/> Spanish	217
<input type="checkbox"/> Russian	133
<input type="checkbox"/> Portuguese	127

[Show all](#)

---

Keyword	
<input checked="" type="checkbox"/> Network Architecture	8,742
<input checked="" type="checkbox"/> Computer Architecture	6,452
<input checked="" type="checkbox"/> Application Programs	6,445
<input type="checkbox"/> Software Architecture	6,272
<input type="checkbox"/> Software Design	4,527

[Show all](#)

[Cancel](#) [Exclude](#) [Limit to](#)

Figura 6: Refinar búsqueda por idioma y palabra clave.

Se limita la búsqueda a 2 idiomas(Inglés y chino) y a 3 tipos de palabras clave(Red de arquitectura, arquitectura de computadores y programas de aplicación).



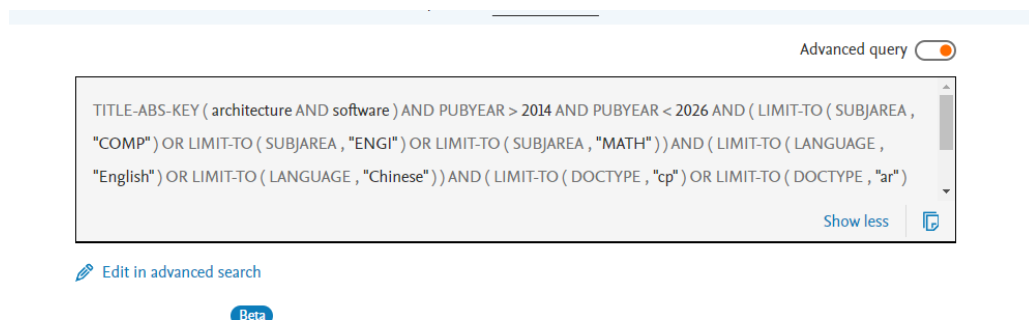


Figura 7: Query exploratorio-parte 1.

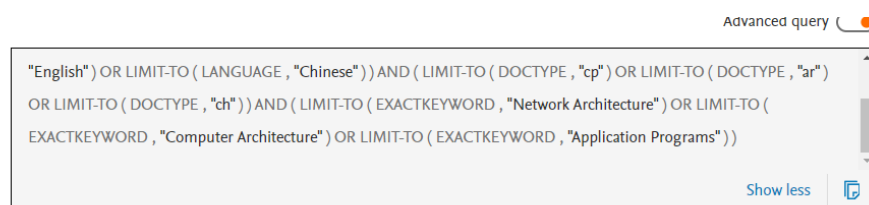


Figura 8: Query exploratorio-parte 2.

### Query exploratorio completo:

```
TITLE-ABS-KEY ( software AND architecture ) AND PUBYEAR > 2014 AND PUBYEAR < 2026 AND ( LIMIT-TO ( SUBJAREA , "COMP" ) OR LIMIT-TO ( SUBJAREA , "ENGI" ) OR LIMIT-TO ( SUBJAREA , "MATH" ) ) AND ( LIMIT-TO ( DOCTYPE , "cp" ) OR LIMIT-TO ( DOCTYPE , "ar" ) OR LIMIT-TO ( DOCTYPE , "ch" ) ) AND ( LIMIT-TO ( LANGUAGE , "English" ) OR LIMIT-TO ( LANGUAGE , "Chinese" ) ) AND ( LIMIT-TO ( EXACTKEYWORD , "Network Architecture" ) OR LIMIT-TO ( EXACTKEYWORD , "Computer Architecture" ) OR LIMIT-TO ( EXACTKEYWORD , "Application Programs" ) )
```

## 2.2. Métodos de análisis

El colab con el que se realizó un análisis bibliográfico, utiliza técnicas de procesamiento de texto y visualización de datos. A continuación, se describen los principales métodos utilizados:

- **Agrupación por fuente:** Se agrupa el conjunto de datos por la columna "Source title" se cuenta la cantidad de artículos por cada fuente. Esto permite identificar las fuentes con más publicaciones y se visualiza mediante gráficos de barras.
- **Limpieza de texto:** Se define una función `clean_text` que toma el texto de los títulos de los artículos, lo convierte a minúsculas, elimina la puntuación y remueve las palabras vacías (stopwords) del idioma inglés.

- **Nube de palabras:** Después de limpiar los títulos, se concatenan en un solo texto y se genera una nube de palabras utilizando la librería **WordCloud**. Esto permite visualizar las palabras más frecuentes en los títulos de forma gráfica.
- **Frecuencia de palabras:** Se realiza un conteo de la frecuencia de cada palabra en los títulos limpios y se crea un **DataFrame** ordenado por frecuencia descendente. Esto permite identificar las palabras más comunes en los títulos.
- **Conteo de palabras clave:** Se definen un conjunto de palabras clave (las 15 más frecuentes) y se cuenta cuántas de esas palabras clave aparecen en cada título. Luego, se ordenan los artículos de acuerdo con la cantidad de palabras clave que contienen y se grafica un gráfico de barras con los 10 artículos que tienen más palabras clave.
- **Resumen de abstracts:** Se seleccionan los 100 artículos con más palabras clave y se resumen sus abstracts. El resumen se realiza tokenizando las oraciones, calculando la frecuencia de las palabras en cada oración y seleccionando las 3 oraciones con mayor puntaje (basado en la frecuencia de palabras).

### 3. Resultados de análisis

#### 3.1. Análisis de Keywords

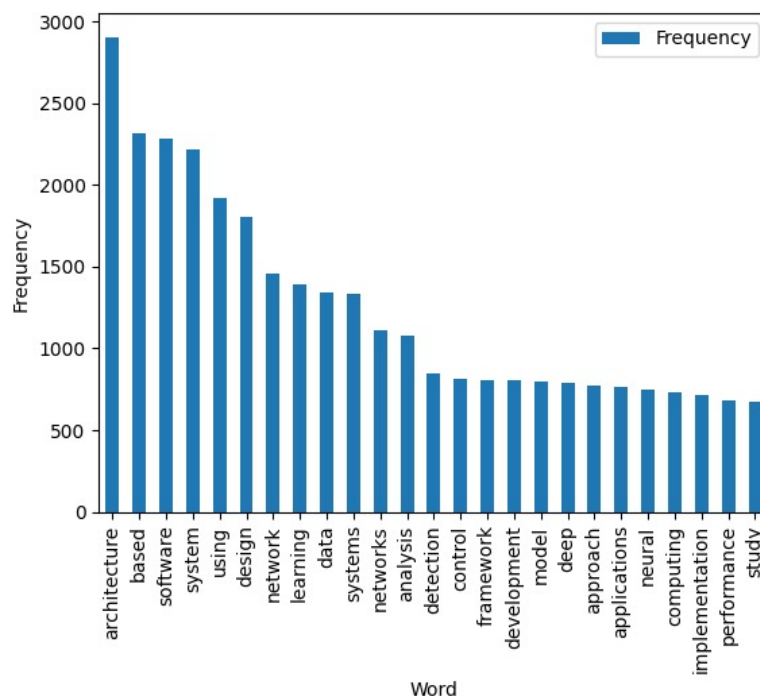


Figura 9: Gráfica de análisis de Keywords.

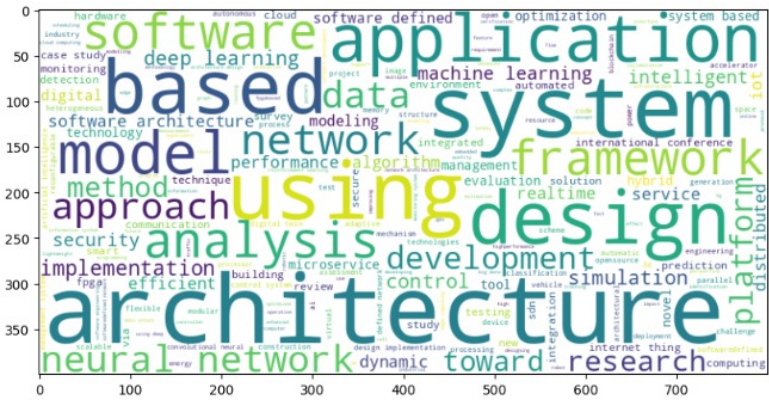


Figura 10: Nube del análisis de Keywords

### 3.2. Análisis de Abstract

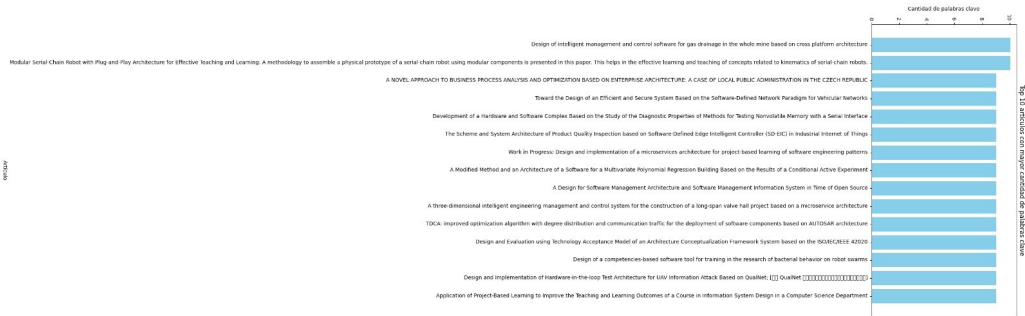


Figura 11: Gráfica de análisis de abstract

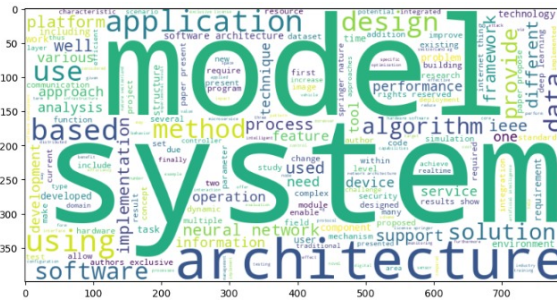


Figura 12: Nube de palabras de análisis de abstract.

### 3.3. Análisis de Title

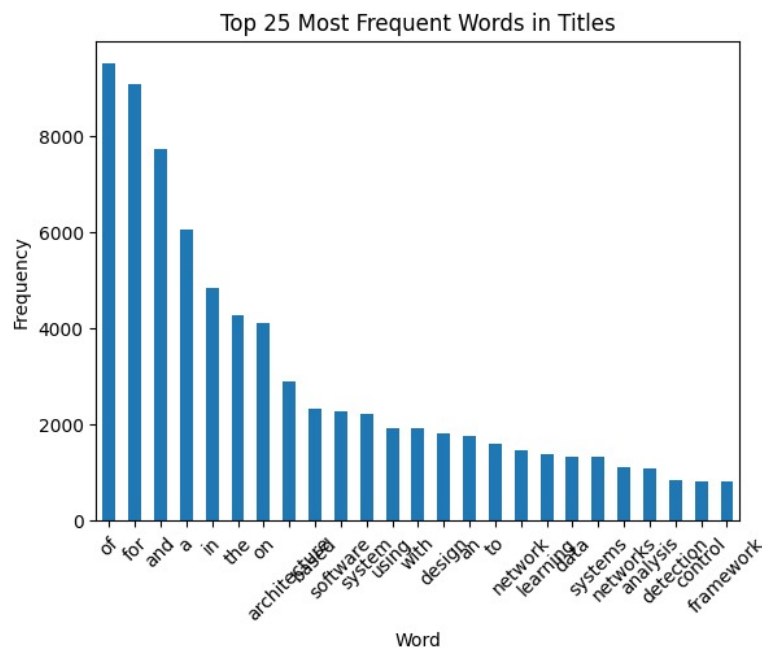


Figura 13: Gráfica de análisis de title.

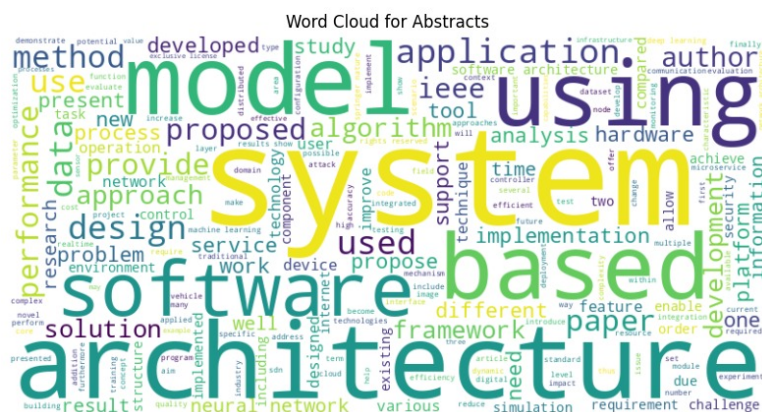


Figura 14: Nube de palabras de análisis de title.

### 3.4. Result Query

### Query del 2014 al 2018

```

TITLE-ABS-KEY ( software AND architecture ) AND PUBYEAR >2014 AND PUBYEAR
<2018 AND ( LIMIT-TO ( SUBJAREA , ‘‘COMP’’ ) OR LIMIT-TO ( SUBJAREA , ‘‘ENGI’’
) OR LIMIT-TO ( SUBJAREA , ‘‘MATH’’ ) ) AND ( LIMIT-TO ( DOCTYPE , ‘‘cp’’ ) OR
LIMIT-TO ( DOCTYPE , ‘‘ar’’ ) OR LIMIT-TO ( DOCTYPE , ‘‘ch’’ ) ) AND (
LIMIT-TO ( LANGUAGE , ‘‘English’’ ) OR LIMIT-TO ( LANGUAGE , ‘‘Chinese’’ ) OR
LIMIT-TO ( LANGUAGE , ‘‘German’’ ) ) AND ( LIMIT-TO ( EXACTKEYWORD , ‘‘Network

```

Architecture'' ) OR LIMIT-TO ( EXACTKEYWORD , ''Computer Architecture'' ) OR  
LIMIT-TO ( EXACTKEYWORD , ''Application Programs'' )

### Query del 2018 al 2025

TITLE-ABS-KEY ( software AND architecture ) AND PUBYEAR >2018 AND PUBYEAR  
<2025 AND ( LIMIT-TO ( SUBJAREA , ''COMP'' ) OR LIMIT-TO ( SUBJAREA , ''ENGI''  
) OR LIMIT-TO ( SUBJAREA , ''MATH'' ) ) AND ( LIMIT-TO ( DOCTYPE , ''cp'' ) OR  
LIMIT-TO ( DOCTYPE , ''ar'' ) OR LIMIT-TO ( DOCTYPE , ''ch'' ) ) AND (  
LIMIT-TO ( LANGUAGE , ''English'' ) OR LIMIT-TO ( LANGUAGE , ''Chinese'' ) OR  
LIMIT-TO ( LANGUAGE , ''German'' ) ) AND ( LIMIT-TO ( EXACTKEYWORD , ''Network  
Architecture'' ) OR LIMIT-TO ( EXACTKEYWORD , ''Computer Architecture'' ) OR  
LIMIT-TO ( EXACTKEYWORD , ''Application Programs'' )

### 3.5. Resultados: comparativas en espacios de tiempo

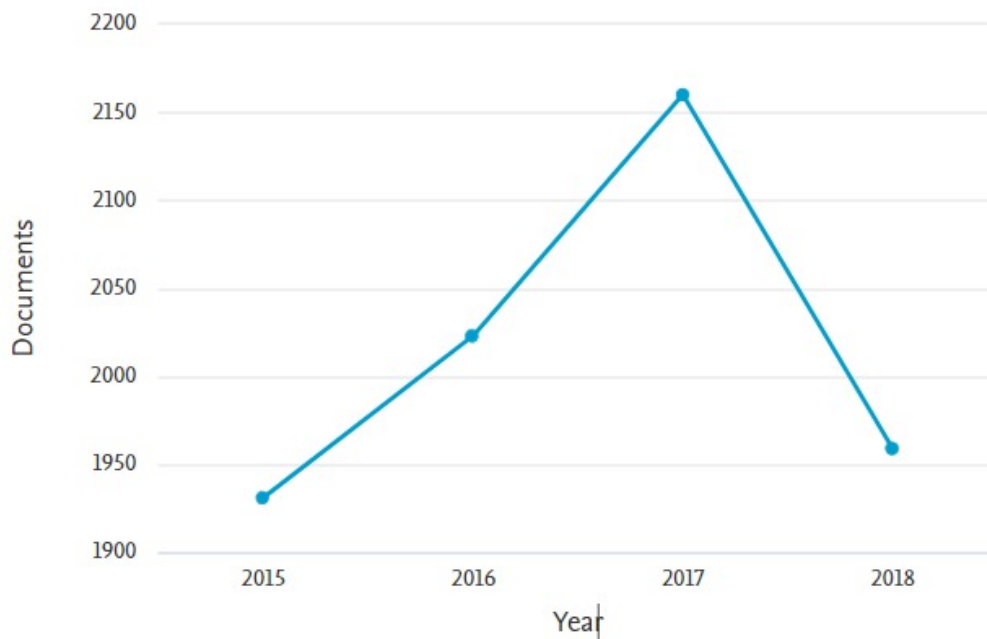


Figura 15: Auge de categorías analizadas conceptuales al 2018.

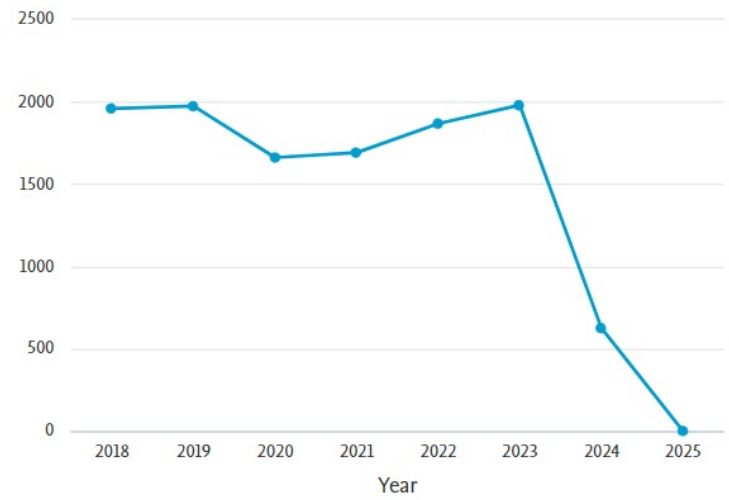


Figura 16: Auge de categorías analizadas conceptuales al 2025.

3.6. Documentos por autor

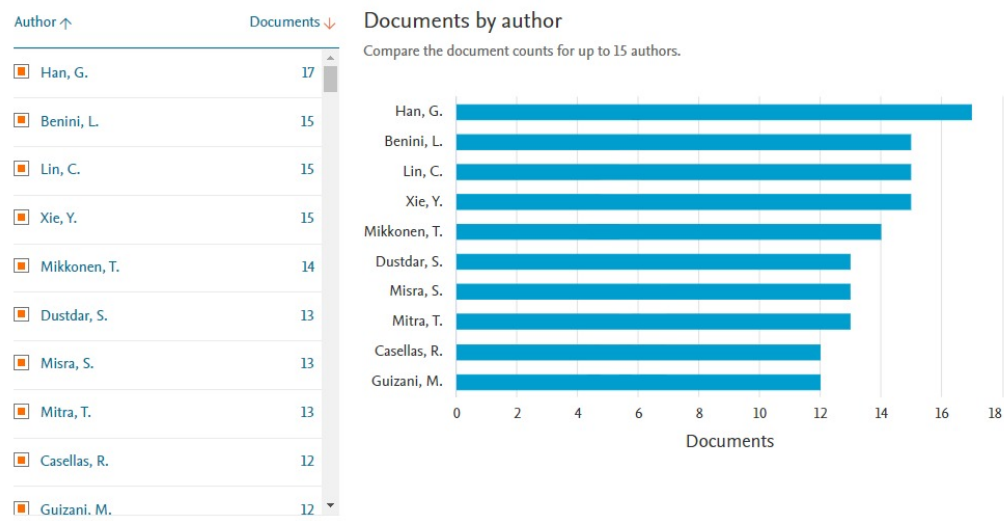


Figura 17: Documentos por autor.

3.7. Categorías principales

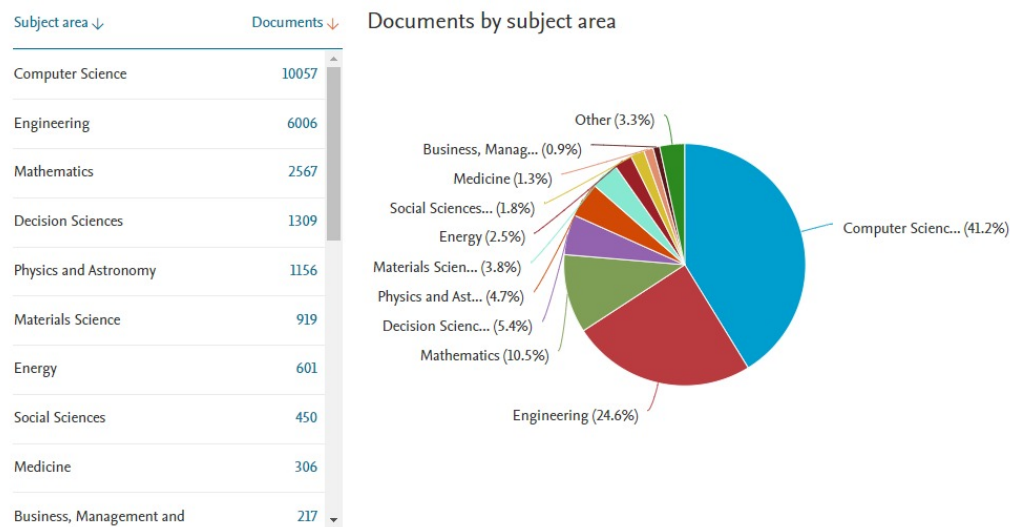


Figura 18: Categorías principales.

## 4. Discusión de los resultados

1. El análisis de palabras clave muestra que términos como “arquitectura de software”, “diseño de software”, “patrones de arquitectura” y “estilos arquitectónicos” son los más frecuentes, lo que refleja la importancia de estos conceptos fundamentales en el campo.
2. Al comparar los períodos de tiempo analizados (2014-2018 y 2018-2025), se observa un aumento en la cantidad de publicaciones relacionadas con la arquitectura de software, lo que sugiere un creciente interés y relevancia de este tema en la industria y la academia.
3. Las categorías principales identificadas, como “Ingeniería”, “ciencias de la Computación” y “matemáticas”, resaltan la naturaleza interdisciplinaria de la arquitectura de software y su relación con diversas áreas de la informática.
4. El análisis de autores revela que no hay un autor dominante en el campo, sino una variedad de contribuciones de diferentes investigadores y profesionales, lo que indica la amplitud y diversidad de enfoques y perspectivas en la arquitectura de software.
5. Con el colab se tuvo la facilidad para importar y procesar grandes volúmenes de datos obtenidos de la base de datos académica como Scopus. Esto permitió ahorrar tiempo y esfuerzo significativo en tareas de preparación y limpieza de datos, que suelen ser tediosas y propensas a errores cuando se realizan manualmente.
6. Se recomienda explorar otras funcionalidades avanzadas de Colab, como la integración con servicios en la nube, el uso de GPUs para tareas de cómputo intensivo, y la capacidad de exportar y compartir los resultados de manera más eficiente.

## 5. Conclusiones y recomendaciones

1. El repositorio de GitLab [1] proporcionó un entorno centralizado y accesible para evaluar y procesar los datos recopilados de la fuente académica 'Scopus'.
2. El uso del Colab demostró ser una herramienta valiosa para realizar el análisis bibliográfico sobre arquitectura de software. Las capacidades de procesamiento de datos, visualización y colaboración que ofrece Colab facilitaron el manejo y análisis de un gran volumen de documentos académicos y científicos.
3. La arquitectura de software es un área en constante evolución y de gran importancia para el desarrollo de sistemas de software de alta calidad, escalables y mantenibles.
4. Es fundamental mantenerse actualizado con las últimas tendencias, enfoques y mejores prácticas en la arquitectura de software, ya que esto permitirá abordar de manera efectiva los desafíos y requisitos de los sistemas modernos.
5. Se recomienda profundizar en temas específicos, como la arquitectura de microservicios, la arquitectura basada en eventos, la seguridad en el diseño arquitectónico y la integración de arquitecturas con tecnologías emergentes, como la inteligencia artificial y el aprendizaje automático.
6. Es importante fomentar la colaboración y el intercambio de conocimientos entre diferentes actores involucrados en la arquitectura de software, como investigadores, desarrolladores, arquitectos y profesionales de la industria, para impulsar el avance y la innovación en este campo.
7. Se sugiere realizar investigaciones futuras enfocadas en el desarrollo de metodologías y herramientas que faciliten el diseño, evaluación y documentación de arquitecturas de software, así como en la definición de estándares y mejores prácticas para la gestión del ciclo de vida de las arquitecturas.



## Bibliografía

- [1] Oscar Mendez. *bibliographic\_analysis*. Repositorio de GitLab. Accedido: 2024-06-06. abril de 2024. URL: [https://gitlab.com/konrad%5C\\_lorenz/bibliographic%5C\\_analysis](https://gitlab.com/konrad%5C_lorenz/bibliographic%5C_analysis).