



ESCUELA DE INGENIERÍA DE FUENLABRADA

GRADO EN INGENIERÍA EN SISTEMAS
AUDIOVISUALES Y MULTIMEDIA

TRABAJO FIN DE GRADO

ANÁLISIS DE AUTORES EN COMÚN DE CONGRESOS Y
REVISTAS CIENTÍFICAS USANDO DBLP

Autor : Sergio García Sánchez

Tutor : Dr. Gregorio Robles

Cotutor: (si procede)

Curso académico 2023/2024

Trabajo Fin de Grado

ANÁLISIS DE AUTORES EN COMÚN DE CONGRESOS Y REVISTAS CIENTÍFICAS USANDO DBLP

Autor : Sergio García Sánchez

Tutor : Dr. Nombre del Profesor/a

La defensa del presente Proyecto Fin de Carrera se realizó el día de
de 202X, siendo calificada por el siguiente tribunal:

Presidente:

Secretario:

Vocal:

y habiendo obtenido la siguiente **Calificación:**

Fuenlabrada, a de de 202X



©2024 Sergio García Sánchez

Algunos derechos reservados

Este documento se distribuye bajo la licencia “Atribución-CompartirIgual 4.0 Internacional” de
Creative Commons, disponible en

<https://creativecommons.org/licenses/by-sa/4.0/deed.es>

*Dedicado a
mi familia / mi abuelo / mi abuela*

Agradecimientos

Aquí vienen los agradecimientos... Aunque está bien acordarse de la pareja, no hay que olvidarse de dar las gracias a tu madre, que aunque a veces no lo parezca disfrutará tanto de tus logros como tú... Además, la pareja quizás no sea para siempre, pero tu madre sí.

Resumen

Aquí viene un resumen del proyecto. Ha de constar de tres o cuatro párrafos, donde se presente de manera clara y concisa de qué va el proyecto. Han de quedar respondidas las siguientes preguntas:

- ¿De qué va este proyecto? ¿Cuál es su objetivo principal?
- ¿Cómo se ha realizado? ¿Qué tecnologías están involucradas?
- ¿En qué contexto se ha realizado el proyecto? ¿Es un proyecto dentro de un marco general?

Lo mejor es escribir el resumen al final.

Summary

Here comes a translation of the “Resumen” into English. Please, double check it for correct grammar and spelling. As it is the translation of the “Resumen”, which is supposed to be written at the end, this as well should be filled out just before submitting.

Índice general

| | |
|---|-----------|
| 1. Introducción | 1 |
| 1.1. Sección | 1 |
| 1.1.1. Estilo | 2 |
| 1.2. Estructura de la memoria | 4 |
| 2. Objetivos | 5 |
| 2.1. Objetivo general | 5 |
| 2.2. Objetivos específicos | 5 |
| 2.3. Planificación temporal | 5 |
| 3. Estado del arte | 7 |
| 3.1. Sección 1 | 8 |
| 4. Diseño e implementación | 9 |
| 4.1. Arquitectura general | 9 |
| 5. Experimentos y validación | 11 |
| 6. Resultados | 13 |
| 7. Conclusiones | 15 |
| 7.1. Consecución de objetivos | 15 |
| 7.2. Aplicación de lo aprendido | 15 |
| 7.3. Lecciones aprendidas | 16 |
| 7.4. Trabajos futuros | 16 |

A. Manual de usuario**17**

Índice de figuras

| | |
|---|----|
| 1.1. Página con enlaces a hilos | 3 |
| 1.2. Estructura del parser básico | 4 |
| 4.1. Estructura del parser básico | 10 |

Capítulo 1

Introducción

En este capítulo se introduce el proyecto. Debería tener información general sobre el mismo, dando la información sobre el contexto en el que se ha desarrollado.

No te olvides de echarle un ojo a la página con los cinco errores de escritura más frecuentes¹.

Aconsejo a todo el mundo que mire y se inspire en memorias pasadas. Las memorias de los proyectos que he llevado yo están (casi) todas almacenadas en mi web del GSyC².

En mayo de 2023 me apunté a un curso de innovación docente donde nos pidieron hacer un podcast con temática docente. Aproveché entonces para hacer un podcast de unos 30 minutos donde en los primeros quince minutos introducía LaTeX y la memoria, y en los segundos hacía hincapién en aquellas cosas que más os cuestan utilizar en la memoria: las figuras, las tablas y las citas. Podéis escuchar el podcast en Internet³.

1.1. Sección

Esto es una sección, que es una estructura menor que un capítulo.

Por cierto, a veces me comentáis que no os compila por las tildes. Eso es un problema de codificación. Al guardar el archivo, guardad la codificación de “ISO-Latin-1” a “UTF-8” (o viceversa) y funcionará.

¹<http://www.tallerdeescritores.com/errores-de-escritura-frecuentes>

²<https://gsyc.urjc.es/~grex/pfcs/>

³<https://podcasters.spotify.com/pod/show/gregorio-robles9/episodes/>

Tu-memoria-de-Trabajo-Fin-de-Grado-o-de-Mster-en-LaTeX-e23hucr/a-a58kp2

1.1.1. Estilo

Recomiendo leer los consejos prácticos sobre escribir documentos científicos en L^AT_EX de Diomidis Spinellis⁴.

Lee sobre el uso de las comas⁵. Las comas en español no se ponen al tuntún. Y nunca, nunca entre el sujeto y el predicado (p.ej. en “Yo, hago el TFG” sobre la coma). La coma no debe separar el sujeto del predicado en una oración, pues se cortaría la secuencia natural del discurso. No se considera apropiado el uso de la llamada coma respiratoria o *coma criminal*. Solamente se suele escribir una coma para marcar el lugar que queda cuando omitimos el verbo de una oración, pero es un caso que se da de manera muy infrecuente al escribir un texto científico (p.ej. “El Real Madrid, campeón de Europa”).

A continuación, viene una figura, la Figura 1.1. Observarás que el texto dentro de la referencia es el identificador de la figura (que se corresponden con el “label” dentro de la misma). También habrás tomado nota de cómo se ponen las “comillas dobles” para que se muestren correctamente. Nota que hay unas comillas de inicio (“) y otras de cierre (”), y que son diferentes. Volviendo a las referencias, nota que al compilar, la primera vez se crea un diccionario con las referencias, y en la segunda compilación se “rellenan” estas referencias. Por eso hay que compilar dos veces tu memoria. Si no, no se crearán las referencias.

A continuación un bloque “verbatim”, que se utiliza para mostrar texto tal cual. Se puede utilizar para ofrecer el contenido de correos electrónicos, código, entre otras cosas.

```
From gaurav at gold-solutions.co.uk  Fri Jan 14 14:51:11 2005
From: gaurav at gold-solutions.co.uk  (gaurav_gold)
Date: Fri Jan 14 19:25:51 2005
Subject: [Mailman-Users] mailman issues
Message-ID: <003c01c4fa40$1d99b4c0$94592252@gaurav7klgnyif>
```

```
Dear Sir/Madam,

How can people reply to the mailing list?  How do i turn off
this feature? How can i also enable a feature where if someone
replies the newsletter the email gets deleted?

Thanks
```

⁴<https://github.com/dspinellis/latex-advice>

⁵<http://narrativabreve.com/2015/02/opiniones-de-un-corrector-de-estilo-11-recetas-par>
html

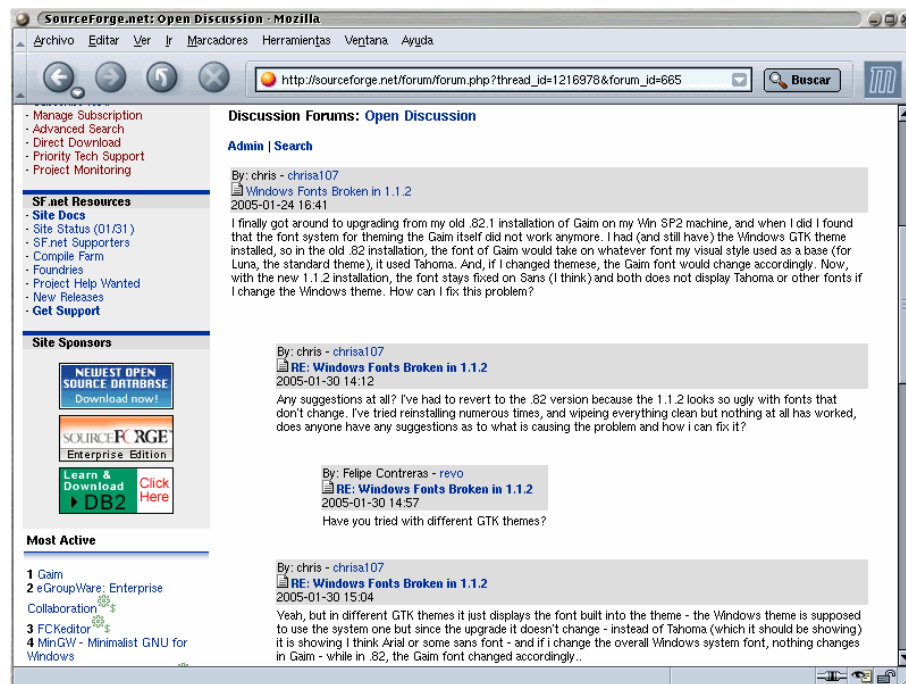


Figura 1.1: Página con enlaces a hilos

From msapiro at value.net Fri Jan 14 19:48:51 2005
 From: msapiro at value.net (Mark Sapiro)
 Date: Fri Jan 14 19:49:04 2005
 Subject: [Mailman-Users] mailman issues
 In-Reply-To: <003c01c4fa40\$1d99b4c0\$94592252@gaurav7klgnyif>
 Message-ID: <PC173020050114104851057801b04d55@msapiro>

gaurav_gold wrote:

>How can people reply to the mailing list? How do i turn off
 this feature? How can i also enable a feature where if someone
 replies the newsletter the email gets deleted?

See the FAQ

>Mailman FAQ: <http://www.python.org/cgi-bin/faqw-mm.py>
 article 3.11

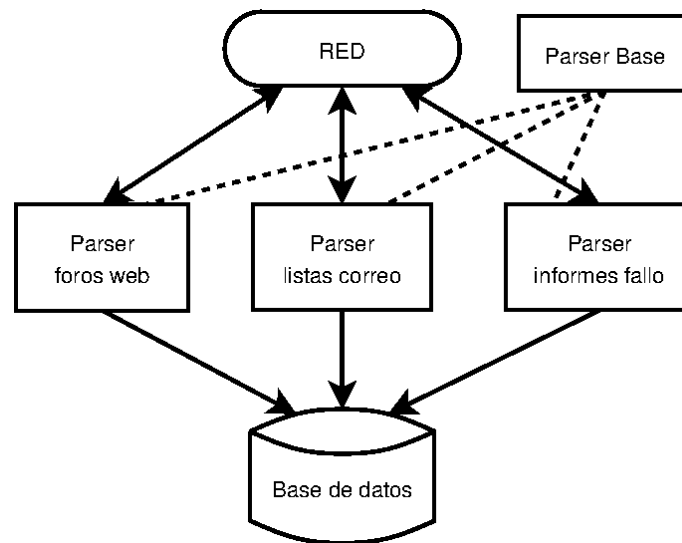


Figura 1.2: Estructura del parser básico

1.2. Estructura de la memoria

En esta sección se debería introducir la estructura de la memoria.

Así:

- En el primer capítulo se hace una intro al proyecto.
- En el capítulo 2 (ojo, otra referencia automática) se muestran los objetivos del proyecto.
- A continuación se presenta el estado del arte en el capítulo 3.
- ...

Capítulo 2

Objetivos

2.1. Objetivo general

Mi trabajo fin de grado consiste en crear de una herramienta de análisis de autores que compartan congresos y revistas utilizando DBLP. Se pretende entender graficamente I) la cantidad de colaboraciones, y II) las dinámicas de publicación de los autores por congreso, todo dentro del marco de la comunidad científica.

2.2. Objetivos específicos

Este proyecto tiene como objetivo crear una herramienta de análisis de autores coincidentes utilizando DBLP, sin embargo, este concepto es lo suficientemente amplio como para abordarlo de manera general. Por lo tanto, los objetivos específicos detallan de la siguiente manera:

1. Decidir las metodologías, que tecnologías y proceso tendrá el ecosistema.
2. Desarrollo de la herramienta de visualización de datos: Para ello se debe implementar:
 - La herramienta extracción de datos, procesamiento y ordenación de los mismos.
 - Una interfaz gráfica de usuario (WEB) que facilite la interacción y visualización de los datos.
3. Comparación de la interfaz gráfica con la herramienta de conteo de autores por congreso.

2.3. Planificación temporal

A mí me gusta que aquí pongáis una descripción de lo que os ha llevado realizar el trabajo. Hay gente que añade un diagrama de GANTT. Lo importante es que quede claro cuánto tiempo llevas (tiempo natural, p.ej., 6 meses) y a qué nivel de esfuerzo (p.ej., principalmente los fines de semana).

Capítulo 3

Estado del arte

En este capítulo se van a introducir las tecnologías utilizadas en este trabajo.

3.1. Python

Python ha adquirido una relevancia como lenguaje de programación mas utilizado en la comunidad académica y científica debido a que se trata de un lenguaje de alto nivel, cuya característica principal es sencillez, la versatilidad y la gran colección de bibliotecas especializadas componen el resto de características mas importantes.

Dentro del contexto de este trabajo, la creación de una herramienta de análisis donde se observa como los autores coinciden en los congresos es muy destacado debido a:

Extracción y la posterior manipulación de datos: Bibliotecas como sqlite3, lxml o json nos ayuda a convertir los datos en una base de datos manejable.

Web Scraping y APIs: En este apartado destacan las librerías como BeautifulSoup (utilizadas en el comienzo del trabajo) y Request utilizadas para obtención de los datos dentro de la web, incluyendo la recolección de la información de los datos bibliográficos como en este caso DBLP.

Visualización de datos: Bibliotecas como plotly o matplotlib se utilizan para la creación de gráficos detallados de los datos de las publicaciones y colaboraciones entre autores. En este caso se desestimó la idea, ya que como se verá a continuación se desarrolló una herramienta web.

Ventajas de utilizar Python en este proyecto:

Amplia comunidad y soporte: facilidad de acceso a recursos y tutoriales y una amplia comunidad de usuarios y desarrolladores activa.

Eficiencia y simplicidad de uso: Con una sintaxis clara y precisa puedes desarrollar una solución eficiente y rápida.

3.1.1. SQLite3

SQLite3 es una librería en Python que proporciona las herramientas para interactuar con la base de datos de SQLite. Dicha base de datos es una biblioteca de C que implementa un motor de bases de datos SQL autónomo, fiable, sin configuración y con todas las funciones de SQL.

La integración con SQL se produce directamente desde python pudiendo utilizar los comandos: 'SELECT', 'INSERT', 'UPDATE', 'DELETE', como otros operadores de las bases de datos.

No precisa de configuración en un servidor independiente, las bases de datos se almacenan en un archivo del sistema de archivos, facilitando el transporte y su uso. Es importante destacar que cada archivo de base de datos es completamente autónomo y contiene los datos y la lógica necesaria para su funcionamiento.

3.1.2. Lxml

Se trata de una librería de python que tiene como funciones el procesamiento y manipulación de archivos HTML y XML, es una herramienta con mucho potencial que está construida sobre bibliotecas C, concretamente 'libxslt' y 'libxml2'.

Tiene como características principales la facilidad de uso, compatibilidad con HTML5, manipulación de árboles de datos y destacan el rendimiento y velocidad.

En nuestro trabajo se utiliza para realizar el parseo de información hacia la base de datos en formato SQL.

3.1.3. Os

Biblioteca que proporciona una interfaz para interactuar con el sistema operativo base, permitiendo ejecutar comandos propios del sistema operativo de forma portátil. En nuestra herramienta se utiliza para levantar un error del sistema.

3.1.4. Argparse

Se trata de una librería estandar que proporciona las herramientas para definir los argumentos que precisa una interfaz de línea, como deben procesarse y generan la ayuda para los usuarios. Fue incorporado a Python 2.7 como remplazo de optparse

3.1.5. Time

Biblioteca que suministra las funciones para trabajar con el tiempo, pudiéndose implementar funciones como medir el tiempo de ejecución del programa, gestionar marcas de tiempo o incluir retrasos en el código.

En nuestro trabajo se utiliza para medir la cantidad de tiempo que tarda en realizar el parseo del XML de DBLP.

3.1.6. Json

Módulo que proporciona las herramientas necesarias para la manipulación de archivos JSON. En nuestro proyecto se utiliza para crear el JSON que será leído mediante la API.

3.2. JSON

JavaScript Object Notation se caracteriza por su simplicidad, legibilidad y compatibilidad con multiples lenguajes de programación, incluyendo como en nuestro caso Python. Este formato tiene como función almacenar información estructurada que se utiliza para la transferencia de datos entre un servicio web y aplicaciones, principalmente.

Originariamente, JSON se utilizaba con la notación basada en objetos de JavaScript, el formato se compone de dos campos diferenciados key y value. Las Keys contienen una cadena de caracteres entrecomillada y los Values o valores contienen un tipo de dato válido JSON, del mismo modo entrecomillado. Dicho campo puede estar formado por un array, un string, un objeto, un diccionario, un boolean, un número o un tipo nulo.

En este punto se presenta un ejemplo de la estructura tipo de un formato JSON:

```
{  
    "Key": {
```

```
"Dict": {
    "option1": "ok",
    "List": [
        "thing1",
        "thing2",
        "thing3"
    ]
},
"Dict2": {
    "opcode": "custom",
    "List2": [
        "customA",
        "customB"
    ]
}
}
```

En el contexto del trabajo, que es la de crear una herramienta de análisis, desempeña un papel fundamental en los siguientes aspectos:

Almacenamiento de datos: Permite almacenar información complicada bien organizada, debido a su estructura flexible y jerárquica. En este trabajo cada autor o publicación se puede almacenar o representar como un objeto JSON con atributos como nombre, año, título... Facilitando el manejo y haciéndolo mas accesible.

Intercambio de datos: Como se mencionó anteriormente, es utilizado primordialmente para el intercambio entre servicios web y aplicaciones. La funcionalidad de este trabajo es la de ser extraídos mediante una API, simplificando el análisis y el procesamiento.

Sencillez a la hora de manipular los datos: Se convierte en una tarea mas sencilla debido a la combinación con bibliotecas de Python, que facilitan la escritura, lectura y transformación de la información.

En conclusión, se trata de un formato muy versátil debido a su compatibilidad multilenguaje, pose flexibilidad y extensibilidad, ya que se puede incorporar nueva información conforme sea

necesario y se trata de un formato sencillo y claro a la hora de su visualización.

3.3. HTML

3.4. CSS

3.5. NODEJS

Capítulo 4

Diseño e implementación

Aquí viene todo lo que has hecho tú (tecnológicamente). Puedes entrar hasta el detalle. Es la parte más importante de la memoria, porque describe lo que has hecho tú. Eso sí, normalmente aconsejo no poner código, sino diagramas.

4.1. Arquitectura general

Si tu proyecto es un software, siempre es bueno poner la arquitectura (que es cómo se estructura tu programa a “vista de pájaro”).

Por ejemplo, puedes verlo en la figura 4.1. \LaTeX pone las figuras donde mejor cuadran. Y eso quiere decir que quizás no lo haga donde lo hemos puesto... Eso no es malo. A veces queda un poco raro, pero es la filosofía de \LaTeX : tú al contenido, que yo me encargo de la maquetación.

Recuerda que toda figura que añadas a tu memoria debe ser explicada. Sí, aunque te parezca evidente lo que se ve en la figura 4.1, la figura en sí solamente es un apoyo a tu texto. Así que explica lo que se ve en la figura, haciendo referencia a la misma tal y como ves aquí. Por ejemplo: En la figura 4.1 se puede ver que la estructura del *parser* básico, que consta de seis componentes diferentes: los datos se obtienen de la red, y según el tipo de dato, se pasará a un *parser* específico y bla, bla, bla. . .

Si utilizas una base de datos, no te olvides de incluir también un diagrama de entidad-relación.

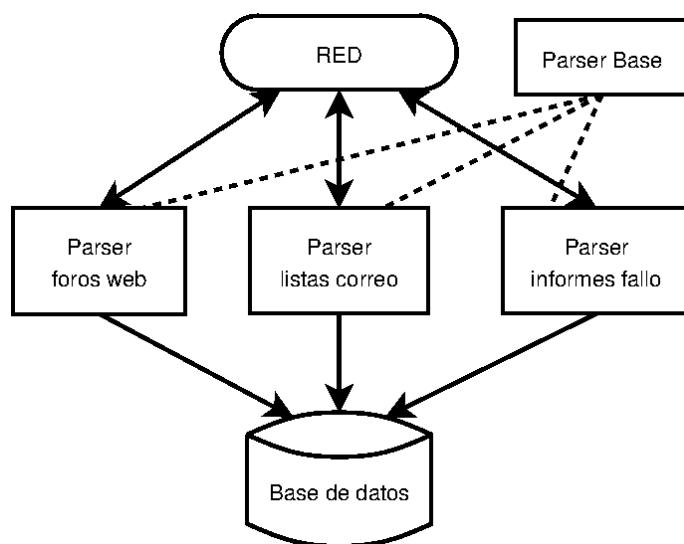


Figura 4.1: Estructura del parser básico

Capítulo 5

Experimentos y validación

Este capítulo se introdujo como requisito en 2019. Describe los experimentos y casos de test que tuviste que implementar para validar tus resultados. Incluye también los resultados de validación que permiten afirmar que tus resultados son correctos.

Capítulo 6

Resultados

En este capítulo se incluyen los resultados de tu trabajo fin de grado.

Si es una herramienta de análisis lo que has realizado, aquí puedes poner ejemplos de haberla utilizado para que se vea su utilidad.

Capítulo 7

Conclusiones

7.1. Consecución de objetivos

Esta sección es la sección espejo de las dos primeras del capítulo de objetivos, donde se planteaba el objetivo general y se elaboraban los específicos.

Es aquí donde hay que debatir qué se ha conseguido y qué no. Cuando algo no se ha conseguido, se ha de justificar, en términos de qué problemas se han encontrado y qué medidas se han tomado para mitigar esos problemas.

Y si has llegado hasta aquí, siempre es bueno pasarle el corrector ortográfico, que las erratas quedan fatal en la memoria final. Para eso, en Linux tenemos `aspell`, que se ejecuta de la siguiente manera desde la línea de *shell*:

```
aspell --lang=es_ES -c memoria.tex
```

7.2. Aplicación de lo aprendido

Aquí viene lo que has aprendido durante el Grado/Máster y que has aplicado en el TF-G/TFM. Una buena idea es poner las asignaturas más relacionadas y comentar en un párrafo los conocimientos y habilidades puestos en práctica.

1. a

2. b

7.3. Lecciones aprendidas

Aquí viene lo que has aprendido en el Trabajo Fin de Grado/Máster.

1. Aquí viene uno.
2. Aquí viene otro.

7.4. Trabajos futuros

Ningún proyecto ni software se termina, así que aquí vienen ideas y funcionalidades que estaría bien tener implementadas en el futuro.

Es un apartado que sirve para dar ideas de cara a futuros TFGs/TFM.

Apéndice A

Manual de usuario

Esto es un apéndice. Si has creado una aplicación, siempre viene bien tener un manual de usuario. Pues ponlo aquí.

