



ugr

Universidad  
de Granada

TRABAJO FIN DE GRADO  
INGENIERÍA INFORMÁTICA

---

# Detección de errores en bases de datos químicas

---

**Autor**

Jesús Navarro Merino

**Directora**

Rocío Celeste Romero Zaliz



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE  
TELECOMUNICACIÓN

---

Granada, julio de 2023



# **Detección de errores en bases de datos químicas**

Jesús Navarro Merino

**Palabras clave:** palabra\_clave1, palabra\_clave2, palabra\_clave3, .....

## **Resumen**

Poner aquí el resumen.



**Project Title: Project Subtitle**

First name, Family name (student)

**Keywords:** Keyword1, Keyword2, Keyword3, ....

**Abstract**

Write here the abstract in English.



---

Yo, **Jesús Navarro Merino**, alumno de la titulación Grado en Ingeniería Informática de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 15429457E, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Jesús Navarro Merino

Granada a X de MES de 201 .





---

Dña. **Rocío Celeste Romero Zaliz**, Profesora del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada.

**Informan:**

Que el presente trabajo, titulado *Detección de errores en bases de datos químicas*, ha sido realizado bajo su supervisión por **Jesús Navarro Merino**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a X de mes de 201 .

**La directora:**

**Rocío Celeste Romero Zaliz**



# Agradecimientos

Poner aquí agradecimientos...



# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación y objetivos . . . . .	1
1.2. Objetivos . . . . .	4
1.3. Estructura de la memoria . . . . .	4
<b>2. Estado del arte y fundamentos teóricos</b>	<b>7</b>
<b>3. Gestión y Planificación del proyecto</b>	<b>9</b>
3.1. Metodología . . . . .	9
3.2. Gestión de recursos . . . . .	9
3.2.1. Recursos humanos . . . . .	9
3.2.2. Recursos materiales . . . . .	9
3.2.3. Recursos software . . . . .	9
3.3. Gestión de costes . . . . .	9
3.3.1. Coste de recursos humanos . . . . .	9
3.3.2. Coste hardware . . . . .	9
3.3.3. Otros costes . . . . .	9
3.3.4. Presupuesto final . . . . .	9
3.4. Análisis de riesgos . . . . .	9



# Índice de figuras

1.1. Distintas cadenas SMILES válidas para la misma molécula de [nombre-molécula]. [empty citation] . . . . .	2
------------------------------------------------------------------------------------------------------------------	---





# Índice de tablas

1.1. Códigos SMILES y sus representaciones 2D según Sigma-	
Aldrich . . . . .	4
1.2. Códigos SMILES y sus representaciones 2D según SciFinder .	5
1.3. Códigos SMILES y sus representaciones 2D según Sigma-	
Aldrich . . . . .	6



# Capítulo 1

## Introducción

Meter aqui una introduccion general sobre la química, la química informática (y cómo surge en base a las necesidades computacionales), mencionar brevemente la organometalica (mas tarde en la seccion de estado del arte profundizo, junto con cosas de dibujado, tutorial de SMILES, y representacion de moléculas)

### 1.1. Motivación y objetivos

Los formatos de notación lineal llevan siendo un tema de interés e investigación para los científicos desde mediados del siglo 19 [1] **Terminar esto**

En la actualidad, existen varias representaciones lineales (**Para esto debo haber hablado antes de las representaciones lineales, o bien lo explico aquí, o bien lo redirijo a los fundamentos teóricos. O bien, justo antes de empezar este párrafo, que será lo mejor-¿quitar entonces la siguiente negrita**), siendo las más usadas SMILES, InChI, y SELFIES [3]. **Como comenté antes, una forma muy potente de representar moléculas y compuestos químicos es mediante cadenas strings**, y de esto justamente se encargan las representaciones lineales: traducir una molécula, con sus átomos, enlaces entre ellos, ciclos y otras propiedades características, en una cadena string que la represente, y que la máquina y los propios químicos puedan entender. Sin embargo, hay diferencias notables entre las representaciones, tanto en la sintaxis de las cadenas que se generan como en las aplicaciones que se le puede dar a cada una de ellas.

SMILES, ideada por David Weininger, sale a la luz en 1988 satisfaciendo con creces las necesidades de procesamiento de información química que había, desbancando a la representación estandarizada del momento, Wiswes-

ser Line Notation (WLN). Desde ese entonces SMILES se convirtió —y sigue siendo a día de hoy— en el estándar de representación lineal, ya que permite describir estructuras moleculares de una forma sencilla en un formato fácil de leer, lo que ha hecho que sea una herramienta popular en la química computacional, siendo la más usada entre investigadores y químicos. Pese a esto, SMILES tiene dos grandes inconvenientes: una misma molécula puede escribirse con varias cadenas SMILES distintas válidas, es decir, tiene sinónimos (Figura 1.1); y no es robusto ni sintáctica ni semánticamente. En este sentido se podría generar un string que no represente una molécula válida, como lo es CC(CCCC, el cual tiene un paréntesis sin cerrar (lo que implica que no se delimita cuándo acaba la rama). O generar una molécula que no sea químicamente viable como CO=CC, que muestra un átomo de oxígeno neutro formando tres enlaces (superando el límite de enlaces covalentes que un oxígeno neutro puede tener) [3].

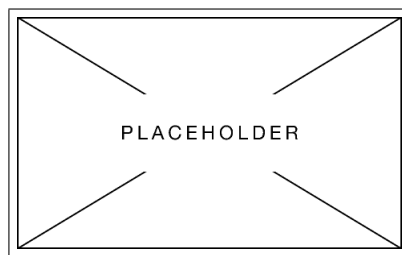


Figura 1.1: Distintas cadenas SMILES válidas para la misma molécula de [nombre-molécula]. [empty citation]

Esto tiene especial relevancia en el ámbito del Machine Learning (ML). Aunque se sale del alcance de este trabajo, uno de los grandes objetivos de la química computacional es la creación o diseño de nuevas moléculas. Se podrían crear modelos de ML o redes neuronales capaces de generar moléculas ficticias válidas, para posteriormente ver sus propiedades, valorarlas energéticamente para ver cuán estables son, y estudiar su viabilidad en distintas aplicaciones, entre otras cosas. SMILES dificulta esta tarea, y por ello, aparece en 2020, SELFIES (SELF-referencIng Embedded Strings), una nueva representación lineal 100 % robusta, muy usada actualmente para modelos generativos. Ver [3, 2] para más detalles de cómo soluciona los problemas de robustez y otras características de la representación. SELFIES es relativamente reciente y continuamente está ampliando sus funcionalidades, mejorando su simplicidad y facilidad de uso para el usuario [4]. Aun así, no se termina de instaurar entre la comunidad investigadora.

**AQUI METER ALGO DE InChI, o lo meto arriba, antes de hablar sobre SELFIES.**

Por todo lo anterior, me centraré en la notación SMILES durante el desa-

rrollo de este trabajo. Dicho esto, existen diversas fuentes de datos en química donde se recoge gran cantidad de información acerca de los compuestos. Menciono las más importantes y las que serán objeto de interés. *PubChem*, una base de datos abierta que sirve información a millones de usuarios en todo el mundo, desde investigadores y estudiantes hasta el público general. Recogen para cada compuesto, información sobre su estructura, representaciones 2D y 3D, identificadores, propiedades químicas y físicas, patentes, avisos de toxicidad, etc. [5]

*SciFinder*, una herramienta de investigación muy potente que permite explorar las bases de datos de CAS (American Chemical Society) las cuales contienen literatura sobre Química y otras disciplinas afines como Física, Biomedicina, Geología, Ingeniería Química, etc. Incluye referencias bibliográficas y resúmenes de artículos, informes, y libros entre otras cosas. Permite realizar búsquedas por estructura, nombres de sustancias o identificadores, reacciones en la que participa dicha sustancia, artículos y publicaciones que nombren el compuesto en cuestión, e incluso proveedores de compra. Para el uso de esta herramienta es necesario acceder mediante la red de una institución autorizada (en este caso trabajo mediante VPN de la UGR) y seguir los pasos para registrarte <sup>1</sup>

uno de los principales problemas que se detectan es la heterogeneidad en las distintas bases de datos **continuar esto**

**Aquí empezar ya con la tabla comparativa. Para eso, tengo que introducir las fuentes de datos**

**Meter parrafo de los quimicos colaboradores...mirar pedro**

Por todo lo anterior, una solución que permita aunar... **continuar esto**

Por tanto, el objetivo principal de este Trabajo Fin de Grado sería modificar el paquete OpenBabel creando un método para canonizar códigos SMILES, orientado específicamente para compuestos organometálicos, de los cuales dispongo de un conjunto de datos de 30 moléculas considerados de interés por la tutora y los químicos con los que colabora. Para ello, se establecen los siguientes subobjetivos:

- Analizar y comparar las cadenas SMILES de distintas bases de datos (p.ej. Sigma-Aldrich, SciFinder) viendo los posibles sinónimos para una misma molécula.
- Determinar un sistema que genere, a partir de cualquier sinónimo SMILES de la misma molécula, un único SMILES canónico.
- Definir otro algoritmo o conjunto de reglas que mejore, aunque sea

---

<sup>1</sup>Pasos para el registro en SciFinder [https://bibliotecaugr.libguides.com/scifinder\\_scholar](https://bibliotecaugr.libguides.com/scifinder_scholar)

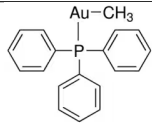
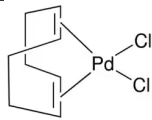
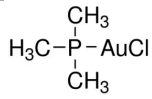
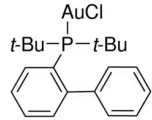
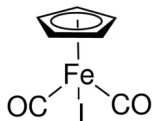
Código SMILES	Representación 2D
<chem>C[Au].c1ccc(cc1)P(c2ccccc2)c3ccccc3</chem>	
<chem>Cl[Pd]Cl.C1CC=CCCC=C1</chem>	
<chem>Cl[Au].CP(C)C</chem>	
<chem>Cl[Au].CC(C)(C)P(c1ccccc1-c2ccccc2)C(C)(C)C</chem>	
<chem>[Fe]I.[C-]#[O+].[C-]#[O+].[CH]1[CH][CH][CH][CH]1</chem>	

Tabla 1.1: Códigos SMILES y sus representaciones 2D según Sigma-Aldrich

mínimamente, el sistema de dibujado de los compuestos.

## 1.2. Objetivos

## 1.3. Estructura de la memoria

Código SMILES	Representación 2D
<chem>[Au+]( [CH3-] ) [P] ( C=1C=CC=CC1 ) ( C=2C=CC=CC2 ) C=3C=CC=CC3</chem>	
<chem>[Cl-] [Pd+2] 123 ( [Cl-] ) [CH]=4CC[CH]3=[CH]2CC[CH]41</chem>	
<chem>[Cl-] [Au+ ] [P] ( C ) ( C ) C</chem>	
<chem>[Cl-] [Au+ ] [P] ( C=1C=CC=CC1C=2C=CC=CC2 ) ( C(C)(C)C ) C(C)(C)C</chem>	
<chem>O#C [Fe+2] 1234 ( [I-] ) ( C#O ) [CH]=5[CH]4=[CH]3[CH-]2[CH]51</chem>	

Tabla 1.2: Códigos SMILES y sus representaciones 2D según SciFinder

Código SMILES	Representación 2D
<chem>C[Au].c1ccc(cc1)P(c2ccccc2)c3ccccc3</chem>	
<chem>Cl[Pd]Cl.C1CC=CCCC=C1</chem>	
<chem>Cl[Au].CP(C)C</chem>	
<chem>Cl[Au].CC(C)(C)P(c1ccccc1-c2ccccc2)C(C)(C)C</chem>	
<chem>[Fe]I.[C-]#[O+].[C-]#[O+].[CH]1[CH][CH][CH][CH]1</chem>	

Tabla 1.3: Códigos SMILES y sus representaciones 2D según Sigma-Aldrich



## Capítulo 2

# Estado del arte y fundamentos teóricos

Quizas sea mejor mover esta seccion justo despues de la introduccion para seguir con la tematica de la motivacion, y ya luego me meto con la gestion y planificacion

Puedo hacer una revision de la literatura existente hasta dia de hoy sobre el tema Usar SCOPUS para esto, con terminos tipo: "SMILESmolecule organometalic" (juntarlos o separarlos segun vea)

Hablar por aqui de la organometalica, cosas de dibujado de moleculas (los paquetes que hay), tutorial de SMILES, y representacion de moléculas) No se si meterlo aqui o en otro apartado, el diagrama de clases



## Capítulo 3

# Gestión y Planificación del proyecto

### 3.1. Metodología

### 3.2. Gestión de recursos

#### 3.2.1. Recursos humanos

#### 3.2.2. Recursos materiales

#### 3.2.3. Recursos software

### 3.3. Gestión de costes

#### 3.3.1. Coste de recursos humanos

Esto quizás dejarlo para el final, cuando tenga la cantidad total de horas trabajadas (aunque podría hacerlo con las "300" horas que se supone que le tengo que dedicar)

#### 3.3.2. Coste hardware

#### 3.3.3. Otros costes

#### 3.3.4. Presupuesto final

### 3.4. Análisis de riesgos



# Bibliografía

- [1] William J. Wiswesser. “107 Years of Line-Formula Notations (1861-1968)”. en. En: *Journal of Chemical Documentation* 8.3 (ago. de 1968), págs. 146-150. ISSN: 0021-9576, 1541-5732. DOI: 10.1021/c160030a007. URL: <https://pubs.acs.org/doi/abs/10.1021/c160030a007>.
- [2] Mario Krenn et al. “Self-referencing embedded strings (SELFIES): A 100 % robust molecular string representation”. en. En: *Machine Learning: Science and Technology* 1.4 (dic. de 2020), pág. 045024. ISSN: 2632-2153. DOI: 10.1088/2632-2153/aba947. URL: <https://iopscience.iop.org/article/10.1088/2632-2153/aba947>.
- [3] Mario Krenn et al. “SELFIES and the future of molecular string representations”. en. En: *Patterns* 3.10 (oct. de 2022). arXiv:2204.00056 [physics], pág. 100588. ISSN: 26663899. DOI: 10.1016/j.patter.2022.100588. URL: <http://arxiv.org/abs/2204.00056> (visitado 20-11-2022).
- [4] Alston Lo et al. *Recent advances in the Self-Referencing Embedding Strings (SELFIES) library*. arXiv:2302.03620 [physics]. Feb. de 2023. DOI: 10.48550/arXiv.2302.03620. URL: <http://arxiv.org/abs/2302.03620> (visitado 26-02-2023).
- [5] PubChem. *PubChem*. en. URL: <https://pubchem.ncbi.nlm.nih.gov> (visitado 01-02-2023).