



Universidad Internacional de la Rioja (UNIR)

Escuela Superior de Ingeniería y
Tecnología

Máster en Computación Cuántica

Procesamiento cuántico del
lenguaje natural: Revisión
sistemática de la literatura

Trabajo Fin de Estudios

Presentado por: Sergio Jiménez Fernández

Dirigido por: Francisco José Orts Gómez

Ciudad: Tomelloso, Ciudad Real

Fecha: día de mes de 2024

Índice de Contenidos

Resumen	v
Abstract	vi
1. Introducción	1
1.1. Motivación del tema a tratar	1
1.1.1. Computación cuántica	1
1.1.1.1. El cúbit y la superposición	2
1.1.1.2. Puertas cuánticas	6
1.1.1.3. Circuito cuántico	12
1.1.2. Procesamiento del lenguaje natural	14
1.1.2.1. Definición	15
1.1.2.2. Proceso de procesado	17
1.1.2.3. Aplicaciones	20
1.2. Planteamiento del trabajo	22
1.3. Estructura del trabajo	23
2. Contexto y estado de la técnica	24
3. Objetivos	27
3.1. Objetivos generales	27
3.2. Objetivos específicos	27
4. Desarrollo del trabajo	28
4.1. Planificación	30
4.1.1. Metodología	30
4.1.2. Guión del proceso	31
4.1.3. Preguntas de investigación	32
4.1.4. Cadena de búsqueda	34
4.1.4.1. Repositorios a consultar	34
4.1.4.2. Cadenas de búsqueda	36
4.1.5. Criterios de inclusión y exclusión	37
4.1.6. Extracción de información	38

4.2. Ejecución	39
4.2.1. Conteo por etapas	41
5. Resultados	45
5.1. Preguntas de investigación	45
5.1.1. PI_1 : Conceptos/algoritmos empleados	46
5.1.2. PI_2 : Aplicaciones e intenciones	48
5.1.3. PI_3 : Lenguajes investigados	50
5.2. Datos adicionales	52
5.2.1. Tipos de contribución	52
5.2.2. Países	53
5.2.3. Fechas	55
6. Conclusiones	57
7. Trabajo futuro	58
Bibliografía	59
Lista de Estudios Primarios (PSL)	64
Proyectos <i>software</i>	64
Contribuciones académicas	68
A. Apéndices	82
A.1. Diferencias entre SMS y SLR	82
A.2. <i>Script</i> de <i>Python</i>	82
A.3. Repositorio de <i>GitHub</i>	83
A.3.1. Justificación	83
A.3.2. Contenidos	83

Índice de Ilustraciones

1.1. Esfera de Bloch	3
1.2. Principales puertas de un cúbit	8
1.3. Principales puertas controladas de dos cúbits	10
1.4. Otras puertas cuánticas multi-cúbit	11
1.5. Aplicación de puertas para entrelazar dos cúbits	12
1.6. Diagrama con la relación entre AI, ML y NLP	17
1.7. Árbol sintáctico tras análisis sintáctico	20
4.1. Proceso para el estado del arte	29
4.2. Guión del proceso de ejecución	33
5.1. Gráficos circulares de conceptos fundamentales (PI1)	47
5.2. Histograma de lenguajes procesados (PI3)	51
5.3. Gráfico de barras con los países más activos	54
5.4. Evolución histórica global (por mes) de contribuciones	55

Índice de Tablas

1.1. Estados de cada eje principal de la Esfera de Bloch	4
1.2. Estados entrelazados de Bell	12
1.3. Principales categorías léxicas	18
4.1. Tipo de búsqueda por motor	37
4.2. Tabla de resultados crudos	41
4.3. Tabla de resultados procesados	42
4.4. Tabla de resultados tras el primer filtro	43
4.5. Tabla de resultados tras el segundo filtro	44
4.6. Cantidad de contribuciones de cada tipo	44
5.1. Tabla de intenciones/aplicaciones de cada estudio (PI2)	49
5.2. Tabla de tipos de contribuciones	52
A.1. Tabla comparativa SMS y SLR	82

Resumen

El presente trabajo aborda la intersección entre la computación cuántica y el procesamiento del lenguaje natural (conocido como *quantum natural language processing*), explorando su estado actual, implementaciones y aplicaciones. Se establecen objetivos generales y específicos para analizar cómo estas disciplinas pueden potenciarse mutuamente. El desarrollo del trabajo se estructura en forma de mapeo sistemático de la literatura (*systematic mapping study*), seguida de la ejecución y análisis de resultados. Se concluye con reflexiones sobre los hallazgos y se plantean direcciones futuras. Este estudio ofrece una visión integral de las posibilidades que emergen de la convergencia entre la computación cuántica y el procesamiento del lenguaje, identificando áreas clave para la investigación actual y futura.

Palabras clave: Computación cuántica, procesamiento del lenguaje natural, mapeo sistemático de la literatura.

Abstract

This dissertation presents the current intersection between quantum computing and natural language processing, known as quantum natural language processing, exploring its current state, implementations and applications. General and specific goals have been established to analyze how these disciplines can mutually power up. The study is structured as a Systematic Mapping Study, followed by execution and analysis of results. Finally, reflections on the findings are proposed, along with proposed next steps. In summary, this document offers an integrated point of vision on the possibilities emerging from merging quantum computing and natural language processing, highlighting key aspects for current and future investigations.

Keywords: Quantum computing, natural language processing, sistematic mapping study.

1. Introducción

La computación cuántica (*Quantum Computing* - QC) se encuentra actualmente en una época de crecimiento, en la que se está tratando de encontrar diferentes ámbitos y campos de estudio en los que esta pueda aportar una ventaja a la hora de acelerar los cómputos respecto a los algoritmos clásicos (Gyongyosi e Imre , 2019).

Una de estas ramas es el aprendizaje automático (*Machine Learning* - ML) y la inteligencia artificial en general (*Artificial Intelligence* - AI), en la que varios aspectos que ofrece la QC ofrecen una sinergia muy prometedora y digna de estudio (Biamonte *et al.* , 2017). De hecho, actualmente el aprendizaje automático cuántico (*Quantum Machine Learning* - QML) es una de las principales puntas de lanza en la investigación de en QC (Melnikov *et al.* , 2023).

Sin embargo, el campo de ML es muy frondoso e involucra a su vez muchas subramas dignas de estudio propio. Una de ellas es el procesamiento del lenguaje natural (*Natural Language Processing* - NLP) en el que se intenta integrar el lenguaje humano con el ML (Bird *et al.*, 2009).

Por tanto, dado que NLP es un campo del ML, una inquietud lógica sería buscar el punto en común con la QC. Esta combinación de campos se denomina procesamiento cuántico del lenguaje natural (*quantum* NLP - qNLP) y será el tema principal de esta disertación.

1.1. Motivación del tema a tratar

Para poder comprender la motivación de esta disertación, primeramente es necesario conocer y entender una serie de conceptos (paradigma de QC y campo del NLP) así como los retos que estos afrontan actualmente.

1.1.1. Computación cuántica

En los años 1980 y 1982, Paul Benioff y Richard Feynman propusieron las bases de lo que ahora es la QC. Por un lado, Benioff (1980) propuso un modelo de computador basado en la mecánica cuántica. Y por otro lado, Feynman (1982) dio una conferencia en la que abogaba por el uso de la misma naturaleza, refiriéndose a la física cuántica, para simular y entender procesos de la naturaleza.

A partir de entonces, investigadores de ámbitos muy diversos fueron aportando y contribuyendo para la consolidación de este nuevo paradigma de computación. Entre las principales aportaciones, encontramos la primera demostración de que un ordenador cuántico podría resolver un problema en menor tiempo que uno clásico (Deutsch, 1985). Posteriormente, se consiguió romper el algoritmo RSA utilizando la QC, lo que supuso un hito en el interés de la sociedad en este paradigma (Shor, 1997).

1.1.1.1. El cúbit y la superposición

Un cúbit es un análogo del bit clásico, solo que en lugar de que pueda tener valor 0 o 1 exclusivamente, el cúbit puede estar en una combinación lineal de estos estados. Con esto, podemos representar un cúbit matemáticamente mediante la Ecuación 1.1 (Hughes *et al.*, 2021).

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1.1)$$

donde identificamos estados cuánticos $|0\rangle$ y $|1\rangle$, que se corresponden con los valores clásicos 0 y 1 de los bits clásicos, y el $|\psi\rangle$ que representa la posible combinación lineal de los estados base ponderado por los coeficientes complejos α y β denominados amplitudes de probabilidad (Hughes *et al.*, 2021).

Esta combinación de los estados base es lo que se conoce como la famosa superposición, en la que, vagamente, se suele decir que el cúbit está en los dos estados a la vez. Concretamente, habrá una probabilidad del $|\alpha|^2$ de que, al medir el cúbit, el estado resultante sea $|0\rangle$, y de la misma manera, una probabilidad de $|\beta|^2$ de que se obtenga el estado $|1\rangle$ (Wilde, 2019).

Una vez se tiene preparado este estado superpuesto, es interesante obtener información de él, y poder así computar, es decir, medir. Aunque se explicará más en detalle posteriormente, la medición es un proceso delicado en la QC, dado que el estado del cúbit colapsa en uno de los estados posibles en la superposición (Cohen-Tannoudji *et al.*, 2005).

Cabe destacar también, que, debido al factor probabilístico de esta superposición, se aplica la regla de Born de normalización, en la que la suma de los cuadrados de las amplitudes debe dar siempre uno, independientemente del número de cúbits que tengamos.

Aunque, pese a la gran utilidad que presenta esta aproximación basada en álgebra lineal, también es muy útil recurrir en ocasiones a su representación matricial y vectorial. Para ello, se define cada estado base como un vector de dos dimensiones (una para

cada posible estado base), de este modo, el estado de la Ecuación 1.1 se representaría matricialmente de la siguiente forma (Hughes *et al.*, 2021):

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (1.2)$$

En la Ecuación 1.2 se puede ver qué asociamos la primera fila al estado $|0\rangle$ y la segunda al estado $|1\rangle$ y escribiremos la amplitud de probabilidad de cada uno. De este modo, también podríamos representar los estados base de la siguiente forma (Ecuación 1.3):

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (1.3)$$

Para concluir esta explicación, cabe resaltar la correspondencia entre la información que se necesita para computar (0 y 1) con los estados cuánticos que permiten la implementación física de los procesadores de QC (estados $|0\rangle$ y $|1\rangle$, entre otros).

La esfera de Bloch Una forma de visualizar el estado de un cúbit es utilizando la esfera de Bloch representada en la Figura 1.1. El estado $|\psi\rangle$ viene representado por el punto rojo de la superficie de una esfera, con un radio de longitud 1. De esta forma, podemos ver de forma muy directa hacia qué estado base (si el $|0\rangle$ o el $|1\rangle$) nos estamos acercando más.

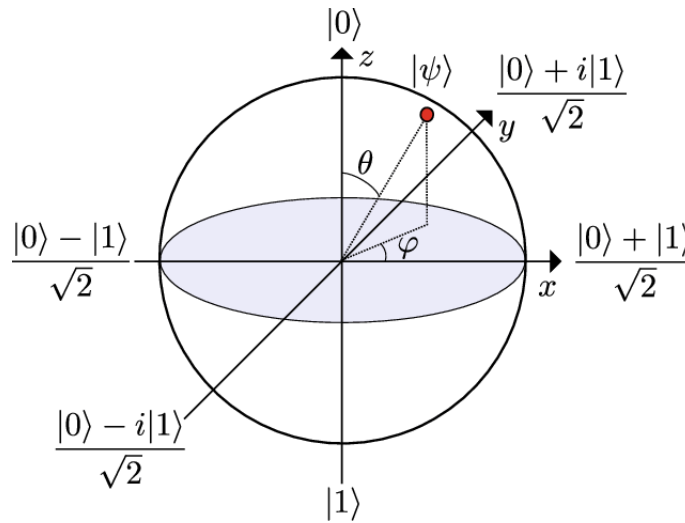


Figura 1.1: Esfera de Bloch para representar el estado de un cúbit. Fuente: (Kockum y Nori, 2019).

Precisamente, en esta esfera encontramos otros tres ejes que se asemejan bastante, en el sentido de que son perpendiculares entre sí, lo que se denominan bases ortonormales

(Nielsen y Chuang, 2011). De hecho, existen infinitas bases ortonormales, pero las más utilizadas son las que se corresponden con los ejes X , Y y Z de la Figura 1.1. La base que contiene los estados del bit clásico se denomina la base computacional y es la del eje Z de la esfera (Nielsen y Chuang, 2011).

Cabe destacar que, a partir de ahora, por convención, a no ser que se exprese lo contrario explícitamente, se utilizará la base computacional como referencia. Los estados de cada una de las bases mencionadas se resumen en la Tabla 1.1.1.1.

Eje de la Esfera		
X	Y	Z
$ +\rangle = \frac{ 0\rangle+ 1\rangle}{\sqrt{2}}$	$ +i\rangle = \frac{ 0\rangle+i 1\rangle}{\sqrt{2}}$	$ 0\rangle$
$ -\rangle = \frac{ 0\rangle- 1\rangle}{\sqrt{2}}$	$ -i\rangle = \frac{ 0\rangle-i 1\rangle}{\sqrt{2}}$	$ 1\rangle$

Tabla 1.1: Tabla explicativa con los estados ortogonales de los principales ejes de la Esfera de Bloch de un solo cúbit. Fuente: Elaboración propia.

Teniendo esto en cuenta, se puede vislumbrar como la superposición que definimos anteriormente es dependiente de la base con la que estemos trabajando.

Veamos esto con un ejemplo: pongamos que tenemos el estado de superposición equiprobable¹ $|+\rangle$ junto con la base computacional. Sin embargo, se cambia la base por la de los estados ortogonales del eje X (donde los estados son $|+\rangle$ y $|-\rangle$). Efectivamente, ya no es una superposición puesto que el estado pertenece a los estados de la nueva base.

La medición Con todo lo que se ha explicado sobre los posibles estados de un sólo cúbit, es lógico pensar que se trabaja con un espacio continuo, y dado que en la superficie de la esfera hay infinitos puntos, que se puede manejar información infinita.

Y, por desgracia, esto no ocurre. Y es que, existe un matiz que suele pasarse por alto, pero no es para nada insignificante: si es cierto que la capacidad de cómputo se ve empujada por esta capacidad, pero no se ha mencionado como extraer la información de estos cúbits.

De acuerdo con uno de los postulados de la mecánica cuántica, como consecuencia del factor probabilístico de esta, al medir, el estado de cualquier sistema cuántico colapsa, rompiendo la posible superposición que se pudiera tener (Cohen-Tannoudji *et al.*, 2005).

¹Tendremos un $\left(\frac{1}{\sqrt{2}}\right)^2 = 0,5 \rightarrow 50\%$ de probabilidad de encontrar el estado $|0\rangle$ y la misma para el estado $|1\rangle$.

Dicho con otras palabras, para el hipotético caso de que tengamos el estado de superposición $|+\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$, al medir, suponiendo que la salida de dicha medición es el estado $|1\rangle$, se habrá roto la superposición, y el estado del cúbit pasa a ser el $|1\rangle$ que medimos.

Es por esto, por lo que la manipulación de los cúbits con el paradigma de QC es ciertamente delicado, dado que hay que trabajar con ellos pero sin saber con concreción un estado base.

Como nota adicional, notar que es posible hacer mediciones con otras bases, en lugar de la base computacional del ejemplo mostrado anteriormente (Nielsen y Chuang, 2011).

Sistemas multi-cúbit En efecto, la potencia que ofrece un sólo cúbit comparada con la que ofrece un bit clásico es considerable. Pero, en la inmensa mayoría de ocasiones, un sólo bit no es suficiente para resolver problemas. Pues lo mismo ocurre con los cúbits. Es necesario poder utilizar varios de ellos a la vez, lo que aumenta considerablemente la capacidad de cómputo.

Para obtener un sistema cuántico con dos cúbits, se utiliza el producto tensorial. De esta forma, en el caso de que tengamos dos cúbits en estados arbitrarios $|\psi_1\rangle = \begin{bmatrix} \alpha_1 \\ \beta_1 \end{bmatrix}$ y $|\psi_2\rangle = \begin{bmatrix} \alpha_2 \\ \beta_2 \end{bmatrix}$, el estado del sistema global se calcularía de acuerdo con la siguiente Ecuación (Wilde , 2019):

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{bmatrix} \alpha_1 \\ \beta_1 \end{bmatrix} \otimes \begin{bmatrix} \alpha_2 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} \alpha_1 \otimes \begin{bmatrix} \alpha_2 \\ \beta_2 \end{bmatrix} \\ \beta_1 \otimes \begin{bmatrix} \alpha_2 \\ \beta_2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \alpha_1\alpha_2 \\ \alpha_1\beta_2 \\ \beta_1\alpha_2 \\ \beta_1\beta_2 \end{bmatrix} \quad (1.4)$$

De forma análoga a la interpretación de los estados de un sólo cúbit en forma de vector, el estado resultante de la Ecuación 1.4 se expresaría como combinación lineal de la siguiente forma:

$$|\psi_1\rangle \otimes |\psi_2\rangle = \alpha_1\alpha_2 (|0\rangle \otimes |0\rangle) + \alpha_1\beta_2 (|0\rangle \otimes |1\rangle) + \beta_1\alpha_2 (|1\rangle \otimes |0\rangle) + \beta_1\beta_2 (|1\rangle \otimes |1\rangle) \quad (1.5)$$

Pero, por simplicidad, se suele obviar el producto tensorial, lo que reduce la Ecuación 1.5 a la forma:

$$|\psi_1\psi_2\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle \quad (1.6)$$

donde α , β , γ y δ son las amplitudes de probabilidad de los estados $|00\rangle$, $|01\rangle$, $|10\rangle$ y $|11\rangle$ respectivamente.

Nótese cómo ha evolucionado la base computacional en función al producto tensorial que se mencionó anteriormente. La base computacional en forma vectorial se representaría de la siguiente forma (Wilde , 2019):

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (1.7)$$

Esto es completamente generalizable para n cúbits, en lugar de dos. Simplemente, habría que multiplicar con el producto tensorial $n - 1$ veces en lugar de solamente una. De esta forma, para n cúbits, tendríamos una base computacional de tamaño 2^n , que pueden ser superpuestas todas en un estado. Es aquí donde reside una de las ventajas de cómputo de la QC.

Cabe destacar, que para los sistemas de varios cúbits, no es posible representar un estado multi-cúbit con una sola esfera de Bloch, sino que tendríamos que representar tantas esferas de Bloch como cúbits tengamos.

1.1.1.2. Puertas cuánticas

Retomando la referencia de la computación clásica, la cual (en un nivel muy bajo de abstracción), opera sobre los cúbits utilizando puertas lógicas. Pues precisamente, la QC se asemeja bastante de nuevo, dado que opera con los cúbits mediante puertas, en esta ocasión, cuánticas.

Uni-cúbit Las puertas cuánticas que afectan a un sólo cúbit, como su propio nombre da a entender, permiten manipular el estado de un solo cúbit. Formalmente, definimos una puerta cuántica (de un solo cúbit) cómo un operador unitario que actúa sobre un espacio de dos dimensiones (Kaye *et al.* , 2007).

Geométricamente, se puede entender la aplicación de una puerta genérica arbitraria U como una rotación de la flecha en la Esfera de Bloch (Kaye *et al.* , 2007), lo que la llevaría

a apuntar a otra dirección, es decir, otro estado distinto.

Con otras palabras, una puerta U aplicada sobre un estado $|\psi\rangle$ cualquiera, resulta en otro estado cuántico $U|\psi\rangle$ (Kaye *et al.*, 2007), lo que concuerda con otro de los postulados de la mecánica cuántica (Zettili, 2009).

Por otro lado, desde un punto de vista matemático, una puerta cuántica es una matriz unitaria que, para el caso de ún solo cúbit, es de tamaño 2×2 (Barenco *et al.*, 1995), tal y como se muestra en la Ecuación 1.8:

$$U = \begin{bmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{bmatrix} \quad (1.8)$$

De esta forma, aplicado sobre un estado cuántico arbitrario $|\psi\rangle$ como se mencionaba anteriormente, para saber la expresión matemática del nuevo estado, simplemente habría que realizar una multiplicación de matrices. Como vemos en la Ecuación 1.9, quedaría de la siguiente forma:

$$U|\psi\rangle = \begin{bmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} u_{00}\alpha + u_{01}\beta \\ u_{10}\alpha + u_{11}\beta \end{bmatrix} = \begin{bmatrix} \alpha' \\ \beta' \end{bmatrix} = \alpha'|0\rangle + \beta'|1\rangle = |\psi'\rangle \quad (1.9)$$

Por poner nombres, algunas de las puertas más conocidas de ún solo cúbit son las siguientes:

- Puerta identidad (I): Equivalente a no aplicar ninguna puerta, deja el estado del cúbit como estaba. Matricialmente, se representa de acuerdo con la Ecuación 1.10 (Wilde, 2019):

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (1.10)$$

- Puertas Pauli (X, Y, Z): Rotan el estado del cúbit π radianes sobre cada uno de los tres ejes de la Esfera de Bloch. Concretamente, tenemos las puertas Pauli X, Y y Z . Las matrices correspondientes para cada una se muestran en la Ecuación 1.11 (Wilde, 2019):

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (1.11)$$

- Puerta Hadamard (H): Rota el estado del cúbit un ángulo de π sobre el eje Z , de tal forma que lo sitúa sobre el plano XY . Concretamente, si se aplica la puerta H al estado $|0\rangle$, lo sitúa en el estado $|+\rangle$ del eje X , creando una superposición equiprobable. Del mismo modo, matemáticamente, esta puerta se representa de la forma que se muestra en la Ecuación 1.12 (Wilde , 2019):

$$I = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (1.12)$$

Las representaciones gráficas de estas puertas se reflejan en la Figura 1.2.

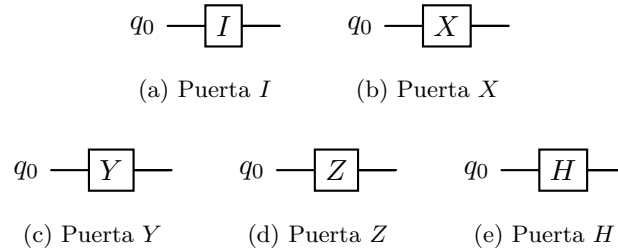


Figura 1.2: Representación visual de las principales puertas cuánticas aplicadas sobre un solo cúbit. Fuente: Elaboración Propia.

Otro requisito que han de tener todas las puertas cuánticas, ya no solo las de un cúbit, es que deben ser reversibles (Wilde , 2019). Esto viene, matemáticamente, como consecuencia de la unitariedad de los operadores que representan formalmente las puertas cuánticas (Wilde , 2019).

Con otras palabras, esto significa que, al aplicar una puerta arbitraria, a partir de la aplicación de otro conjunto de puertas (pudiendo ser incluso la misma que se aplicó), el estado del cúbit debe quedarse como estaba antes de aplicar la primera puerta.

Cabe destacar, que para llevar a cabo la medición que se mencionó anteriormente, se realiza mediante una puerta de un único cúbit. La peculiaridad de esta puerta es que, al contrario que las puertas cuánticas que se han presentado que debían de ser operadores unitarios, la operación de medición es la única que no cumple la condición de reversibilidad.

Esto se debe a que la evolución del estado de un sistema cuántico ante una medición de este, específicamente de cara a la QC nos referimos a los cúbits, es un proceso que no es unitario (Kaye *et al.* , 2007). Y, dado que la unitariedad lleva a la reversibilidad (Wilde , 2019), la medición no es reversible.

Esto puede interpretarse como la aplicación en la QC del colapso del sistema cuántico al realizar una medición sobre este, haciendo que el estado del cúbit cambie de manera irreversible al realizar una medición sobre este.

Multi-cúbit Al igual que en los circuitos clásicos, las puertas lógicas no solo se aplican sobre una sola entrada, es la aplicación de puertas con dos entradas lo que potencia el cómputo clásico. Pues con el análogo cuántico, ocurre lo mismo con las puertas cuánticas aplicadas sobre más de un cúbit, o lo que se denominará en este documento por simplicidad, las puertas multi-cúbit.

Una diferencia respecto a los circuitos clásicos y las puertas de más de un bit, donde solo es posible hacer interaccionar a dos bits por cada puerta, las puertas cuánticas multi-cúbit permiten la interacción de, como mínimo, dos cúbits.

Matemáticamente, para definir una puerta cuántica que involucre a n cúbits, recordando que una puerta de un sólo cúbit se define con una matriz 2×2 , necesitaremos una matriz de $2^n \times 2^n$.

Es precisamente la aplicación de puertas multi-cúbit la que permite conseguir el conocido efecto cuántico de entrelazamiento (de la que se hablará más en detalle posteriormente), en el que no se puede describir el estado en función de las descripciones individuales de cada uno de los cúbits involucrados (Li *et al.*, 2010).

La principal forma de involucrar dos cúbits es mediante las puertas controladas. Una puerta controlada, o dicho de manera más formal, una puerta *Controlled* – U (o CU por reducir terminología) es una puerta cuántica de dos cúbits con dos principales roles para cada uno de los cúbits (Kaye *et al.* , 2007): un cúbit de control cuyo estado hará que se aplique la puerta U sobre el cúbit objetivo.

Visto de manera más rigurosa, en forma de matriz de $2^2 \times 2^2 = 4 \times 4$, una puerta controlada arbitraria CU se define de acuerdo con la Ecuación 1.13 (Rieffel y Polak , 2011).

$$CU = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{i\theta} & 0 \\ 0 & 0 & 0 & e^{i\theta} \end{bmatrix} \quad (1.13)$$

donde $e^{i\theta}$ es un número complejo y, la submatriz $\begin{bmatrix} e^{i\theta} & 0 \\ 0 & e^{i\theta} \end{bmatrix}$ se corresponde con el efecto de la puerta U independientemente del cúbit de control.

De esta forma, para un sistema de dos cúbits $|\Phi\rangle$, aplicación de la puerta CU que se acaba de definir tendría la siguiente formulación (Rieffel y Polak , 2011):

$$CU|\Phi\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) \longrightarrow \frac{1}{\sqrt{2}} (|00\rangle + e^{i\theta}|11\rangle) \quad (1.14)$$

La puerta más común de este tipo es la puerta $CNOT$, que tiene la siguiente representación matricial, de acuerdo con la Ecuación 1.13 (Nielsen y Chuang, 2011):

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.15)$$

De hecho, aplicando el concepto de la submatriz que define la puerta U que se pretende controlar, para la puerta $CNOT$, la puerta de un cúbit a controlar es la puerta Pauli X (Wilde , 2019).

En la Figura 1.3 se muestran también las representaciones visuales, tanto de la puerta arbitraria CU como de la puerta $CNOT$.

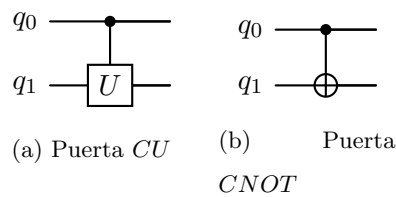


Figura 1.3: Representación visual de las puertas controladas aplicadas sobre dos cúbits.
Fuente: Elaboración Propia.

Pero, como se mencionó anteriormente, también se pueden construir puertas que involucren a más de dos cúbit. Para esto, se recurre a este esquema de puerta controlada: controlando puertas controladas.

El primer caso de esta generalización es la puerta Toffoli, y no es más que controlar la aplicación de la puerta $CNOT$. Dicho de otra manera, tendremos dos cúbits de control y un cúbit de objetivo, llegando a involucrar tres cúbits con un sola puerta (Nielsen y

Chuang, 2011). Y esto es generalizable para n cúbits, lo que, en ocasiones, se conoce como puertas multi-controladas.

Por otro lado, respecto a puertas multi-cúbit, también se destaca la puerta *SWAP*, que afecta a dos cúbits, intercambiando los estados de estos dos. Matemáticamente, ante un estado de un sistema cuántico de dos cúbits de la forma $|\psi_1\rangle|\psi_2\rangle$, al aplicar la puerta *SWAP*, el estado resultante será $SWAP(|\psi_1\rangle|\psi_2\rangle) = |\psi_2\rangle|\psi_1\rangle$ (Rieffel y Polak, 2011).

En la Figura 1.4 se muestran las correspondientes representaciones visuales tanto de la puerta Toffoli y de la puerta *SWAP*.

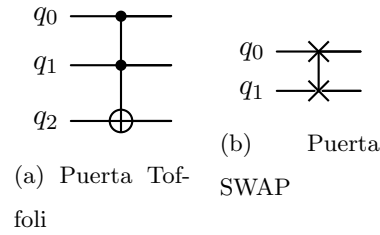


Figura 1.4: Representación visual de las puertas cuánticas Toffoli y *SWAP*. Fuente: Elaboración Propia.

El entrelazamiento Habiendo visto estos conceptos, resulta relevante presentar otro efecto propio de la mecánica cuántica aplicado sobre los cúbits: el entrelazamiento.

El entrelazamiento, desde un punto de vista mecánico-cuántico, se describe como la existencia de un estado global de un sistema cuántico que no puede ser descrito desde un punto de vista individualista, es decir, componente por componente (Li *et al.*, 2010).

Con los conocimientos que se han adquirido hasta este punto, se puede entender este efecto de entrelazamiento aplicado para dos cúbits.

Las puertas necesarias para lograr este entrelazamiento con dos cúbits serán: una puerta Hadamard aplicada sobre un cúbit, y una puerta *CNOT* tal que el cúbit de control sea el mismo sobre el que aplicó la puerta *H* (Symons *et al.*, 2023).

Esto, de forma visual en base a sus representaciones gráficas de cada una de las puertas se muestra en la Figura 1.5.

Para entender mejor el efecto del entrelazamiento, se plantea el siguiente desarrollo matemático:

1. Se parte de un **estado inicial** $|\psi_1\rangle = |00\rangle$.

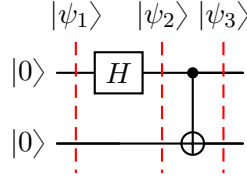


Figura 1.5: Representación visual de aplicación de puertas necesarias para poder entrelazar dos cúbits. Fuente: Elaboración Propia.

2. Se aplica una **puerta Hadamard** al primer cúbit, lo que modifica el estado tal como se refleja en la Ecuación 1.16:

$$|\psi_2\rangle = H|0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |10\rangle) \quad (1.16)$$

3. Se aplica la **puerta controlada, CNOT**, teniendo el primer cúbit como control y el segundo como objetivo. Esto, matemáticamente, se refleja en la Ecuación 1.17:

$$|\psi_3\rangle = CNOT|\psi_2\rangle = \frac{1}{\sqrt{2}} CNOT (|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) \quad (1.17)$$

Una vez se tiene el estado final $|\psi_3\rangle$, es momento de recordar la definición vectorial de los estados cuánticos. Es entonces, donde se puede observar que es imposible expresar este estado como la combinación lineal de los vectores de los estado de la base computacional. Esto implica que no podamos describir este estado cuántico en base a sus cúbits por separado, lo que encaja con la definición mecánico-cuántica del entrelazamiento.

De hecho, este estado es uno de los cuatro posibles estados entrelazados que se pueden construir con dos cúbits, denominados los estados de Bell (Symons *et al.*, 2023). En concreto, los cuatro estados de Bell son los que se resumen en la Tabla 1.2 (Rieffel y Polak, 2011).

$$\begin{array}{l|l} |\Phi_+\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) & |\Phi_-\rangle = \frac{1}{\sqrt{2}} (|00\rangle - |11\rangle) \\ \hline |\Psi_+\rangle = \frac{1}{\sqrt{2}} (|01\rangle + |10\rangle) & |\Psi_-\rangle = \frac{1}{\sqrt{2}} (|01\rangle - |10\rangle) \end{array}$$

Tabla 1.2: Estados entrelazados que se pueden construir con dos cúbits: los estados de Bell. Fuente: Elaboración Propia.

1.1.1.3. Circuito cuántico

Un circuito cuántico se define como un conjunto de puertas cuánticas conectadas por cables o hilos, que definen cada uno de los cúbits utilizados para la computación, que

termina con el mismo número de cúbits con los que empieza (Shende *et al.*, 2005).

De acuerdo con esta definición, un claro y sencillo ejemplo que ya ha sido explicado es el circuito cuántico aplicado para entrelazar dos cúbits y lograr uno de los estados de Bell, que es el que se muestra en la Figura 1.5.

Visto desde otro enfoque, esta serie de puertas cuánticas pueden ser entendidas como un algoritmo cuántico (Nielsen y Chuang, 2011), dado que no deja de ser una serie de instrucciones -las puertas cuánticas- aplicadas sobre un conjunto de unidades de información -los cúbits- con la finalidad de obtener unos resultados -mediante la medición- que resuelva un problema determinado.

De hecho, teniendo en cuenta esta definición, surge la idea de los circuitos cuánticos como modelo de computación *per se*, y que además, ofrece una ventaja en lo que respecta a la complejidad de las soluciones a ciertos problemas, llegando a una escala exponencial de mejora (Nielsen y Chuang, 2011). También se le conoce a este paradigma la QC basada en puertas (Kwon *et al.* , 2021).

Entrando más en este concepto de circuito cuántico como paradigma de computación, Nielsen y Chuang (2011) detalla cinco características relevantes que describen este modelo. Estos son:

1. **Recursos clásicos:** Aunque no lo parezca, la QC y la computación clásica no están destinados a competir entre ellos como rivales. El punto interesante es precisamente el contrario, tratar de encontrar y acotar el dominio de problemas donde cada uno de ellos resuelva problemas (o partes de ellos) de la forma más eficiente.

Esto es lo que se conoce como computación híbrida cuántico-clásica, y es precisamente una de las formas en las que se está habilitando el avance e integración de la QC en varios tipos de soluciones (Rosmanis , 2024).
2. **Un espacio de estados adecuado:** Es decir, construir un espacio de estados empleando un conjunto de cúbits tal que todos los posibles estados que se pueden lograr con estos permitan codificar la información de entrada para el cómputo a realizar.
3. **Capacidad para inicializar en la base computacional:** Como se ha mencionado anteriormente, existen infinitas bases ortogonales con las que podemos modelar información en un cúbit. Sin embargo, el paradigma de QC basado en puertas se acoge a la base computacional, y por ende, hay que ser capaces de codificar la información de entrada del algoritmo en la base computacional en n pasos.

4. **Capacidad de aplicar puertas cuánticas:** Para poder computar, es necesario ser capaces de aplicar las puertas cuánticas sobre el conjunto de cúbits sobre el que se ha codificado la información en la base computacional, aunque no necesariamente se opere en esta base.
5. **Capacidad de medir en la base computacional:** Aunque se puedan aplicar puertas cuánticas en diferentes bases, es obligatorio que la salida de las mediciones sobre los cúbits del algoritmo estén en la base computacional.

El *quantum annealing* Finalmente, cabe mencionar que el paradigma basado en puertas no es el único modelo de QC que existen. El otro principal modelo a destacar (aunque haya varios más en los que no entraremos en detalle por la poca relación con esta disertación) es el *quantum annealing*.

El *quantum annealing* es un modelo computacional cuántico restringido a la QC adiabática, enfocado únicamente a resolver problemas de optimización combinatoria (Venegas-Andraca *et al.*, 2018).

1.1.2. Procesamiento del lenguaje natural

Allá por la década de los 50, el NLP empezó a abrirse paso en el ámbito de la traducción automática (Jones, 1994), donde se destaca un algoritmo (ciertamente elemental) de traducción de textos en ruso al inglés, con un vocabulario de 250 palabras y seis reglas gramaticales (Hutchins, 2004).

Posteriormente, en las décadas de los 60 y los 70, comienza una segunda fase del NLP, siendo enmarcado ya como un campo dentro de la AI, siendo ya mencionado por el mismo Minsky (1997) por sus labores en traducción automática.

Este fue evolucionando tanto teórica como prácticamente en diferentes ámbitos y campos de investigación, llegando al punto en el que el NLP es uno de los principales pilares de los *large language models*, entre los que destacamos hitos de la AI como *ChatGPT*², Gemini³ o Claude⁴ entre otros, y que se están arraigando como herramientas necesarias en nuestro día a día, mejorando nuestra productividad de manera considerable.

²<https://chatgpt.com/?oai-dm=1>

³<https://gemini.google.com/?hl=es-ES>

⁴<https://claude.ai/login?returnTo=%2F%3F>

1.1.2.1. Definición

Brevemente, se puede definir el procesamiento del lenguaje natural como aquellas técnicas computacionales (algoritmos) con la finalidad de aprender, entender y reproducir el lenguaje natural del ser humano (Hirschberg y Manning , 2015).

El NLP no es ni mucho menos unidisciplinar, más bien al contrario, puesto que los principales factores que permiten el crecimiento de este campo vienen de la mano de cuatro diferentes campos de conocimiento. Los cuatro factores que están permitiendo la evolución del NLP relacionado con cuatro campos diferentes son (Hirschberg y Manning , 2015):

1. **Potencia de cómputo:** Es crucial el crecimiento de las capacidades de cómputo para poder ver y descubrir nuevas capacidades en NLP, desde el nivel de usuario o desarrollador que investiga en equipos relativamente pequeños, hasta el nivel más alto, llegando a centros de computación de altas prestaciones.
2. **Disponibilidad de datos lingüísticos:** Debido a la estrecha relación entre QNLP y ML, los datos son un componente fundamental, tanto su disponibilidad como la calidad de estos. De aquí precisamente parte la relación con campos como la ingeniería y la analítica de datos.
3. **Evolución de algoritmos de ML:** Del hilo del punto anterior, el propio avance y propuesta de nuevos y mejores algoritmos de ML permitirán, por su relación directamente proporcional, un avance en el NLP, estrechando el lazo con la ciencia de datos.
4. **Mejor comprensión de la estructura lingüística del humano:** Por supuesto, si hablamos de lenguaje natural, es imprescindible la labor del campo de la lingüística para la evolución del NLP.

Esta multidisciplinaridad es la que hace que el NLP sea atractivo, aparte de los investigadores y practicantes de las ciencias de computación, profesionales de otros ámbitos como la psicología y filosofía (Khurana *et al.* , 2023).

AI, ML y NLP Para terminar de definir el NLP, es interesante contextualizarlo dentro del campo de la AI, dado que muchas veces los términos ML, AI y NLP suelen usarse

de forma indistinguida de forma errónea. Para esto, se propone definir, de acuerdo a la literatura, AI y ML y ver que relación presenta con NLP.

1. **Inteligencia Artificial (AI):** De acuerdo con Russel y Norvig (2010), la AI es tan amplia que puede definirse de varias formas desde diferentes puntos de vista, y que resume en base a testimonios y citas de otros reconocidos autores en el campo de la IA:

- La tarea de hacer a los ordenadores pensar, a modo de máquinas con mentes. Lo que Russel resume como “pensar de forma humana”.
- El estudio de las facultades mentales a través del uso de modelos computacionales, condensado con el término “pensar de forma racional”.
- El arte de construir máquinas que puedan llevar a cabo tareas que requieran de inteligencia humana. Lo que parafrasea como “actuar de forma humana”.
- El estudio del diseño de agentes inteligentes, resumido como “actuar de forma racional”.

2. **Aprendizaje automático (ML):** Según Marsland (2014), el aprendizaje automático trata de hacer a las máquinas (los ordenadores) modificar o adaptar sus acciones (acciones que dependen del problema a resolver) de modo que estas acciones mejoran la precisión de la máquina.

Dicho de un modo más metafórico, la AI permite a las máquinas hacer cosas de humanos, mientras que el ML permite a las máquinas aprender a hacer cosas de humanos.

En la Figura 1.6 se muestra, en forma de diagrama de Venn, la relación entre AI, ML y NLP. De esta relación se pueden destacar dos aspectos relevantes:

- La AI no solo es ML, incluye otras áreas como los sistemas inteligentes o los sistemas basados en el conocimiento (Russel y Norvig, 2010).
- El NLP efectivamente entra dentro del paraguas de la AI, con puntos en común con el ML, pero también tiene sus aportaciones propias, que viene de la mano de la multidisciplinaridad (Khurana *et al.*, 2023).

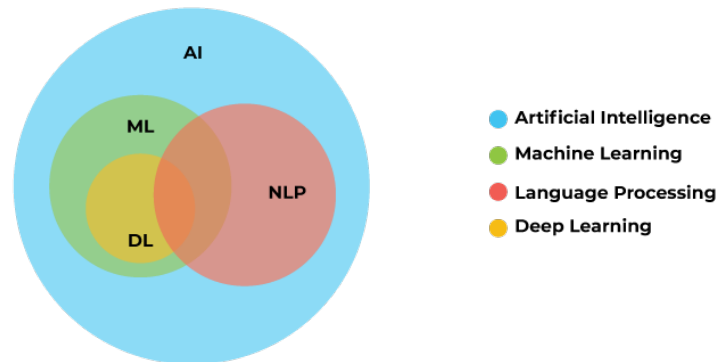


Figura 1.6: Diagrama de Venn con la relación entre AI, ML y NLP. Fuente: (Bolaños , 2021).

1.1.2.2. Proceso de procesado

Una vez se ha contextualizado de forma correcta el NLP, un siguiente paso interesante de cara a una mejor comprensión de este campo es ver los diferentes pasos o niveles del lenguaje sobre los que el NLP puede actuar.

Análisis fonológico Aunque no se entrará en detalle en este paso, merece la pena mencionar que, en varias ocasiones, la fonología es una etapa interesante. Esto se debe a que es el punto de entrada para aquellos algoritmos de NLP con entrada de voz (Liddy , 2001).

Para procesar la voz, realmente, lo que se aplica es un análisis y codificación de las ondas sonoras en base a ciertas reglas estudiadas gracias, en gran parte, al estudio fonológico del lenguaje (Liddy , 2001).

Análisis morfológico Se puede definir el análisis morfológico como el estudio de la naturaleza de las palabras y tratar de encontrar su forma más reducida, lo que se denominan los morfemas. Los morfemas son la unidad más pequeña de significado (Liddy , 2001).

Y es que, aunque pueda parecer ciertamente trivial, es un paso muy útil para que las máquinas puedan entender que el significado de la palabra *frutería* está muy relacionado con la palabra *fruta*, lo que permitirá relacionar conceptos y sacar conclusiones (Bird *et al.*, 2009).

De lo contrario, si no se realizara este paso previo, habría que cotejar un corpus de palabras enorme para casos que no lo necesiten. Un ejemplo puede ser tratar de desarrollar

un modelo de extracción de temáticas: si no realizamos este paso previo, todo un texto estará utilizando palabras con la misma raíz, pero con diferente escritura, lo que hace complicado extraer un tema común de no ser precisamente por el análisis morfológico.

Análisis léxico Después de un análisis morfológico, un punto de vista considerablemente importante de cara a dar valor a los textos y documentos es realizar un correcto análisis léxicos. El principal objetivo de esta etapa léxica es tratar de extraer el significado de palabras individuales (Liddy , 2001).

El análisis léxico implica ser capaces de asignar a cada palabra del documento de entrada la categoría léxica, o lo que se conoce como etiquetado de partes del discurso (*Part-Of-Speech tagging* - *POS tagging*) (Bird *et al.*, 2009).

En la Tabla 1.3 se muestran las principales etiquetas o categorías léxicas junto con algunos ejemplos para una mejor comprensión de cada una de las categorías.

Categoría Léxica	Ejemplos
Adjetivo	Nuevo, bueno, alto, especial, grande, local
Adverbio	Realmente, aquí, acaso, jamás, mucho, además
Sustantivo	Año, pueblo, casa, maceta, planta, mesa
Nombre propio	España, Calatrava, Daniel, María, José Luis, Tomelloso
Pronombre	Yo, tú, ella, nosotros, vosotros, ellos
Preposición	A, ante, bajo, con, contra, desde
Verbo	Hacer, patrullar, trabajar, teclear, jugar, escribir

Tabla 1.3: Principales categorías léxicas a tener en cuenta en la fase de *POS tagging*.
Fuente: Elaboración Propia.

Análisis sintáctico Una vez se han analizado las palabras, una a una, es relevante también subir el punto de vista y tratar de analizar las estructuras sintácticas. Y es que, este paso se enfoca a entender la estructura gramatical de las frases que componen el documento de entrada (Liddy , 2001).

En efecto, el análisis sintáctico se enfoca en la extracción de información a partir del estudio de las frases. Para ello, se descompondrán las frases que se encuentren en los datos de entrada para poder analizarla en función de sus componentes (Mote , 2002) (componentes como el sujeto, el verbo, complemento directo, objeto pasivo, etc.).

Los principales componentes necesarios para un análisis sintáctico son dos (Liddy , 2001):

1. Una gramática que describa de una manera formal las posibles estructuras sintácticas pertenecientes al lenguaje que se está estudiando (independientemente de la finalidad del estudio).

En este campo, resultó muy relevante la contribución de (Chomsky , 1959), en la que estableció la muy conocida jerarquía gramatical de Chomsky.

Esta jerarquía no era más que una descripción de 4 tipos de gramáticas en forma de teoría de conjuntos. De esta forma, cada tipo pertenecía al tipo anterior, poseyendo de características más restrictivas a otras menos restrictivas.

2. Un parser que pueda llevar a cabo el análisis sintáctico en función del tipo de gramática establecida (Liddy , 2001).

Este parser, junto con la gramática definida, dará como resultado una representación digital de la frase de entrada, en términos de su estructura sintáctica (Bird *et al.*, 2009).

Una de las formas más comunes de almacenar esta información de forma computarizada es a través de árboles sintácticos, en los que se relaciona los diferentes elementos de la gramática, desde la definición formal hasta los términos concretos extraídos del análisis léxico.

En la Figura 1.7 se muestra un ejemplo para visualizar mejor la labor de estos árboles sintácticos. Concretamente, con la frase en inglés: “They expected her election in September”.

Cabe destacar además, sobre esta etapa de análisis sintáctico, que es la fase con un estudio más profundo en la literatura (Mote , 2002).

Análisis semántico Hasta este punto, en el procesamiento de un texto arbitrario, se dispone de identificación de las palabras individualmente (análisis léxico) y la estructura gramatical que las relaciona para componer una frase (análisis sintáctico).

Sin embargo, falta una etapa crucial para poder aprovechar esta información extraída: asignar significado a dichas estructuras de información, y este es precisamente el principal objetivo del análisis semántico (Liddy , 2001).

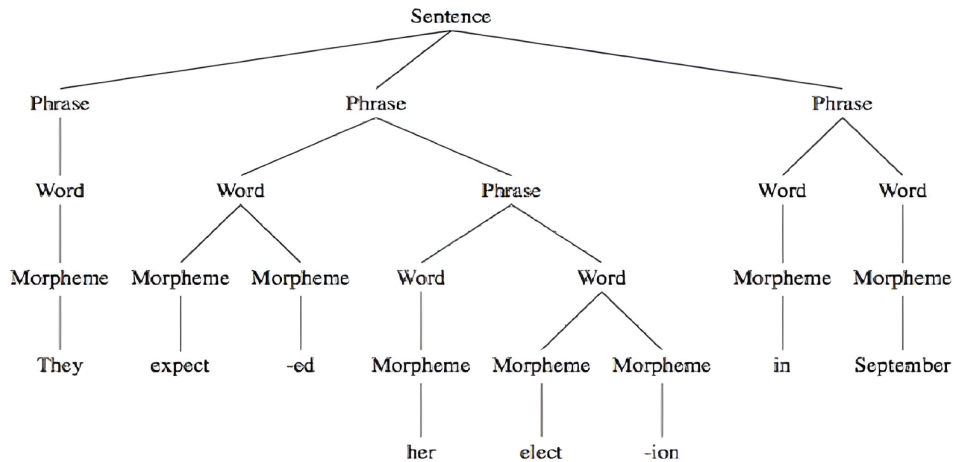


Figura 1.7: Árbol sintáctico obtenido como resultado del análisis sintáctico de una frase demostrativa en inglés. Fuente: (Hirst , 2012).

Aunque no queda muy clara la interfaz entre análisis sintáctico y análisis semántico (Mote , 2002), implica la realización de una tarea crucial, incluso con labores, no solo de asignación de significado, sino también de desambiguación. Y es que, como bien es sabido, una palabra puede tener muchos significados (Liddy , 2001).

Es precisamente por la aplicación secuencial de todas estas etapas en el procesamiento del lenguaje las que contribuyen a poder sacarle partido al lenguaje a través de un enfoque computacional.

Justamente, si se recuerda la definición anterior de AI, queda más clara la relación entre este y el NLP: poder analizar los textos, extraer información y aplicarla para diferentes tareas a través de máquinas.

De aquí la importancia de tratar y explicar (aunque sea de forma relativamente superficial) las diferentes y principales etapas en un ciclo de procesamiento del lenguaje natural y poder contextualizar su alcance y relaciones con otros campos.

1.1.2.3. Aplicaciones

Al hilo del párrafo anterior, es un momento oportuno para mostrar algunas de las muchas aplicaciones que se han o que se están llevando a cabo con el NLP, principalmente con técnicas de ML y AI.

Concretamente, se mostrarán tres aplicaciones que se consideran interesantes: traducción automática, análisis de sentimientos y generación de texto. Aunque, por supuesto,

existen otras muchas aplicaciones que, por motivos de extensión, no abordaremos en esta disertación.

Traducción automática Como se mencionó al principio de esta Subsección, la traducción automática de textos fue una de las primeras aplicaciones del NLP.

Como el propio nombre indica, se define como el proceso de traducción automática de textos, de un lenguaje humano a otro, por ejemplo, de inglés a español (Gupta , 2014).

Una peculiaridad de este campo digna de destacar es que implica una complejidad en todas las etapas, lo que se debe a la combinación de varios idiomas en el procesado de lenguaje (Gupta , 2014).

Algunas aportaciones implicadas en la traducción automática destacables y recientes se dejan a modo de futura indagación para el lector:

- Yuxiu (2024) y Wang (2024) con recientes contribuciones de la traducción automática para la enseñanza, desde clases de inglés hasta clases del grado de traducción.
- Zhu *et al.* (2024) donde muestra una interesante aportación para la traducción automática aportando, como datos auxiliares de entrada, información visual.

Análisis de sentimientos Otra de las aplicaciones más relevantes del NLP es la detección y extracción de información subjetiva extraída de conjuntos de datos textuales (Gupta , 2014).

Dicho más formalmente, el análisis de sentimientos (también denominado minado de opiniones) es un estudio computacional, ligado al NLP, de las opiniones, actitudes y emociones (Medhat *et al.* , 2014).

Dada la alta componente social de esta aplicación, las contribuciones acogidas bajo esta temática resultan muy interesantes para diferentes utilidades. Algunas más recientes y destacables son:

- Breve *et al.* (2024) y Koonchanok *et al.* (2024) muestran estudios de opiniones colectivas sobre temas de actualidad en los que el punto de vista social es relevante en su evolución y desarrollo. En concreto, analizan “tweets” sobre la guerra entre Rusia y Ucrania y sobre el uso ChatGPT, respectivamente.
- Farhoudinia *et al.* (2024) es una instancia de una de las ramas del análisis de sentimientos, además de la extracción de opinión colectiva: la detección de textos deter-

minados. En concreto, esta contribución trata de detectar “tweets” con información falsa sobre la COVID-19.

Generación de texto En pocas palabras, la generación de texto, como la propia denominación indica, consiste en generar texto de forma computarizada.

En concreto, acogiendo el curioso e interesante enfoque de Gupta (2014) para definir la generación de texto, se trata de la traducción de la información almacenada en los ordenadores involucrados en la labor de NLP a lenguaje humano.

De nuevo, veamos algunas instancias relevantes en esta rama del NLP para completar mejor esta definición teórica de la generación de texto:

- Neshaei *et al.* (2024) propone la integración de NLP con el mundo académico, concretamente, apoyando la rigurosa revisión por pares mediante técnicas de generación de texto para obtener una descripción textual de la revisión.
- Freitas *et al.* (2023) implementa la generación automática de ejercicios de programación para estudiantes de ciencias de la computación, siendo los principales objetivos mejorar la capacidad y accesibilidad de aprendizaje de la programación, como agilizar la labor de los docentes de este campo.
- Guo *et al.* (2023) aporta un enfoque generativo del NLP un tanto diferente, pero muy lógico si se razona: la generación (y evaluación) de música. Y es que, la notación musical no es más que un lenguaje en concreto, en el que en lugar de describir el habla humana, se describen las diferentes armonías y partituras de cada instrumento.

Todas estas aplicaciones nos muestran, no solo el nivel de progreso de las técnicas de NLP, sino además la amplia capacidad de introducción en campos absolutamente dispares, como se han visto en los ejemplos mostrados: sociología, ciencias de la computación, investigación científica, música, etc.

1.2. Planteamiento del trabajo

Una vez se han entendido estos conceptos, es momento de destacar el enfoque que se pretende aplicar a esta disertación. Y es que, el primer paso en cualquier investigación, independientemente del campo de estudio, es tener claro el estado actual de la cuestión.

La intención, por tanto, de este trabajo es identificar de la mejor manera posible este estado del arte para el campo del qNLP. De cumplirse los objetivos, se podrán desbloquear varias ventajas, entre las cuales se destacan.:

1. Se habrá definido una **clara taxonomía** de los diferentes campos de aplicación del qNLP hasta la fecha de esta disertación.
2. Se podrán **identificar futuros ámbitos** de estudio que no hayan sido ya explorados.
3. Se podrá **profundizar** en los ámbitos y aplicaciones ya identificadas.
4. La metodología escogida permitirá **aumentar el estudio** en fechas posteriores.

En resumen, la intención de esta disertación es obtener un mapa claro de los diferentes ámbitos de aplicación del qNLP hasta la fecha así como otros datos útiles tanto cuantitativos como cualitativos.

1.3. Estructura del trabajo

Para materializar este planteamiento, se propone la siguiente estructura de lo que queda de documento:

- **Capítulo 2 - Contexto y estado de la técnica**, en la que se mostrarán trabajos similares a este y las principales diferencias con las que se pretende aportar al campo.
- **Capítulo 3 - Objetivos**, donde se plantearán tanto los objetivos globales de esta disertación así como los objetivos específicos.
- **Capítulo 4 - Desarrollo del trabajo**, en el que se presentará y explicará en detalle la metodología de trabajo elegida para cumplir los objetivos planteados.
- **Capítulo 5 - Resultados**, donde se mostrarán y explicarán los resultados de la metodología de estudio propuesta en capítulos anteriores.
- **Capítulo 6 - Conclusiones**, en el que se reflejarán algunas ideas que se han podido obtener tanto con los resultados del estudio así como con la propia ejecución de este.
- **Capítulo 7 - Trabajo futuro**, donde se identificarán varias posibilidades de trabajo tanto en el propio campo de qNLP como para la mejora de este estudio.

2. Contexto y estado de la técnica

Como en cualquier otro ámbito, resulta de gran utilidad tener en mente las aportaciones (o al menos algunas) que se han hecho ya a nivel de aportación. Es decir, un estudio sobre qué otros estudios se han publicado ya similares a este, a modo de “metaestudio”. Y es precisamente esta intención la que se pretende llevar a cabo en este Capítulo.

Comenzando por una de las más recientes, encontramos la contribución de Widdows *et al.* (2024). Presenta un estudio (más cercano a un nivel teórico) de cómo se están aplicando las técnicas del lenguaje (como por ejemplo, los *word embeddings*¹) dentro del marco de la computación cuántica.

Además, también proponen una nueva técnica para codificar la información (concretamente, las palabras para los modelos) que no había sido propuesta anteriormente. Y, como última aportación, analizan el reciente impacto de los *Large Language Models* (LLM)².

Otro estudio del estado del arte sobre qNLP y también muy reciente es el de Widdows *et al.* (2024). En este documento se trata de describir diferentes aplicaciones tangibles actuales sobre qNLP así como algunas posibles comparaciones entre ellas para lograr diferentes tareas.

Destacamos el caso de la clasificación de textos por temáticas, pues muestra una aplicación que puede ser bastante cotidiana (entre otras aplicaciones podemos incluir clasificar correos, detección de odio, procesado de textos, etc.). Concretamente, uno de los enfoques que plantea destaca por su simplicidad de comprensión (no de implementación). Este utiliza rotaciones sobre el estado del cúbit (realmente son más de uno, pero usaremos uno solo por simplicidad) en función de las palabras que aparecieran. De este modo, al terminar el procesado de la frase o texto que se está analizando, en función del estado del cúbit, se podría determinar a que categoría pertenece ese texto.

A parte, se muestran otras muchas aplicaciones concretas, aportando datos de los resultados de estas para poder así contextualizar mejor cómo se encuentra en términos de utilidad el campo de qNLP. Y es que, es importante recordar que la teoría y la práctica no siempre avanzan a la vez, y más en campos relacionados con la inteligencia artificial, donde, normalmente por problemas de *hardware*, la propuesta de soluciones y algoritmos

¹Un *word embedding* es una forma de representar palabras como puntos en un espacio, de manera que palabras con significados similares estén cerca una de la otra.

²Algunos ejemplos de estos son *ChatGPT*, *Gemini*, *LLaMA*, *Claude*, etc (entre otros muchos).

suele ir bastante por delante de la aplicación práctica de estos.

Por otro lado, destacamos la aportación de Guarasci *et al.* (2022). En este trabajo, se presenta un análisis, efectivamente, sobre qNLP pero con un enfoque más dirigido tanto a los diferentes enfoques de aplicación (teóricos, inspirados cuánticamente pero ejecutados en *hardware* clásico, o cuánticos puros ejecutados en ordenadores cuánticos) así como los ámbitos en los que se usan dichos enfoques (extracción de información, confección de respuestas a preguntas, análisis de sentimientos, etc). Además, proporciona otra información útil:

1. Un contexto teórico sobre la computación cuántica y uno de los modelos formales más usados en qNLP (el modelo *DisCoCat*).
2. Una descripción de algunos enfoques anteriores a qNLP, principalmente con el origen y evolución de *DisCoCat*.
3. Una reflexión sobre los futuros retos a los que se enfrentará qNLP, en todos los enfoques mostrados.

Finalmente, cabe resaltar la contribución de Wu *et al.* (2021) en la que presentan también un estudio sobre los principales métodos resultantes del cruce entre los dos grandes campos involucrados en los últimos diez años³: computación cuántica y procesamiento del lenguaje natural.

Para ello, clasifica las diferentes aportaciones en función de varios criterios:

1. El tipo de algoritmos que se utilizan en proyectos relacionados con qNLP, que bien pueden ser algoritmos clásicos inspirados por el marco computacional cuántico, o bien algoritmos puramente cuánticos.
2. El objetivo de modelado, es decir, a qué aspecto del lenguaje o estudio de este nos estamos enfocando. En concreto, habla de la representación de palabras en circuitos cuánticos y de la comprensión sintáctica y gramática de frases.
3. Las propias aplicaciones de las contribuciones, es decir, el enfoque más práctico de estas (en el caso de que lo tuvieran). Por ejemplo, extracción de información, respuesta de preguntas, reconocimiento del discurso,

³Teniendo en cuenta la fecha de publicación del estudio de Wu *et al.* (2021), este involucra aquellas aportaciones entre 2011 y 2021.

Por último, presenta una discusión acerca de los posibles retos que se deberán afrontar en un futuro, tanto para la evolución del campo como para un mejor entendimiento y comprensión.

Es momento entonces de destacar el enfoque novedoso que se pretende aportar con esta disertación. Y es que, todos los estudios que se han mencionado a lo largo de este Capítulo muestran el estado del arte de una forma más cercana a la teoría de qNLP.

Sin embargo, también es interesante, de cara a poder contextualizar bien el campo, obtener información, no solo cualitativa como la que se muestran en estos estudios, sino también cuantitativa (aportación que no existe aún y con la que se pretende que esta disertación resulte de utilidad).

Es decir, la novedad que se pretende aportar con este trabajo reside en el enfoque sistemático de la metodología de investigación, que es lo que permitirá obtener dichos resultados cuantitativos y cualitativos, relacionándolos y tratando de encajarlos entre sí. En concreto, y como se explicará en posteriores capítulos, trataremos de llevar a cabo un estudio de mapeo sistemático de la literatura, que nos aporta esa metodología sistemática.

3. Objetivos

Aún por redactar.

3.1. Objetivos generales

3.2. Objetivos específicos

4. Desarrollo del trabajo

Para obtener un estudio del estado del arte sobre el campo de qNLP y por ende, cumplir los objetivos de este trabajo, se ha definido la hoja de ruta que se muestra en la Figura 4.1.

Textualmente, el desarrollo del estudio se divide en tres etapas:

1. **Planificación:** Se comenzará definiendo el tipo de estudio que se va a realizar (teniendo en cuenta que se quiere contextualizar el campo del qNLP). Esto dará como resultado la metodología a seguir en función del tipo de estudio.

Entonces, se podrá adaptar para lograr un proceso de ejecución. Este proceso es un guión de los pasos a seguir más en concreto para procesar y depurar todos los artículos y contribuciones.

Una vez se ha definido como ejecutar el proceso de depuración, es momento de definir los parámetros necesarios para las siguientes etapas.

2. **Ejecución:** Como bien dice el nombre, comenzaremos esta etapa ejecutando el guión del proceso que se obtuvo en la fase anterior.

Esto dará como resultado el conjunto de artículos y contribuciones finales que se consideran lo suficiente relevantes para poder extraer conclusiones sobre el estado de la cuestión. Este conjunto se denomina la lista de estudios primarios.

Con esta lista construida, es se procederá a extraer la información relevante para la siguiente etapa. Esta información será contenida en forma de tabla de conocimiento (por ejemplo, un documento *Excel*).

3. **Resolución:** Es la etapa final del proceso, en la que básicamente, aprovechando la información generada en la fase anterior, se definirá el estado de la cuestión mediante dos contextualizaciones: una cualitativa obtenida gracias a responder las preguntas de investigación y otra cuantitativa que se corresponde con los resultados de un análisis estadístico de aspectos como las fechas de publicación, países más activos en este ámbito, tipo de publicaciones, etc.

En esta sección, se explicarán en detalle las etapas de planificación y ejecución. La etapa restante con la resolución del trabajo será detallada en el Capítulo 5.

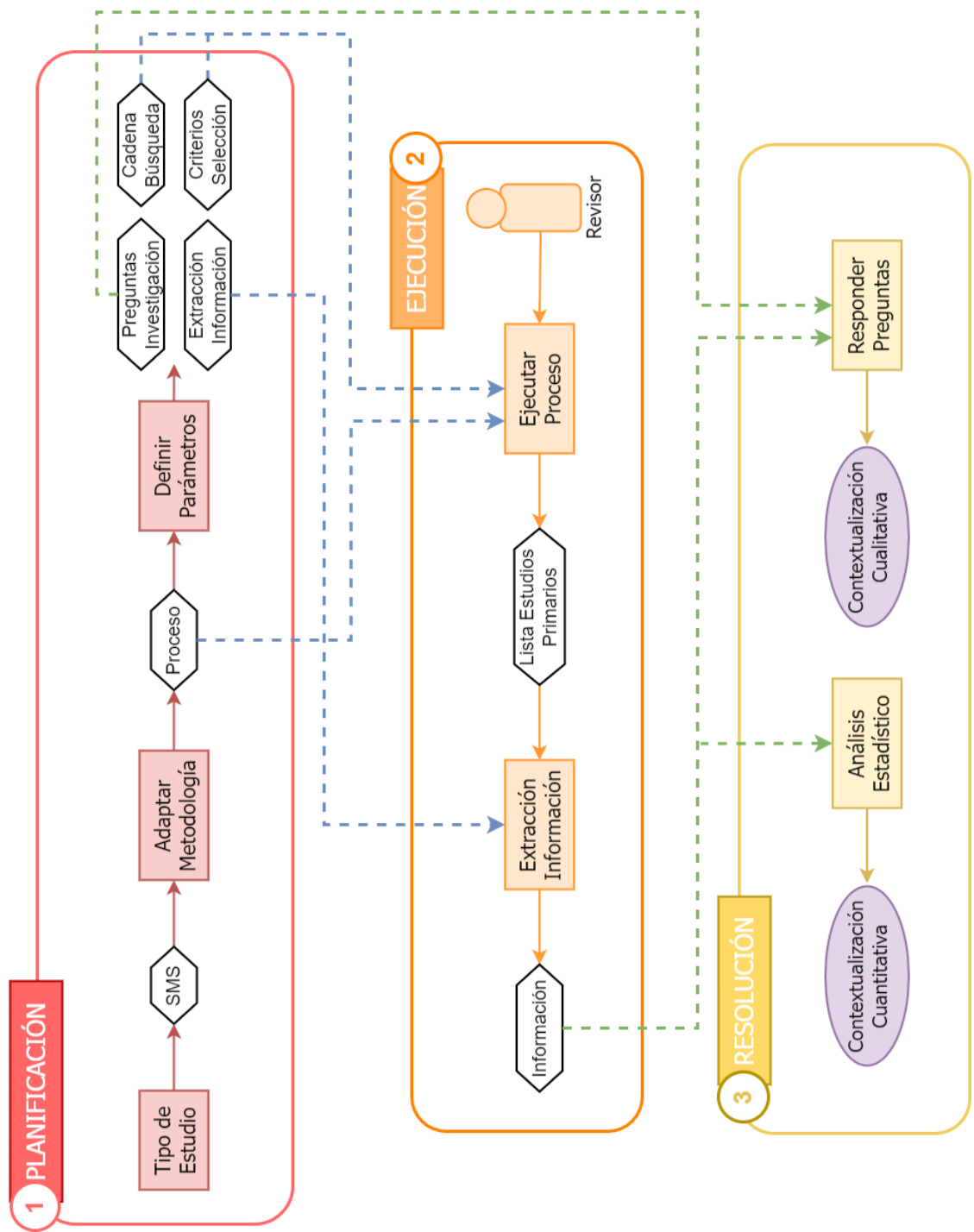


Figura 4.1: Diagrama con el proceso definido para elaborar el estudio del estado del arte sobre el procesamiento cuántico del lenguaje natural.
Fuente: Elaboración Propia.

4.1. Planificación

Como ya se explicó en párrafos anteriores, la planificación es la etapa que permite definir y estructurar el guión así como los parámetros intermedios para poder ejecutar el proceso de forma sistemática.

En esta subsección se definirán los conceptos y parámetros necesarios para tener una planificación correcta en el proceso.

4.1.1. Metodología

Dado que este estudio requiere una búsqueda en proyectos *software* así como algoritmos y técnicas muy ligadas a la computación e informática, se recurrirá a metodologías de estudio del estado del arte propias del campo de la ingeniería informática.

En concreto, se pueden encontrar principalmente dos posibles tipos de estudios de la literatura que podrán ser llevados a cabo para extraer información relevante sobre el estado actual de este campo. Estos son:

- **Revisión sistemática de la literatura:** (SLR, del inglés *systematic literature review*) Busca respuesta a una pregunta o línea de investigación en concreto (Rozanc, 2018). Suele resultar bastante útil en situaciones con una densa literatura para profundizar muy linealmente en un tema.

En este caso, los resultados del estudio serán respuestas a las preguntas de investigación que se plantearon inicialmente (Kitchenham *et al.*, 2010).

- **Estudio de mapeo sistemático:** (SMS, del inglés *Systematic Mapping Study*) Trata de clasificar los trabajos publicados sobre un tema, así como responder a varias preguntas de investigación y extraer otros metadatos (Rozanc, 2018). Además, las preguntas que se plantean son de un cierto nivel de abstracción. Por ejemplo, detectar cuales son los principales sub-temas que se tratan sobre el tema general (Kitchenham *et al.*, 2010).

Sin embargo, en este tipo de estudio, el resultado es un conjunto de artículos relacionados con la temática, categorizados y contados para obtener resultados numéricos, de acuerdo con Kitchenham *et al.* (2010).

Una de las principales ventajas de los estudios SMS es que, dado que se obtendrá un conjunto clasificado de artículos como resultado del estudio, es más fácil identificar

huecos o posibles ramas de estudio (Petersen *et al.*, 2015).

En el Apéndice A.1 se muestra en la Tabla A.1 un estudio comparativo más exhaustivo entre ambas tipologías.

Cabe destacar que, pese a que puedan entenderse como dos tipos de trabajo diferenciados, en ocasiones, pueden presentar secciones/partes en común (Kitchenham *et al.*, 2010).

Más en detalle, se empleará una aplicación de la metodología propuesta por Petersen *et al.* (2008). Se puede clasificar como una mezcla entre SLR y SMS, donde destaca más el papel de SMS. Esto es, se tratarán de definir preguntas de investigación más bien generales (dado el reciente surgimiento del campo de estudio) pero, para obtener como resultado, no sólo el conjunto de estudios y un análisis estadístico, sino que también las respuestas procedentes a las preguntas de investigación para contextualizar el campo.

Los parámetros necesarios que tendrán que definirse para lograr una correcta aplicación de la tipología SMS son los siguientes:

- **Preguntas de investigación**, a las que se tratarán de responder con el SMS.
- **Cadena de búsqueda** que poder lanzar contra diferentes bases de datos académicas y repositorios y obtener el espacio de búsqueda¹.
- **Criterios de inclusión y exclusión** para poder filtrar los estudios obtenidos como espacio de búsqueda y obtener la lista de estudios primarios.
- **Extracción de información** de la lista de estudios primarios estableciendo que datos serán extraídos de cada uno de ellos.

A continuación, se irán definiendo en las siguientes subsecciones cada una de estos parámetros y como se han aplicado a este trabajo.

4.1.2. Guión del proceso

Para guiar la etapa de ejecución, es necesario confeccionar un buen guión del proceso a seguir. De este modo, el objetivo será poder depurar de manera precisa, pero optimizando el tiempo y el esfuerzo, así como el espacio de búsqueda, y restringir el foco solamente

¹Se define como espacio de búsqueda al conjunto inicial de todos los artículos y contribuciones encontradas simplemente lanzando la cadena contra los motores de búsqueda, sin ningún tipo de filtro

con los artículos que realmente sean interesantes de cara a asentar el estado del arte en el campo de qNLP.

Se partirá del espacio de búsqueda obtenido una vez lanzada la cadena (se definirá en la Subsección 4.1.4) sobre los motores seleccionados (también listados y justificados en la Subsección 4.1.4).

Entonces, sobre el espacio de búsqueda se aplicará un **primer filtrado** en el que aquellas contribuciones que cumplan a simple vista al menos alguno de los criterios de exclusión serán destacadas. Esto dejará como resultado un subconjunto del espacio de búsqueda que solo contiene a aquellos estudios potencialmente útiles.

Este filtrado dará lugar a dos resultados:

1. **Subconjunto de la lista de estudios primarios** con aquellos que sí que cumplen también todos los criterios de inclusión de forma evidente.
2. **Conjunto potencial** con aquellas contribuciones en las que el cumplimiento de algún criterio de inclusión no es tan evidente. Esta lista será filtrada una vez procesado el espacio de búsqueda al completo.

Finalmente, otro **segundo filtrado** será aplicado, en el que se examinarán de forma más exhaustiva las contribuciones del conjunto potencial, descartando si fuera necesario aquellas contribuciones que, tras un segundo análisis más profundo y enfocado al criterio de inclusión conflictivo, finalmente no lo cumplan. Este análisis en detenimiento se realizará buscando en el cuerpo del documento completo.

Este segundo filtro dará como resultado el subconjunto restante de la lista de estudios primarios.

Todo este proceso se resume de forma gráfica en la Figura 4.2, donde se pueden apreciar todos los aspectos explicados en esta subsección: partiendo de la cadena de búsqueda sobre los distintos repositorios, pasando por la construcción del espacio de búsqueda, sobre el cual se aplicarán dos filtros que permitirán confeccionar finalmente la lista de estudios primarios.

4.1.3. Preguntas de investigación

El primer paso de la metodología a aplicar es definir unas preguntas de investigación que permitirán extraer información sobre el estado del arte.

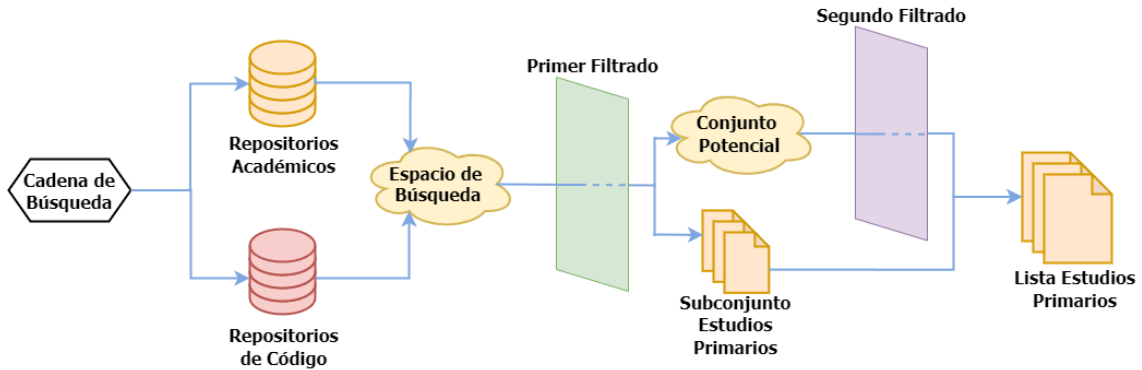


Figura 4.2: Diagrama con el guión definido para la fase de ejecución de la SMS. Fuente: Elaboración Propia.

Como se mencionó anteriormente, dado que estamos ante estudio del tipo SMS, las preguntas de investigación han de ser ciertamente genéricas. De este modo, la contextualización con un punto de vista amplio el estado de la cuestión será posible.

Las preguntas confeccionadas son las siguientes:

- **PI_1 : ¿Cuáles son las principales formas de implementar algoritmos y/o técnicas de qNLP?**

Acogiéndonos a la generalidad que nos ofrece la tipología SMS, es razonable tratar de encontrar cuales son los principales algoritmos que se usan en las diferentes aplicaciones.

La respuesta a esta pregunta podría orientar a futuros proyectos de qNLP para decidir que algoritmo emplear acorde al estado del arte.

- **PI_2 : ¿Cuáles son las principales intenciones de aplicar algoritmos y/o técnicas de qNLP?**

Dentro de esta pregunta se pueden acoger diferentes usos de qNLP: desde proyectos para soluciones prácticas o incluso para facilitar el desarrollo de nuevos proyectos de qNLP hasta propuestas teóricas de nuevos algoritmos que se enmarquen dentro de este campo.

Esta pregunta permitirá conocer en qué se está usando actualmente el qNLP y poder, sabiendo el potencial de esta tecnología en dichos ámbitos, buscar nuevas aplicaciones dentro de esta. O incluso tratar de abrir nuevos caminos en otros campos no explorados.

- PI_3 : ¿Existe variedad en los lenguajes procesados hasta ahora con qNLP?, ¿si es así, cuáles son los principales idiomas?

Otro aspecto que puede resultar de especial interés es una jerarquía de los principales lenguajes ya explorados con qNLP.

Y es que, dada la similitud o diferencia entre los diferentes idiomas (por ejemplo, el inglés y el español pueden presentar ciertas similitudes, sin embargo, respecto al alemán, son bastante diferentes) esta pregunta puede ser útil para decidir el idioma con el que trabajar en futuros proyectos. Del mismo modo, puede usarse esta información de forma contraria, tratando de identificar otros idiomas no explorados con los que aplicar las técnicas de qNLP.

Además de las preguntas de investigación presentadas, otros datos que serán extraídos del conjunto final de artículos serán los años de publicación, los países (si fuera posible), el tipo de contribución (artículos, conferencias, libros, etc). Así, será posible formular otras conclusiones ciertamente más estadísticas, además de las ya planteadas con las preguntas anteriores.

4.1.4. Cadena de búsqueda

Para obtener un espacio de búsqueda lo suficientemente representativo, es necesario construir, no solo una cadena de búsqueda, sino las que sean necesarias. Para esto, dos decisiones deberán ser abordadas: (i) elegir qué bases de datos o repositorios consultar, y (ii) qué cadenas de búsqueda se ejecutarán contra cada una.

4.1.4.1. Repositorios a consultar

Para construir el espacio de búsqueda, se tendrán que lanzar peticiones a diferentes motores de búsqueda científicos. Los repositorios científicos que serán consultados serán:

1. *Scopus*²: Por ser uno de los repositorios más conocidos mundialmente.
2. *arXiv*³: Pese a contener algunos artículos que no han pasado por el proceso de revisión por pares, es imprescindible incluirlo dado que es una de los repositorios donde más se publican estudios relativos con computación cuántica en general.

²<https://www.scopus.com/search/form.uri?display=basic#basic>

³<https://arxiv.org/>

3. *Quantum*⁴: Es una revista de artículos científicos relacionados con la cuántica en general que han pasado el control por pares. Se incluye precisamente por la temática de esta.
4. *Springer Link*⁵: Contiene una gran cantidad de temáticas que pueden ser útiles y es también bastante conocido.
5. *Google Scholar*⁶: Aunque este solo incluya citas, se añade a la lista para poder encontrar alguna cita de algún repositorio no tan conocido pero que pueda resultar interesante para nuestro estudio.

Pero, debido al aspecto práctico que presenta el campo del qNLP, resulta interesante consultar también algunos repositorios de código. Y es que, quizá existan algunos proyectos que no tienen una publicación a sus espaldas, pero que pueden ser realmente relevantes para el campo.

El repositorio de código que se consultará será *GitHub*⁷, por ser el repositorio de código más famoso a escala mundial.

Por supuesto, existen otros repositorios como *Bitbucket*⁸, que es también bastante conocido. Los motivos por los que no se recurrirá a la búsqueda en este repositorio son los siguientes:

1. *BitBucket* es principalmente usado por empresas que ya utilizan plataformas que ofrece Atlassian, como por ejemplo, Jira, Confluence o Trello⁹.

Precisamente por esto, se considera difícil que existan algunos proyectos de qNLP en este tipo de repositorios, dado que no es precisamente una labor de investigación con intención corporativa.

2. *GitHub* es mundialmente conocido, como ya se dijo anteriormente, lo que ofrece una fiel representación de las diferentes comunidades de desarrolladores, inclusive la comunidad de desarrollo de software cuántico, como es este caso.

De este modo, podremos obtener una muestra suficientemente significativa de la contribución de dicha comunidad al qNLP recurriendo a *GitHub*.

⁴<https://quantum-journal.org/>

⁵<https://link.springer.com/>

⁶<https://scholar.google.es/>

⁷<https://github.com/>

⁸<https://bitbucket.org/>

⁹<https://www.atlassian.com/es>

La principal peculiaridad es que la lista final de estudios primarios es probable que no solo contenga artículos académicos, sino que pueda hacer referencia proyectos *software*.

4.1.4.2. Cadenas de búsqueda

El siguiente paso es confeccionar diferentes cadenas en función de las posibilidades que ofrecen los motores de búsqueda de los repositorios anteriores. Estas son:

- **Búsqueda simple:** Para aquellos motores de búsqueda que no permitan opciones de búsqueda avanzadas se han definido tres versiones:
 - *Versión larga 1* - CB_1 : “quantum natural language processing” (es imprescindible realizar una búsqueda con la definición completa del término).
 - *Versión larga 2* - CB_2 : “quantum language processing” (porque a veces el adjetivo natural se suele omitir).
 - *Versión intermedia* - CB_3 : “quantum nlp” (dado que el término NLP está ampliamente extendido).
 - *Versión corta* - CB_4 : “qnlp” (para aquellos casos la computación cuántica está más arraigada).

Esto implica tres búsquedas diferentes pero interseccionando los resultados (es decir, aquellos que aparezcan en una de ellas, pero en ninguna más).

- **Búsqueda avanzada:** Para aquellos motores de búsqueda que sí que permitan opciones avanzadas. Básicamente, se incluirán las tres cadenas anteriores en una sola gracias a dichas opciones. La cadena en cuestión será:

- *Versión avanzada:* “(quantum AND (natural AND (language processing OR processing) OR nlp)) OR qnlp” ¹⁰.

De este modo, los repositorios con búsqueda simple son *Quantum*, *Springer Link* y *GitHub*. Por otro lado, los que permiten la búsqueda avanzada son *Scopus*, *arXiv*, y *Google Scholar*.

¹⁰Nótese que en las abreviaciones “nlp” o “qnlp” no se incluye como opcional el término *processing* dado que no suele ser abreviado en la literatura sin la letra “p” correspondiente al término.

Sin embargo, cabe resaltar el caso concreto de *arXiv*, que, pese a que soporte una búsqueda avanzada, debido a la implementación del motor de búsqueda, ofrece unos mejores resultados lanzando cadenas simples de búsqueda que construyendo una cadena avanzada de búsqueda. Es por esto que *arXiv* será utilizando mediante la estrategia de búsqueda simple.

Lo mismo ocurre también con *Google Scholar*, de forma que, en aras de obtener mejores resultados y encontrar la mayoría de artículos útiles, será también usado con las técnicas de búsqueda simple definidas.

A modo de resumen, se muestra en la Tabla 4.1 que tipo de búsqueda se aplicará sobre cada motor de búsqueda científico.

<i>Tipo</i>	<i>Motor</i>
Simple	Quantum, Springer Link, GitHub, arXiv, Google Scholar
Avanzada	Scopus

Tabla 4.1: Asignación de tipología de búsqueda definida para esta disertación para cada motor de búsqueda científico sobre los que se realizará la búsqueda. Fuente: Elaboración Propia.

4.1.5. Criterios de inclusión y exclusión

Los criterios de inclusión y exclusión son los que permiten filtrar en el espacio de búsqueda para encontrar la lista final de estudios primarios. Como es lógico, un buen diseño de los criterios de selección permitirán un correcto y productivo estudio del estado del arte en el campo.

En este sentido, estos criterios deben abarcar todas las decisiones posibles a la hora de evaluar un estudio. Para este caso, los criterios de selección a aplicar serán los siguientes:

- **Criterios de inclusión:** Una contribución que pase a la lista final de estudios primarios ha de cumplir todos estos criterios. Estos son:
 - CI_1 : Se mostrará una intención explícita de contribuir al campo del qNLP. Esta intención debe reflejarse claramente en el abstracto o incluso en el título.
 - CI_2 : En caso de las contribuciones en forma de código, deberá mostrarse esta intención en el título del proyecto, resumen o cualquier documentación que este pueda tener.

- CI_3 : Aquellos datados entre 1980 hasta la fecha de la búsqueda (2024). La fecha de inicio es la que se corresponde con el año en el que la computación cuántica dio sus primeros pasos, gracias, entre otros, a Benioff (1980) y Feynman (1982).
- CI_4 : En caso de que la contribución sea en forma de código, debe mostrar los resultados de la implementación, o en su defecto, una explicación de como funciona éste.
- CI_5 : Los documentos de la contribución deben estar escritos en inglés preferentemente o, en su defecto, español, dado que son los lenguajes que el revisor, en este caso el alumno, conoce.

- **Criterios de exclusión:** Toda aquella contribución que cumpla al menos uno de los criterios de exclusión (como bien dice el nombre) será descartada de cara a la lista final.

En este caso, los criterios de exclusión definidos serán los antónimos de los criterios de inclusión. De este modo, el criterio CE_1 se cumplirán cuando no se muestre una intención explícita de contribuir al campo, y así sucesivamente.

Otro aspecto que se tendrá en cuenta será la accesibilidad de los artículos potencialmente útiles. El motivo de no definirlo como un criterio de inclusión *per se* es que solo podría aparecer para resolver dudas, lo que hace que no sea aplicable a todos los artículos del espacio de búsqueda.

En concreto, este aspecto de la accesibilidad definirá que, para aquellos casos en los que la contribución (sobre todo aquellos académicos, dado que los de código son accesibles siempre que sean *open source*) no sea accesible con las licencias disponibles gracias a la universidad, en caso de que haya que analizarlo para resolver una duda, dado que no se podrá acceder al cuerpo completo del documento, este será descartado por imposibilidad de resolución.

Finalmente, destacar que en otros tipos de estudios sistemáticos, sí que pueden aparecer otros criterios de exclusión que no se definan como la negación de los de inclusión, solo que este estudio lo requiere así.

4.1.6. Extracción de información

Una vez se haya construido la lista final de estudios primarios, hay que extraer toda la información relevante para obtener las respuestas que buscamos así como para poder

hacer los análisis procedentes (explicados en la Sección 4.1.3).

Pero, antes de poder extraer la información, es necesario definir concretamente qué información queremos extraer de cada estudio primario. Esta tarea nos permitirá agilizar el procesamiento de cada estudio primario. Los datos a extraer serán:

1. **Técnica o algoritmo utilizado:** Para poder responder a la pregunta PI_1 se tendrán que extraer, de todos los estudios primarios, las técnicas que utiliza o proponen para la solución.
2. **Categoría:** Para poder responder a la pregunta PI_2 , intentando comprender y averiguar qué finalidad tienen las aportaciones analizadas (estudio teórico de un partícular para comenzar su camino en la qNLP, formalización de alguna librería de programación aceptada en alguna revista científica, investigación teórica en forma de artículo que contiene algún matiz de innovación para el campo, etc.).
3. **Idioma de procesamiento:** Para responder a la pregunta PI_3 , referente al idioma que se está dando como entrada a la aplicación de qNLP.
4. **Tipo de contribución:** Para el análisis de metadatos, pudiendo clasificarlos como proyectos *software*, artículos, conferencias, libros, etc.
5. **País:** Para el análisis de metadatos, siendo interesante conocer las regiones geográficas que lideran actualmente la investigación y avance sobre qNLP.
6. **Fecha:** Aunque nos podamos hacer una idea de que estamos ante un reciente crecimiento, para el análisis de metadatos las fechas siempre serán una característica relevante en cualquier estudio.

Con estos datos se estima que será suficiente para poder extraer las conclusiones previstas.

4.2. Ejecución

Una vez detallados todos los aspectos necesarios para poder ejecutar el proceso de la manera más óptima posible, en efecto, se llevó a cabo el proceso de la Figura 4.2.

En otros estudios del estado del arte sistematizados, independientemente del tipo de estudio del que se trate (de los que se explicaron también en apartados posteriores de esta

disertación) se suele disponer de varios ejecutores para digerir todo el espacio de búsqueda obtenido tras lanzar las cadenas de búsqueda.

Sin embargo, en esta ocasión, al tratarse de un trabajo de fin de estudios, fue el propio alumno a cargo de este documento el encargado de digerir todo el espacio de búsqueda.

Con respecto al guión definido en la Figura 4.2, caben destacar las siguientes consideraciones y decisiones tomadas en tiempo de ejecución del plan esperado:

1. En casos de bases de datos con **poca representatividad**, es decir, una baja aparición de artículos relacionados con qNLP, se analizaran en total aproximadamente unos 1000 estudios.

El porqué de esta decisión es que, al haber poca representatividad, los trabajos realmente útiles aparecían en las primeras páginas de la búsqueda. Después de estas primeras páginas, prácticamente no había ninguna aparición de nada relevante (para el campo del qNLP). Entonces, se decidió llegar a 1000 aproximadamente para poder parar el proceso tratando de no perder prácticamente representatividad.

2. Por otro lado, para las bases de datos con **mucha representatividad**, dado que hay más apariciones, lograr una muestra relativamente variada y fiel de las aportaciones al campo es más rápido.

Precisamente por esto, se establecieron límites pero de forma más ajustada a cada caso. En concreto, esta medida se tuvo que tomar exclusivamente para el motor de búsqueda *Google Scholar*. Hablando de números, se decidió establecer un punto de gatillo para cada búsqueda simple: si en total una búsqueda devuelve un resultado de tamaño total mayor o igual a 100, se reducirá el tamaño a 50, de lo contrario, se analizarán todos los obtenidos.

Con esto, se podrá obtener una muestra suficientemente representativa para dicha fuente de artículos sin perder demasiada fidelidad respecto a la muestra y optimizando tiempos.

Por otro lado, algunas conclusiones cualitativas simplemente del proceso de ejecución (no de los resultados obtenidos) es la baja aparición de artículos y trabajos dudosos, obteniendo un total de cinco dudas. Todas ellas concentradas con las diferentes búsquedas en *arXiv* (tres para la cadena simple larga 1 y dos para la intermedia).

Resultados Crudos

	Cadena de Búsqueda			
	CB_1	CB_2	CB_3	CB_4
<i>Quantum</i>	15	3	2	17
<i>Springer Link</i>	10000	10000	1577	45
<i>GitHub</i>	27	37	39	66
<i>Scopus</i>	1393			
<i>arXiv</i>	18	3	6	19
<i>Google Scholar</i>	373	30	134	469

Tabla 4.2: Tabla con el recuento de los resultados obtenidos del primer paso del estudio, obtenidos directamente tras realizar las búsquedas. Fuente: Elaboración Propia.

4.2.1. Conteo por etapas

En la Tabla 4.2 se muestra un conteo con la cantidad de contribuciones obtenidas de cada fuente tras realizar cada una de las cadenas de búsqueda. Destacar cómo, efectivamente, se ejecutó la búsqueda simple sobre todos los motores menos con Scopus, que se muestra en la tabla como una sola columna.

Entrando más en profundo con los resultados, hay que mencionar que, pese a las altas cifras que puedan haber en algunas búsquedas, tal y como se mencionó anteriormente, se ajustará el número de contribuciones a procesar y además, hay un alto componente repetitivo, ya no solo entre diferentes fuentes, sino que entre diferentes búsquedas simples dentro de la misma fuente.

A continuación, en la Tabla 4.3 se puede ver el conteo correspondiente a un segundo paso de selección, siguiendo los criterios que se mencionaron al principio de esta Sección de Ejecución.

Aunque, de nuevo, en este recuento, sigue habiendo artículos repetidos tanto en diferentes búsquedas dentro y fuera de la misma fuente.

Notese como, en total, se procesó la cantidad de 2803 contribuciones, dado que, aunque haya repetidas en este conteo, también se procesaron aunque fueran descartados por haber sido ya procesados.

Continuando con el siguiente paso de la ejecución, encontramos en la Tabla 4.4 el conteo tras aplicar el primer filtrado que se describió en el guión del proceso.

Resultados Procesados

	Cadena de Búsqueda				
	CB_1	CB_2	CB_3	CB_4	<i>Total</i>
<i>Quantum</i>	15	3	2	17	37
<i>Springer Link</i>	300	300	300	45	945
<i>GitHub</i>	27	37	39	66	169
<i>Scopus</i>	600				600
<i>arXiv</i>	18	3	6	19	46
<i>Google Scholar</i>	373	30	134	469	1006
Total	2803				

Tabla 4.3: Tabla con el recuento de los resultados obtenidos que sí que serán procesados y utilizados como entrada al proceso de evaluación. Fuente: Elaboración Propia.

Además, cabe destacar que, en esta ocasión, no se contabilizan las contribuciones repetidas dentro la misma fuente, aunque si que pueda haber repetición entre diferentes fuentes.

Otro comentario procedente es sobre los artículos potencialmente útiles, es decir, las dudas que surgieron. Si no se incluyen en una tabla de conteo a parte, es precisamente porque no se han generado prácticamente dudas si lo comparamos con los 2803 estudios procesados.

En concreto las los artículos etiquetados como potencialmente útiles y que, posteriormente, resultaron ser realmente interesantes y pasaron a la PSL vienen de la fuente *arXiv* y son:

- Tres provenientes de la cadena de búsqueda CB_1 .
- Dos provenientes de la cadena de búsqueda CB_2

Precisamente por haber sido calificadas estas pocas dudas como estudios primarios, la Tabla 4.5 es exactamente igual que la Tabla 4.4, aunque, se indica de manera independiente para evitar confusiones y representar mejor el proceso basado en dos filtrados.

Ahora sí, con los resultados obtenidos tras el segundo filtro, donde todos los estudios que han llegado hasta aquí ya son estudios primarios, y por ende, ya se ha compuesto la PSL.

<i>Resultados tras Primer Filtrado</i>					
	Cadena de Búsqueda				
	CB_1	CB_2	CB_3	CB_4	<i>Total</i>
<i>Quantum</i>	0	0	0	0	0
<i>Springer Link</i>	17	0	9	1	27
<i>GitHub</i>	16	1	5	26	48
<i>Scopus</i>	63				63
<i>arXiv</i>	18	2	5	3	28
<i>Google Scholar</i>	49	4	14	8	75
Total	241				

Tabla 4.4: Tabla con el recuento de los resultados obtenidos que sí que pasaron el primer filtro y formaron parte de la PSL. Fuente: Elaboración Propia.

Es decir, se ha conseguido extraer una muestra significativa y lo más representativa posible sobre el campo del qNLP. Esta PSL cuenta con un total de 241 contribuciones, lo que significa un 8,59 % del espacio de búsqueda procesado que se mostró en la Tabla 4.4.

A modo de observación llegados a este punto de la disertación, los estudios de qNLP no son para nada comunes, ni fáciles de identificar, tal y como se refleja en el porcentaje anterior, que incluye además una alta componente de repetición, lo que hará que esta proporción de los estudios útiles en el campo del qNLP sea incluso menor.

Pero, como ya se ha visto en estas tablas, aún hay un componente alto de repetición, no dentro del mismo motor de búsqueda pero con diferentes cadenas puesto que ya se mostró en la Tabla 4.4.

Para paliar esto, y poder limpiar la PSL, eliminando aquellos estudios redundantes se ha dedicado otro tiempo adicional en filtrar de nuevo esta lista y purificarla.

La PSL final limpia y, ahora sí, terminada al completo, contiene un total de 160 contribuciones significativas, lo que deja una proporción de 5,70 % sobre el espacio de búsqueda obtenido.

Esto refleja aún más como este campo de estudio está en una etapa temprana de su desarrollo, aunque, como se verá en el Capítulo 5, está en crecimiento significativo.

Otra observación considerable, aunque se detallará más en profundidad en el siguiente Capítulo con los resultados, es que un 30 % de la PSL se corresponde con contribuciones

Resultados tras Segundo Filtrado

	Cadena de Búsqueda				
	CB_1	CB_2	CB_3	CB_4	<i>Total</i>
<i>Quantum</i>	0	0	0	0	0
<i>Springer Link</i>	17	0	9	1	27
<i>GitHub</i>	16	1	5	26	48
<i>Scopus</i>	63				63
<i>arXiv</i>	18	2	5	3	28
<i>Google Scholar</i>	49	4	14	8	75
Total	241				

Tabla 4.5: Tabla con el recuento de los resultados obtenidos que pasaron el primer y segundo filtro, lo que define la propia PSL al completo, por base de datos y por cadena de búsqueda. Fuente: Elaboración Propia.

Lista de Estudios Primarios (PSL)

<i>Académicos</i>	<i>Código</i>	<i>Total</i>
112	48	160

Tabla 4.6: Tabla con el recuento de los resultados obtenidos por cada tipo de contribución, distinguiendo entre proyectos de *software* y aportaciones académicas. Fuente: Elaboración Propia.

en forma de proyectos *software*.

Esta cifra no es en absoluto insignificante y refleja la importancia que tiene la comunidad de desarrolladores (ya no solo de *software* cuántico) actualmente en las nuevas tecnologías y ramas de la ciencia.

Una vez detallado el proceso y la ejecución del proceso definido al principio de este Capítulo, la siguiente etapa es tratar de responder a las preguntas de investigación y calcular esas otras estadísticas tanto cualitativas como cuantitativas que nos permitan definir las conclusiones del estudio.

La lista completa enumerada por tipo, tal y como se muestra en la Tabla 4.6, se puede encontrar en la Sección 7 de la Bibliografía.

5. Resultados

Con todo esto, se han cubierto los primeros puntos del proceso definido para la SMS en la Figura 4.1. Por ende, el siguiente paso a tomar es la resolución, donde se tratará de contestar a las preguntas de investigación y a aportar cierta información adicional considerada en la primera parte de planificación.

Precisamente este tercer y último paso del proceso del estudio del estado del arte es el que se cubrirá en este Capítulo, aportando, explicando y analizando la información tanto cualitativa como cuantitativa extraída gracias a los pasos anteriores, definida en la Subsección 4.1.6.

Como nota previa, cabe destacar que se ha programado un *script*¹ de *Python* para poder llevar a cabo la extracción de la información concretada de forma sistemática, automática y estandarizada. Para más detalle consultar el Anexo A.2.

Este *script* junto con los datos generados por este que se han utilizado para contestar a las preguntas y la elaboración de resultados, y los diagramas usados en esta memoria se encuentran en un repositorio de *GitHub*. Más información sobre este en el Anexo A.3.

5.1. Preguntas de investigación

De acuerdo a los resultados establecidos a ofrecer en la fase de planificación, el primer conjunto de información que se debe aportar son las respuestas a las preguntas de investigación definidas y explicadas en la Subsección 4.1.3.

A la hora de explicar los resultados de cada pregunta de investigación, la estructura a seguir es la siguiente:

1. Una **introducción** con algún comentario o aclaración previa a tener en cuenta.
2. Los **resultados** en forma de gráfica o tablas, a tres niveles: *general* con todos los estudios primarios (independientemente de su categoría), proyectos de código y, contribuciones académicas.
3. Un **análisis individual** de los resultados a cada nivel de los tres mencionados.
4. Un **análisis global** con los resultados de los tres niveles de forma conjunta.

¹Un *script* es un archivo de código que contiene una serie de instrucciones o comandos que se ejecutan de manera secuencial para automatizar tareas específicas dentro de un entorno de programación.

5.1.1. PI_1 : Conceptos/algoritmos empleados

¿Cuáles son las principales formas de implementar algoritmos y/o técnicas de qNLP?

Los conceptos que se detallan a continuación han sido definidos concretamente siguiendo la siguiente premisa: Encontrar el concepto teórico más relevante y fundamental de cada contribución que hace que esta logre cumplir con lo que se promete.

Es por esto que, algunos conceptos puedan estar anidados o repetidos, como es el caso de los circuitos cuánticos y el algoritmo de Grover. Este tiene como objetivo encontrar uno o más elementos en una base de datos no estructurada con una complejidad temporal menor de la que ofrecen los algoritmos clásicos.

De acuerdo a lo que se ha explicado en la Subsección 1.1.1, donde se introducen los conceptos teóricos mínimos para comprender superficialmente la QC, el algoritmo de Grover es un circuito cuántico también, pero puesto que los fundamentos de dichas contribuciones se basan específicamente en el algoritmo de Grover en lugar de los circuitos cuánticos genéricos, el concepto fundamental es este algoritmo.

Los resultados se muestran en la Figura 5.1 en forma de gráfico circular y en cada uno se muestran los 10 conceptos más recurrentes empleados en los tres niveles definidos en la introducción de esta Sección 5.1.

Cómo se puede observar en la Figura 5.1, comenzando por el **análisis general** de todas las contribuciones de la PSL (independientemente si son proyectos *software* o contribuciones académicas), el concepto más extendido es *DisCoCat*. *DisCoCat* es un marco teórico para la representación de frases en función de su sintaxis en forma de diagrama y que bebe de la teoría de las gramáticas de pregrupos (Le Du , 2021).

En segundo lugar, vemos que también es una de las más extendidas a **nivel de código**, junto al *BobcatParser* (CQCL , s. f.), que se trata de un procesador de lenguaje que tiene el objetivo de convertir los diagramas del marco *DisCoCat* a circuito cuántico de forma automática. Este es una clase de la librería *lambeq* (Kartsaklis *et al.* , 2021), una de las más utilizadas en la comunidad.

Por otro lado vemos como a **nivel académico** *DisCoCat* recoge de nuevo la gran mayoría de contribuciones en este ámbito.

Si se examinan estos conjuntos de conceptos en su totalidad (incluyendo los tres grupos), también son dignas de mencionar otros fundamentos teóricos. Entre los cuales se destacan:

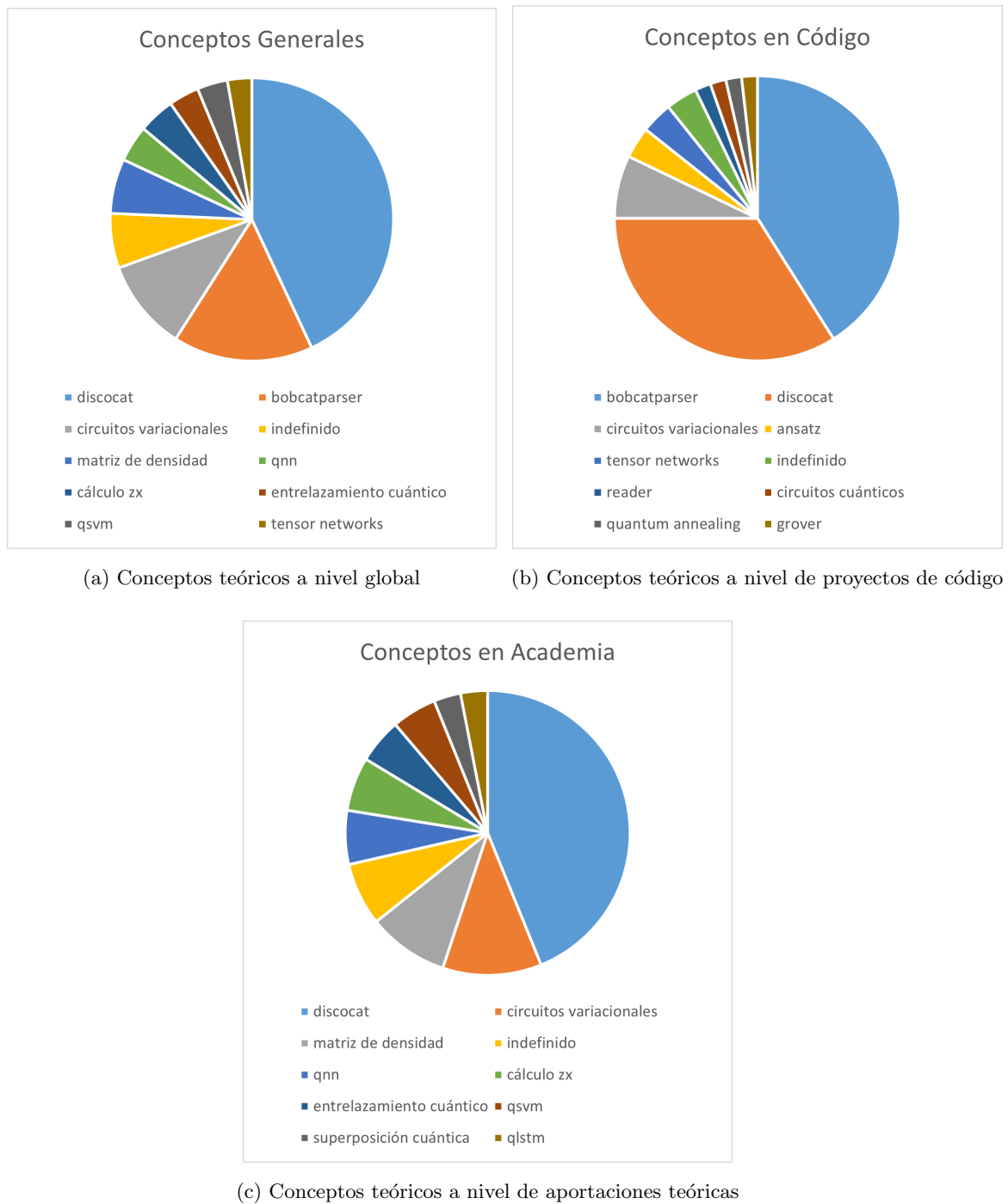


Figura 5.1: Gráficos circulares de conceptos fundamentales empleados para contribuir al campo de la qNLP. Contando con todas las contribuciones, sólo las de código, y solo las académicas, respectivamente de izquierda arriba hasta abajo. Fuente: Elaboración Propia.

1. Los **circuitos variacionales**, considerados como el homólogo cuántico de los algoritmos clásicos de aprendizaje automático, consisten en circuitos cuánticos para-

metrizados donde el valor de dichos parámetros se calcula mediante algoritmos de optimización clásicos y de forma iterativa (Cerezo *et al.*, 2021).

Resulta interesante la aparición de este tipo de circuitos puesto que suponen un enfoque híbrido para lograr el tratamiento del lenguaje, lo que ya se mencionó en la Subsección 1.1.1.

2. El *quantum annealing*, que también se explicó en la Subsección 1.1.1, y que, pese a que su principal foco de actuación sean los problemas de optimización, este tipo de problemas también tienen cabida dentro de las tareas del procesamiento del lenguaje, aunque no parecieran muy relacionados.

5.1.2. PI_2 : Aplicaciones e intenciones

¿Cuáles son las principales intenciones de aplicar algoritmos y/o técnicas de qNLP?

Para esta segunda pregunta de investigación, lo primero será definir bien las categorías con las cuales se han clasificado los estudios primarios de la PSL y su significado para entender bien los resultados. Estos son:

- **Investigación:** Referencia a todos los proyectos *software* con cierta componente innovadora que se pueden incluir en la punta de lanza de la investigación sobre qNLP.
- **Investigación teórica:** Similar a la categoría anterior pero que quedan reflejadas como publicación académica en lugar de proyectos de código.
- **Formalización librería:** Publicación académica que tiene como principal objetivo presentar formalmente a la comunidad científica/tecnológica la creación de una librería de apoyo al desarrollo de qNLP.
- **Estudio teórico:** Proyectos *software* que sirven como estudio y aprendizaje para el/los autores, principalmente, sobre el uso de ciertas librerías.
- **Estado del arte:** Contribución académica cuya finalidad es realizar un mapa de la situación sobre qNLP en el momento que fue realizado .
- **Explicación:** Contenido académico pero sin intención de indagar en nuevas investigaciones sobre qNLP, sino de explicar (a diferente nivel de detalle) ciertos contenidos de esta.

- **Librería desarrollo:** Proyecto *software* con nuevas herramientas para facilitar el desarrollo de futuras contribuciones de código para explorar la qNLP.

Teniendo estas categorías en mente, la extracción se lleva a cabo con un análisis de la intención principal de cada contribución, pudiendo tener un solo estudio más de una intención. Los resultados de la clasificación se muestran en la Tabla 5.1.2.

Nivel	Intención	Num	Num	Intención	Nivel
Generales	<i>Investigación teórica</i>	101	33	<i>Estudio teórico</i>	Código
	<i>Estudio teórico</i>	33	16	<i>Investigación</i>	
	<i>Investigación</i>	16	1	<i>Librería desarrollo</i>	
	<i>Estado del arte</i>	9	101	<i>Investigación teórica</i>	Academia
	<i>Explicación</i>	3	9	<i>Estado del arte</i>	
	<i>Formalización librería</i>	2	3	<i>Explicación</i>	
	<i>Librería desarrollo</i>	1	2	<i>Formalización librería</i>	

Tabla 5.1: Tabla de intenciones/aplicaciones de cada estudio dentro del campo de la qNLP, dividida en niveles: general, código, académicos. Fuente: Elaboración Propia.

Empezando esta vez con las **contribuciones *software*** destacan los estudios teóricos, seguido por las investigaciones prácticas y, finalmente, una librería de desarrollo. Esto da a entender que, a nivel más ‘‘tangible’’ la comunidad de desarrollo se está formando de las bases teóricas necesarias para este tipo de proyectos, así como las herramientas disponibles. Aunque no hay que despreciar la componente de investigación ni la librería *software* de código abierto.

Respecto a las **contribuciones académicas** se observa que la investigación teórica predomina indudablemente, aunque, teniendo en cuenta la importancia de los estados del arte, también se encuentra un porcentaje relevante de aportaciones.

Si se analiza a **nivel general**, se encuentra más heterogeneidad que en otros resultados, puesto que la propia categorización definida para responder a esta pregunta requiere de la distinción entre proyectos *software* y académicos.

Pese a esto, se aprecia como, entre otras cosas, en proporción son más las investigaciones académicas que las *software*. Además, el desarrollo de librerías, así como su normalización, es poco frecuente pero no inexistente, aunque con tendencia de crecer en los próximos años de acuerdo con la madurez de la qNLP.

Finalmente, esta mencionada heterogeneidad hace que el análisis general sea equivalente al análisis global de ambos tipos simultáneamente (*software* y *académicos*).

5.1.3. PI_3 : Lenguajes investigados

¿Cuáles son las principales formas de implementar algoritmos y/o técnicas de qNLP?

Continuando con los lenguajes más investigados en estudios y aportaciones en el campo de la qNLP, estos han sido identificados a partir de, directamente declaraciones en el propio estudio, o bien, buscando y comprobando los *datasets* que se utilizan en dicha aportación (esto principalmente predomina en las contribuciones de código).

En la Figura 5.2 se muestran varios histogramas de los 10 principales lenguajes utilizados en estudios y contribuciones de qNLP, de nuevo, comenzando con un recuento general de ambos tipos, a continuación de aquellos utilizados en proyectos *software* (en este tipo, pese a que el total de lenguajes no supera los 10, se muestran todos ellos), y finalmente, los más estudiados en aportaciones académicas.

De acuerdo a la Figura 5.2, se observa que, bajo un **análisis general**, el inglés predomina notablemente sobre el resto, seguido por la categoría “indefinido” (que será estudiada al final de esta Subsección) y otros lenguajes, como los formales (aquellos definidos con gramáticas formales que se acogen a la jerarquía de Chomsky), el japonés, el chino, e incluso lenguajes de programación, como *C* y *C++* y lenguajes musicales.

Respecto a las **contribuciones software**, de nuevo el inglés se postula como el idioma principal sin ningún tipo de rival. Seguido, muy de lejos, por otros idiomas como el japonés o el español.

Del lado de las **contribuciones académicas**, el inglés sigue siendo ganador indiscutible, pero los lenguajes formales ganan más representación, siendo este tipo de contribuciones donde más peso ganan (lógico debido a la componente teórica de estas). Aunque vemos que se mantiene esa representatividad, con una gama de lenguajes internacionales, lenguajes de programación, lenguaje musical, etc.

Finalmente, a nivel global, se destacan las siguientes observaciones:

1. La categoría “**indefinido**” hace referencia a todos aquellos estudios primarios de los que no se pudo identificar el lenguaje principal con la información disponible. En total, a nivel general, supone un 17,1 % de todos los idiomas analizados en la PSL.

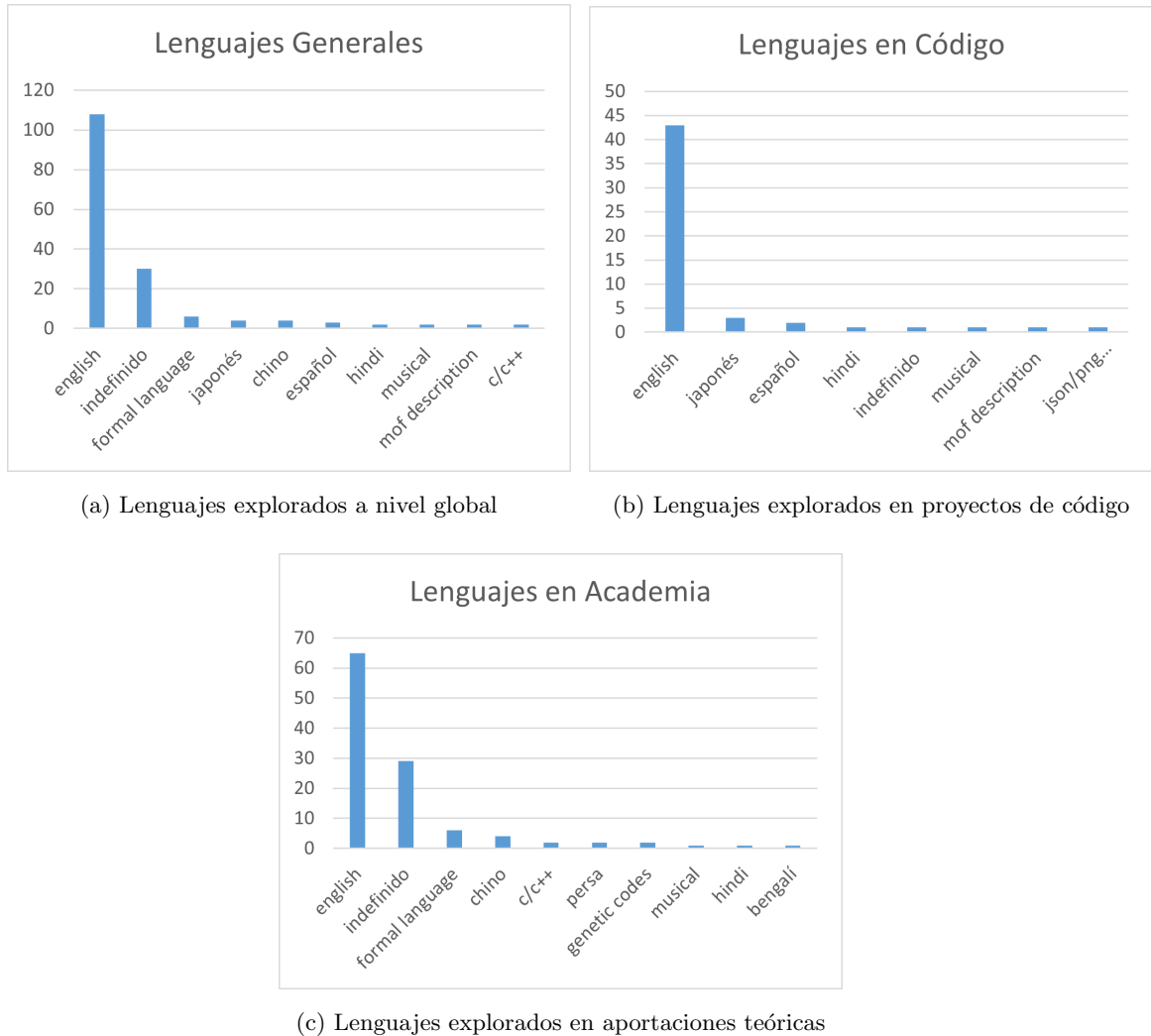


Figura 5.2: Histograma de los 10 lenguajes más procesados para contribuir al campo de la qNLP, a nivel global, de código y académico. Fuente: Elaboración Propia.

El origen de este hueco viene principalmente de los estudios académicos, donde esta indefinición supone un 24 % de todos los idiomas estudiados en esta categoría, mientras que en los de tipo *software* solo se encuentra uno.

Cabe destacar que esto puede suponer un problema en aras de asegurar un rigor y formalidad en las aportaciones. Es fundamental que se justifique (o por lo menos se mencione) el lenguaje del/los *dataset/s* que se están utilizando.

2. El lenguaje “**musical**”, aunque sin mucha representación, puede resultar, cuanto menos, sorprendente. De hecho, ya existen algoritmos cuánticos para la generación de música mediante qNLP, como por ejemplo el proyecto *software* lokesMCI (2023).

3. Las categorías de “**MOF description**” y “**Genetic codes**” hacen referencia a lenguajes definidos de manera sintética para representar marcos metal-orgánicos (*Metal-Organic Frameworks* en inglés) y códigos genéticos (concretamente basados en aminoácidos y las secuencias de ADN).

Esto da muestra de que la qNLP no solo se emplea para lenguajes puros, como pueden ser el inglés o el español (como también se pudo observar en párrafos anteriores) sino que también se emplea en otras ramas de la ciencia como es la biología.

5.2. Datos adicionales

Además de las preguntas de investigación explicadas en la Sección anterior, también se propone en este trabajo aportar información adicional que permita dibujar un contexto actual que complemente a las respuestas de las preguntas de investigación.

5.2.1. Tipos de contribución

Dando comienzo a esta Sección con los tipos de contribuciones que predominan en el campo de qNLP, el resultado del conteo de los estudios primarios se muestra en la Tabla 5.2.1 a nivel general (independientemente del tipo de estudio primario en cuestión).

Tipo	Num
Artículo	76
Proyecto software	44
arXiv	21
TFM	7
Tesis doctoral	5
TFG	3
Capítulo de libro	2
Indefinido	1
Trabajo académico	1

Tabla 5.2: Tabla de tipos de contribuciones a nivel general en qNLP. Fuente: Elaboración Propia.

Cabe mencionar que la tipología de Artículo incluye tanto publicaciones en revistas

académicas como actas de conferencias, pero no incluye *preprints*, que se recogen en la categoría de *arXiv* por ser la única fuente de estos en la PSL.

Si se analizan los números mostrados en dicha tabla, vemos como, a nivel general, los artículos son la tipología de publicaciones más recurridas en el campo de la qNLP, lo que muestra que, pese a la considerable aportación de la comunidad de desarrollo de *software*, las contribuciones académicas son fundamentales (aunque esto también se podía entrever en las preguntas de investigación).

También se puede observar que la comunidad está activa académicamente, puesto que hay 21 *preprints* localizados, así como siete TFMs (Trabajos de Fin de Master), cinco tesis doctorales y tres TFGs (Trabajos de Fin de Grado). Estos tres últimos pueden resultar más sorprendentes, puesto que a ese nivel académico, tratar temas como la qNLP requiere un conjunto de conocimientos poco común en los grados universitarios.

Por otro lado, y con una aportación menos significativa cuantitativamente pero relevante cualitativamente, hay dos capítulos de libros donde ya se incluye esta temática, lo que puede interpretarse como los primeros pasos de consolidación de estos contenidos. Además, esto podría dar pie a que, en los años siguientes, se encuentren aportaciones en forma de libros integramente sobre qNLP.

Finalmente, se puede apreciar una aportación sin clasificación, debido a que no fue posible encajarla en ninguna de las categorías, y un trabajo académico, es decir, la memoria de alguna entrega de alguna actividad incluida en el plan de formación de cierta asignatura.

5.2.2. Países

Como segunda pieza de información adicional por extraer de la PSL se decidió analizar los países donde se generan los estudios primarios en cuestión. Dichos resultados se almacenan en la Figura 5.3 en forma de gráfico de barras horizontales. Además, se muestran en los tres niveles de análisis mencionados en la Sección anterior: general, proyectos *software* y contribuciones académicas.

En cómputo general, vemos que China lidera el *ranking* con 27 contribuciones, seguido de cerca por Estados Unidos y Reino Unido, con 25 y 21 respectivamente. De hecho, este *top* tiene bastante en común con el de los países con mayores inversiones en computación cuántica, donde, en 2022, China se reportó una inversión de 15.3 billones de dólares; Estados Unidos, 1.9 billones; y la Unión Europea 7.2 billones, de acuerdo a Weber (2024).

Aunque la clasificación no sea exactamente la misma, encontramos similitudes, teniendo

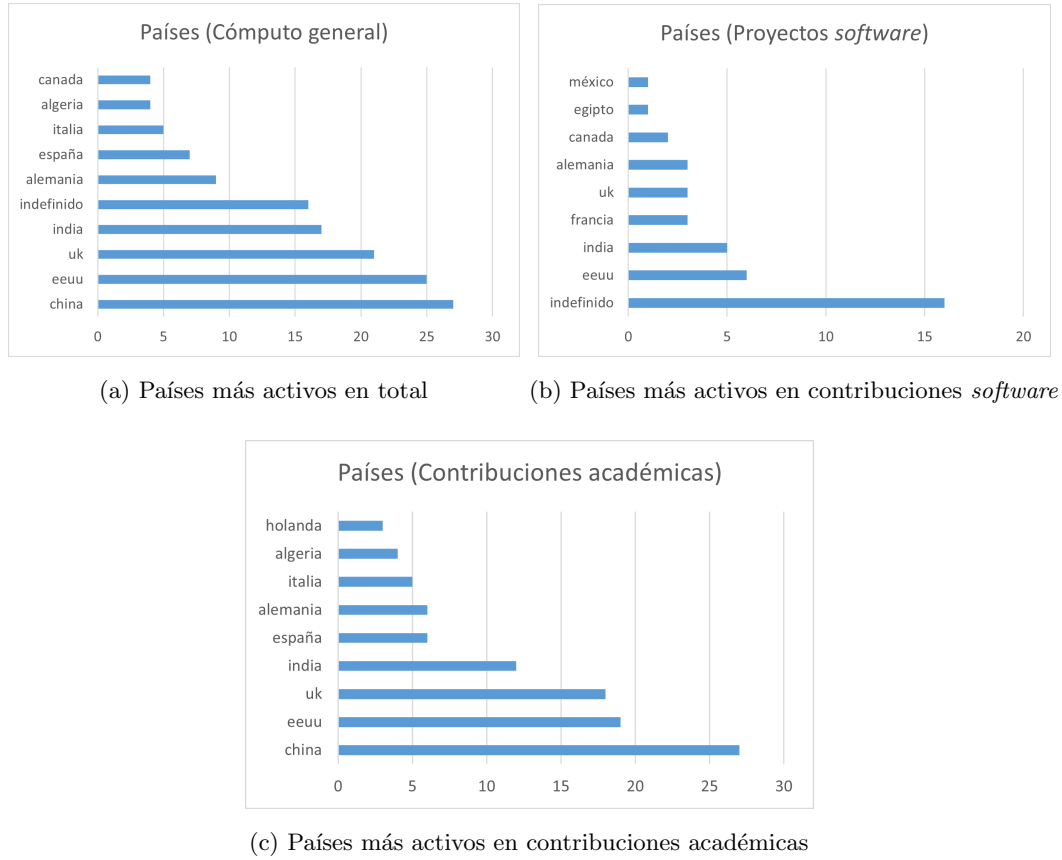


Figura 5.3: Gráfico de barras con un *ranking* de los 10 países más activos respecto a contribuciones al campo de qNLP. Fuente: Elaboración Propia.

en cuenta que también hay países de la Unión Europea en esta, como son España, Alemania e Italia, junto con otros, como son Canadá y Algeria.

Si se centra el foco en las contribuciones *software* se puede encontrar de nuevo una incertidumbre considerable, siendo la categoría más recurrente en este tipo de aportaciones. Esto, igual que se expresó en la Sección anterior, se considera un problema en aras del rigor y transparencia de una correcta y eficiente investigación en este campo.

Finalmente, a nivel de contribuciones académicas, China tiene la completa hegemonía, coincidiendo el *top* con el del cómputo general, pero donde se elimina la incertidumbre (atribuida a las aportaciones de código) y entra en juego otro país europeo, Holanda, lo que iguala más la situación con el *ranking* de inversión en computación cuántica mencionado anteriormente.

5.2.3. Fechas

Por último, las fechas de publicaciones de los estudios primarios de la PSL también se recopilaban para su posterior estudio, en concreto, con una granularidad mensual. Estos valores se muestran en la Figura 5.4 en forma de gráfico de líneas con el mismo desglose en tres niveles: general (color azul), contribuciones *software* (color rojo), y contribuciones académicas (color verde), así como la tendencia del nivel general.

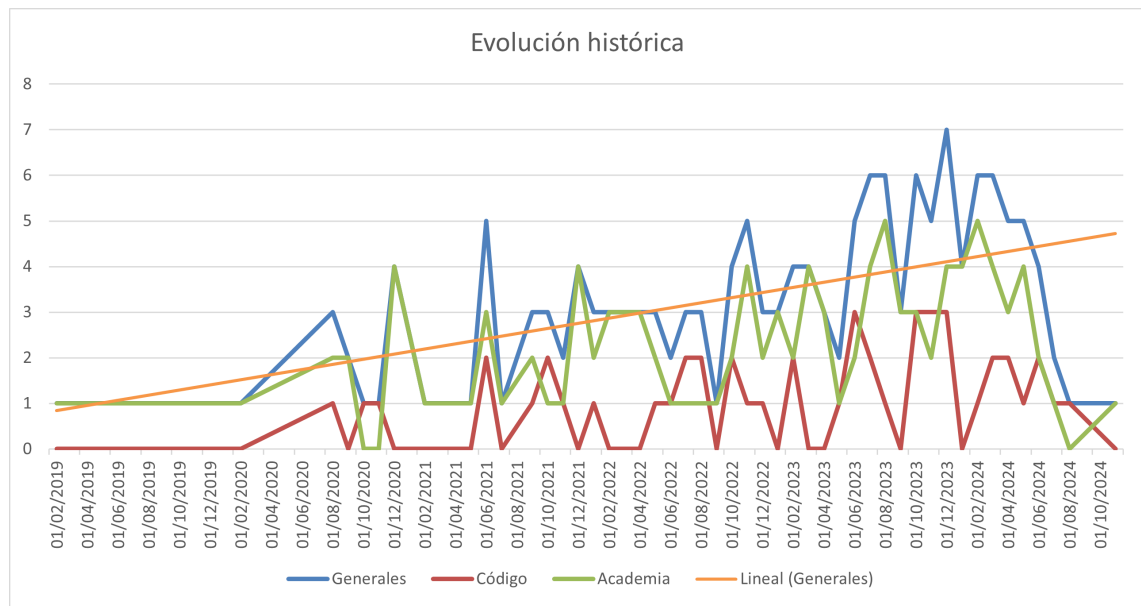


Figura 5.4: Evolución histórica global de contribuciones en el campo de la qNLP. En rojo, la evolución de las contribuciones *software*; en verde, las académicas; en azul la suma de ambas dos a nivel general; y en naranja, la tendencia global de contribución. Fuente: Elaboración Propia.

Para explicar la progresión histórica de estudios primarios, se pueden encontrar tres fases principales:

1. **Primera etapa (02/2019 - 02/2021):** A nivel de código, las contribuciones son eventuales, siendo el máximo en cada mes de una contribución. A nivel teórico, la situación es equivalente pero con un poco más de volumen de contribuciones, alcanzando un mínimo local de 4 publicaciones en enero de 2021.
2. **Segunda etapa (02/2021 - 05/2023):** Se puede observar un crecimiento del volumen de contribuciones en ambos niveles, siendo superior el volumen académico, aunque con ciertos y ligeros valles donde las contribuciones *software* fueron predo-

minantes.

Sin embargo, el mínimo a nivel académico ya no es cero, sino uno, que, pese a no ser muy superior ya da indicios de continuidad en la investigación.

3. **Tercera etapa (05/2023 - Final):** En esta fase cabe destacar el crecimiento en el volumen de contribuciones, lo común empieza a ser entre dos y tres publicaciones de media por mes. Aunque no lo parezca, si se tiene en cuenta el reciente nacimiento de este campo (2019 como se puede observar, poco más de cinco años) es una progresión considerable.

A nivel de código, la situación es algo más irregular, habiendo meses sin contribuciones *software*, pero con unos máximos locales también más altos.

Finalmente, no hay duda de que se tiene una tendencia a la alta, de acuerdo a la recta naranja de la Figura 5.4, lo que asegura un crecimiento del campo en los próximos meses y años. Aunque esto también se podía entrever en resultados anteriores, como por ejemplo es el caso del tipo de contribuciones, lo que reafirma esta predicción.

6. Conclusiones

Aún por redactar.

7. Trabajo futuro

Aún por redactar.

Bibliografía

- Deutsch, D. (1985). Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society A*, 400(1818), 97-117. <https://doi.org/10.1098/rspa.1985.0070>.
- Shor, P. W. (1997). Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Journal on Computing*, 26(5), 1484-1509. <https://doi.org/10.1137/S0097539795293172>.
- Hughes, C., Isaacson, J., Perry, A., Sun, R. F. y Turner, J. (2021). *Quantum Computing for the Quantum Curious*. Springer Nature.
- Kockum, A.F., Nori, F. (2019). Quantum Bits with Josephson Junctions. En: Tafuri, F. (eds) Fundamentals and Frontiers of the Josephson Effect (703-741). *Springer Series in Materials Science*.
- Nielsen, M. A., y Chuang, I. L. (2011). *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press.
- Wilde, M. M. (2019). From Classical to Quantum Shannon Theory. arXiv. <https://arxiv.org/pdf/1106.1445>.
- Cohen-Tannoudji, C., Diu, B., y Laloë, F. (2005). *Quantum mechanics*. Wiley; Hermann.
- Kaye, P., Laflamme, R. y Mosca, M. (2007). *An Introduction to Quantum Computing*. Oxford University Press.
- Zettili, N. (2009). *Quantum mechanics: Concepts and Applications* (2nd Ed.). Wiley.
- Barenco, A., Bennett, C. H., Cleve, R., DiVincenzo, D. P., Margolus, N., Shor, P., Sleator, T., Smolin, J. A. y Weinfurter, H. (1995). Elementary gates for quantum computation. *Physical Review A*, 52, 3457. <https://doi.org/10.1103/PhysRevA.52.3457>.
- Rieffel, E. y Polak, W. (2011). *Quantum Computing: A Gentle Introduction*. The MIT Press.
- Gyongyosi, L. e Imre, S. (2019). A Survey on quantum computing technology. *Computer Science Review*, 31, 51-71. <https://doi.org/10.1016/j.cosrev.2018.11.002>.

- Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N. y Lloyd, S. (2017). *Nature*, 549, 195-202. <https://doi.org/10.1038/nature23474>.
- Melnikov, A., Kordzanganeha, M., Alodjants, A. y Lee, R. K. (2023). Quantum machine learning: from physics to software engineering. *Advances in Physics: X*, 8(1). <https://doi.org/10.1080/23746149.2023.2165452>.
- Bird, S., Klein, E. y Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly.
- Widdows, D., Aboumrad, W., Kim, D., Ray, S. y Mei, J. (2024). *Natural Language, AI, and Quantum Computing in 2024: Research Ingredients and Directions in QNLP*. arXiv. <https://doi.org/10.48550/arXiv.2403.19758>
- Widdows, D., Alexander, A., Zhu, D., Zimmerman, C. y Majumder, A. (2024). *Annals of Mathematics and Artificial Intelligence*. <https://doi.org/10.1007/s10472-024-09940-y>
- Guarasci, R., De Pietro, G. y Esposito, M. (2022). Quantum Natural Language Processing: Challenges and Opportunities. *Applied Sciences*, 12(11), 5651. <https://doi.org/10.3390/app12115651>
- Wu, S., Li, J., Zhang, P. y Zhang, Y. (2021). Natural Language Processing Meets Quantum Physics: A Survey and Categorization. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 3172-3182. <https://doi.org/10.18653/v1/2021.emnlp-main.254>.
- Li, M., Fei, S. M. y Li-Jost, X. (2010). Quantum Entanglement: Separability, Measure, Fidelity of Teleportation, and Distillation. *Advances in Mathematical Physics*, 2010 (301072). <https://doi.org/10.1155/2010/301072>.
- Symons, B. C., Galvin, D., Sahin, E., Alexandrov, V. y Mensa, S. (2023). A practitioner's guide to quantum algorithms for optimisation problems. *Journal of Physics A: Mathematical and Theoretical*, 56(45), 453001. <https://doi.org/10.1088/1751-8121/ad00f0>.
- Shende, V. V., Bullock, S. S. y Markov, I. L. (2005). Synthesis of quantum logic circuits. *ASP-DAC '05: Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, 272-275. <https://doi.org/10.1145/1120725.1120847>.

- Rosmanis, A. (2024). Hybrid Quantum-Classical Search Algorithms. *ACM Transactions on Quantum Computing*, 5(2), 1-18. <https://doi.org/10.1145/3648573>.
- Kwon, S., Tomonaga, A., Bhai, G. L., Devitt, S. J. y Tsai, J. S. (2021). Gate-based superconducting quantum computing. *Journal of Applied Physics*, 129(4), 041102. <https://doi.org/10.1063/5.0029735>.
- Venegas-Andraca, S. E., Cruz-Santos, W., McGeoch, C. y Lanzagorta, M. A cross-disciplinary introduction to quantum annealing-based algorithms. *Contemporary Physics*, 59(2), 174-197. <https://doi.org/10.1080/00107514.2018.1450720>.
- Jones, K.S. (1994). Natural Language Processing: A Historical Review. In: Zampolli, A., Calzolari, N., Palmer, M. (eds), *Current Issues in Computational Linguistics: In Honour of Don Walker. Linguistica Computazionale, vol 9* (3-16). Springer, Dordrecht. https://doi.org/10.1007/978-0-585-35958-8_1.
- Hutchins, J. W. (2004). The Georgetown-IBM experiment demonstrated in January 1954. En *Proceedings of the 6th Conference of the Association for Machine Translation in the Americas: Technical Papers* (102–114). Springer. https://link.springer.com/chapter/10.1007/978-3-540-30194-3_12.
- Minsky, M. (1997). A Framework for Representing Knowledge. En *Mind Design II: Philosophy, Psychology, and Artificial Intelligence*. <https://doi.org/10.7551/mitpress/4626.003.0005>.
- Hirschberg, J. y Manning, C. M. (2015). Advances in natural language processing. *Science*, 349(6245), 261-266. <https://doi.org/10.1126/science.aaa8685>.
- Khurana, D., Koli, A., Khatter, K. y Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82, 3713–3744 (2023). <https://doi.org/10.1007/s11042-022-13428-4>.
- Russel, S. J. y Norvig, P. (2010). *Artificial Intelligence: A Modern Approach (Third Edition)*. Prentice Hall.
- Marsland, S. (2014). *Machine Learning: An Algorithmic Perspective (Second Edition)*. CRC Press.

- Bolaños, X. (2021). *Natural Language Processing and Machine Learning*. Encora.
<https://www.encora.com/insights/natural-language-processing-and-machine-learning>.
- Liddy, E. D. (2001). Natural Language Processing. En *Encyclopedia of Library and Information Science, 2nd Ed.* Marcel Decker, Inc.
- Mote, K. (2002). *Natural Language Processing - A Survey*.
- Chomsky, N (1959). On Certain Formal Properties of Grammars. *Information and Control*, 2, 137-167. [https://doi.org/10.1016/S0019-9958\(59\)90362-6](https://doi.org/10.1016/S0019-9958(59)90362-6).
- Hirst, D (2012). Empirical models of tone, rhythm and intonation for the analysis of speech prosody. *Speech and Language Technology*, 14/15, 23-33. <https://hal.science/hal-01460051>.
- Gupta, V. (2014). A Survey of Natural Language Processing Techniques. *International Journal of Computer Science & Engineering Technology (IJCSET)*, 5(1), 14-16. <https://ijcset.com/docs/IJCSET14-05-01-029.pdf>.
- Yuxiu, Y. (2024). Application of translation technology based on AI in translation teaching. *Systems and Soft Computing*, 6(200072). <https://doi.org/10.1016/j.sasc.2024.200072>.
- Wang, Y. (2024). E-learning application in english writing classroom based on neural machine translation and semantic analysis algorithms. *Entertainment Computing*, 51(100730). <https://doi.org/10.1016/j.entcom.2024.100730>.
- Zhu, S., Li, S. y Xiong, D. (2024). VisTFC: Vision-guided target-sude future context learning for neural machine translation [Formula presented]. *Expert Systems with Applications*, 249(123411). <https://doi.org/10.1016/j.eswa.2024.123411>.
- Medhat, W., Hassan, A. y Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4), 1093-1113. <https://doi.org/10.1016/j.asej.2014.04.011>.
- Breve, B., Caruccio, L., Cirillo, S., Deufemia, V. y Polese, G. (2024). Analyzing the world-wide perception of the Russia-Ukraine conflict through Twitter. *Journal of Big Data*, 11(1). <https://doi.org/10.1186/s40537-024-00921-w>.

- Koonchanok, R., Pan, Y. y Jang, H. (2024). Public attitudes toward chatgpt on twitter: sentiment, topics, and occupations. *Social Network Analysis and Mining*, 14(1). <https://doi.org/10.1007/s13278-024-01260-7>.
- Farhoudinia, B., Ozturkcan, S. y Kasap, N. (2024). Emotions unveiled: detecting COVID-19 fake news on social media. *Humanities and Social Sciences Communications*, 11(1). <https://doi.org/10.1057/s41599-024-03083-5>.
- Neshaei, S. P., Rietsche, R., Su, X. y Wambsganss, T. (2024). Enhancing Peer Review with AI-Powered Suggestion Generation Assistance: Investigating the Design Dynamics. *ACM International Conference Proceeding Series*, 88-102. <https://doi.org/10.1145/3640543.3645169>.
- Freitas, T. C., Neto, A. C., Varanda Pereira, M. J. y Henriques, P. R. (2023). NLP/AI Based Techniques for Programming Exercises Generation. *OpenAccess Series in Informatics*, 112(9). <https://doi.org/10.4230/OASICS.ICPEC.2023.9>.
- Guo, Y., Liu, Y., Zhou, T., Xu, L. y Zhang, Q. (2023). An automatic music generation and evaluation method based on transfer learning. *PLoS ONE*, 18(5), <https://doi.org/10.1371/journal.pone.0283103>.
- Petersen, K., Feldt, R., Mujtaba, S. y Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering. *International Conference on Evaluation & Assessment in Software Engineering*. <https://api.semanticscholar.org/CorpusID:15817433>.
- Rozanc, I. (2018). Effective collection and selection of research articles for a systematic review. *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 1394-1399. <https://api.semanticscholar.org/CorpusID:49646594>.
- Kitchenham, B. A., Budgen, D. y Brereton, P. (2010). The value of mapping studies - A participant-observer case study. *14th International Conference on Evaluation and Assessment in Software Engineering (EASE)*. <https://doi.org/10.14236/ewic/EASE2010.4>.
- Petersen, K., Vakkalanka, S. y Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64, 1-18. <https://doi.org/10.1016/j.infsof.2015.03.007>.

- Benioff, P. (1980). The computer as a physical: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of Statistical Physics*, 22, 563-591. <https://doi.org/10.1007/BF01011339>.
- Feynman, R. P. (1982). Simulating physics with computers. *Int. J. Theory Phys.*, 21, 467-488. <https://doi.org/10.1007/BF02650179>.
- Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., ENdo, S., Fujii, K., McClean, J. R., Mitarai, K., Yuan, X., Cincio, L. and Coles, P. J. (2021). *Variational quantum algorithms*. arXiv. <https://www.arxiv.org/pdf/2012.09265>.
- Weber, V. (2024). The New Quantum Technology Race. *Internationale Politik Quarterly*. [https://ip-quarterly.com/en/new-quantum-technology-race#:~:text=Compared%20globally%2C%20China%20has%20announced,Japan%20\(%241.8%20billion\)%20combined..](https://ip-quarterly.com/en/new-quantum-technology-race#:~:text=Compared%20globally%2C%20China%20has%20announced,Japan%20(%241.8%20billion)%20combined..)

Lista de Estudios Primarios (PSL)

En esta Sección de la bibliografía se define la PSL dividida en función de si son proyectos *software* o contribuciones académicas.

Proyectos *software*

- abiwardani/qnlp-sa-fp: Repository for QNLP sentiment analysis ansatz exploration (Bachelor's Final Project)*. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/abiwardani/qnlp-sa-fp/tree/main>.
- ACE07-Sev/Quantum-Natural-Language-Processing-with-Lambeq [Jupyter Notebook]*. <https://github.com/ACE07-Sev/Quantum-Natural-Language-Processing-with-Lambeq> (Obra original publicada en 2022).
- Nairdanus/SimCircTenNet [Python]*. <https://github.com/nairdanus/SimCircTenNet> (Obra original publicada en 2024).
- Adytiaa/dss_qnlp_talk at master*. (s. f.). Recuperado 15 de junio de 2024, de https://github.com/adytiaa/dss_qnlp_talk/tree/master.

AkimfromParis/quantumNLP_jp [Jupyter Notebook]. https://github.com/AkimfromParis/quantumNLP_jp (Obra original publicada en 2022).

Allenator/qnlp-ansaetze [Jupyter Notebook]. <https://github.com/Allenator/qnlp-ansaetze> (Obra original publicada en 2022).

Henry-dev-atkins/QNLP_MasterWork [Jupyter Notebook]. https://github.com/henry-dev-atkins/QNLP_MasterWork (Obra original publicada en 2023).

Btrainwilson/qnlp: Quantum Natural Language Processing with Quantinuum. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/Btrainwilson/qnlp>.

Dcavar/quantum-ai-nlp [Jupyter Notebook]. <https://github.com/dcavar/quantum-ai-nlp> (Obra original publicada en 2021).

CQCL/lambeq: A high-level Python library for Quantum Natural Language Processing. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/CQCL/lambeq>.

DaveHub5/QNLP_Classification [Jupyter Notebook]. https://github.com/DaveHub5/QNLP_Classification (Obra original publicada en 2022).

DaveHub5/QNLP_Classification_2_3_4_labels: QNLP Classification with 2, 3 and 4 labels. (s. f.). Recuperado 15 de junio de 2024, de https://github.com/DaveHub5/QNLP_Classification_2_3_4_labels/tree/main.

Ebill007/CS833_QNLP [Jupyter Notebook]. https://github.com/ebill007/CS833_QNLP (Obra original publicada en 2023).

Fatemamelg/QNLP-for-sentence-classification [Jupyter Notebook]. <https://github.com/fatemamelg/QNLP-for-sentence-classification> (Obra original publicada en 2023).

FriederikeButt/QEL_Merck_Challenge: QNLP for adverse events detection. (s. f.). Recuperado 15 de junio de 2024, de https://github.com/FriederikeButt/QEL_Merck_Challenge.

Gitaalekh6763/string_diagram_QNLP: string diagram, quantum cir. (s. f.). Recuperado 15 de junio de 2024, de https://github.com/Gitaalekh6763/string_diagram_QNLP/tree/main.

Hkcs1206/QNLP-Sentiment-Analysis-and-Semantic-Interpretation [Jupyter Notebook]. <https://github.com/hkcs1206/QNLP-Sentiment-Analysis-and-Semantic-Interpretation> (Obra original publicada en 2024).

Ketayon/spam_email [Jupyter Notebook]. https://github.com/ketayon/spam_email (Obra original publicada en 2023).

Ketayon/Project_00141 [Jupyter Notebook]. https://github.com/ketayon/Project_00141 (Obra original publicada en 2023).

ICHEC/QNLP [Jupyter Notebook]. Irish Centre for High End Computing. <https://github.com/ICHEC/QNLP> (Obra original publicada en 2020).

Ichrist97/qnlp_finance [Jupyter Notebook]. https://github.com/ichrist97/qnlp_finance (Obra original publicada en 2023).

Jannesgg/quantum-nlp. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/jannesgg/quantum-nlp/tree/master>.

Jaorduz/QNLP. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/jaorduz/QNLP/tree/main>.

Juanrubiop/FslQNLP. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/juanrubiop/FslQNLP/tree/main>.

SuyashNK/Lambeq-library [Jupyter Notebook]. <https://github.com/SuyashNK/Lambeq-library> (Obra original publicada en 2023).

Shinyoung3/MOF-QNLP [Jupyter Notebook]. <https://github.com/shinyoung3/MOF-QNLP> (Obra original publicada en 2024).

Kavacon/qnlp-lit-review-project [Jupyter Notebook]. <https://github.com/kavacon/qnlp-lit-review-project> (Obra original publicada en 2022).

Poig/Womanium_challenge [Jupyter Notebook]. https://github.com/poig/Womanium_challenge (Obra original publicada en 2022).

lokesMCI/QuantumMusician [Jupyter Notebook]. <https://github.com/lokesMCI/QuantumMusician> (Obra original publicada en 2023).

Mullzhang/quantum-nlp [Jupyter Notebook]. <https://github.com/mullzhang/quantum-nlp> (Obra original publicada en 2019).

NEASQC/WP6-QNLP [Jupyter Notebook]. NEASQC. <https://github.com/NEASQC/WP6-QNLP> (Obra original publicada en 2021).

Watermelonich/Sentiment-Model-CLI [Python]. <https://github.com/watermelonich/Sentiment-Model-CLI> (Obra original publicada en 2023).

Inferneo/QNLP—QIntern-22 [Jupyter Notebook]. <https://github.com/Inferneo/QNLP---QIntern-22> (Obra original publicada en 2023).

PaulaGarciaMolina/QNLP-Qiskit-Hackathon [Jupyter Notebook]. <https://github.com/PaulaGarciaMolina/QNLP-Qiskit-Hackathon> (Obra original publicada en 2021).

NatashiaKaurRaina/Quantum-Natural-Language-Processing-Project-Extension [Jupyter Notebook]. <https://github.com/NatashiaKaurRaina/Quantum-Natural-Language-Processing-Project-Extension> (Obra original publicada en 2023).

Helloerikaaa/quweeting [Python]. <https://github.com/helloerikaaa/quweeting> (Obra original publicada en 2021).

Brihat9135/QNLP [Python]. <https://github.com/brihat9135/QNLP> (Obra original publicada en 2022).

Sheldu45/QNLP. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/sheldu45/QNLP/tree/master>.

Rdisipio/qnlp [Jupyter Notebook]. <https://github.com/rdisipio/qnlp> (Obra original publicada en 2020).

Sraj4063/Quantum-Natural-Language-Processing-with-lambeq [Jupyter Notebook]. <https://github.com/Sraj4063/Quantum-Natural-Language-Processing-with-lambeq> (Obra original publicada en 2022).

Toumix/tally [Jupyter Notebook]. <https://github.com/toumix/tally> (Obra original publicada en 2023).

Toumix/thesis [TeX]. <https://github.com/toumix/thesis> (Obra original publicada en 2022).

Truppelli/Quantum_NLP: Grover's search algorithm applied to natural language processing on a quantum processing unit. (s. f.). Recuperado 15 de junio de 2024, de https://github.com/truppelli/Quantum_NLP.

Ua-datalab/QNLP [Jupyter Notebook]. UArizona Data Lab. <https://github.com/ua-datalab/QNLP> (Obra original publicada en 2024).

Peiyong-addwater/Hackathon-QNLP [Jupyter Notebook]. <https://github.com/peiyong-addwater/Hackathon-QNLP> (Obra original publicada en 2022).

Womanium-quantum/Quantum-Natural-Language-Processing-with-lambeq—Quantinuum: Womanium Quantum Hackathon 2022. (s. f.). Recuperado 15 de junio de 2024, de <https://github.com/womanium-quantum/Quantum-Natural-Language-Processing-with-lambeq---Quantinuum>.

Ruixing76/Hackathon-QNLP [Jupyter Notebook]. <https://github.com/ruixing76/Hackathon-QNLP> (Obra original publicada en 2022).

Jakezappin/qnlp [Python]. <https://github.com/jakezappin/qnlp> (Obra original publicada en 2023).

Contribuciones académicas

Le Du, S. (2021). *A gentle introduction to Quantum Natural Language Processing* [Trabajo Fin de Master, Universidad Politécnica de Madrid]. arXiv. <https://arxiv.org/abs/2202.11766>.

O'Riordan, L. J., Doyle, M., Baruffa, F. y Kannan, V. (2021). A hybrid classical-quantum workflow for natural language processing. *Machine Learning*, 2. <https://doi.org/10.1088/2632-2153/abbd2e>.

Zhang, H., Zhao, Q. y Chen, C. (2024). A light-weight quantum self-attention model for classical data classification. *Applied Intelligence*, 54, 3077 - 3091. <https://doi.org/10.1007/s10489-024-05337-w>.

Zhang, W., Gan, G., Gao, H., Zhang, P., Hui, W., Fan, Z. (2023). *A Measurement-Based Quantum-Like Language Model for Text Matching*. In: Tanveer, M., Agarwal, S., Ozawa, S., Ekbal, A., Jatowt, A. (eds) *Neural Information Processing. ICONIP 2022*. Lecture

- Notes in Computer Science, (13625). Springer, Cham. https://doi.org/10.1007/978-3-031-30111-7_4.
- van der Poel, C. (2023). *A Quantum Approach to Language Modeling* [Tesis Doctoral, Universidad de Nueva York]. ProQuest Dissertations & Theses. <https://www.proquest.com/openview/6bf742dc4ac4de5bd73e6ea36b464c52/1?pq-origsite=gscholar&cbl=18750&diss=y>.
- Bar, N. F., Yenilmez, M., Aksu, S. y Karakose, M. (2024). A Quantum Computing based Approach for Sentiment Analysis in Bilateral Conversations. *2024 28th International Conference on Information Technology (IT)*, 1-4. <https://doi.org/10.1109/IT61232.2024.10475718>.
- Yu, Y., Qiu, D. y Yan, R. (2020). A Quantum Entanglement-Based Approach for Computing Sentence Similarity. *IEEE Access*, 8, 174265 - 174278. <https://doi.org/10.1109/ACCESS.2020.3025958>.
- Coyle, B., Lorenz, R., & Grant, E. (2021). *A quantum natural language processing approach to musical intelligence*. arXiv. Recuperado de <https://arxiv.org/abs/2111.06741>.
- Luo, Q., Xiang, T., & Chen, L. (2022). *Node Aware Sparsity through Gaussian Mixture for Unsupervised Learning*. arXiv. <https://arxiv.org/abs/2208.05393>.
- Bausch, J., Subramanian, S. y Piddock, S. (2021). A quantum search decoder for natural language processing. *Quantum Machine Intelligence*, 3(16). <https://doi.org/10.1007/s42484-021-00041-1>.
- Zhu, J., Ma, X., Lin, Z. y De Meo, P. (2023). A quantum-like approach for text generation from knowledge graphs. *CAAI Transactions on Intelligence Technology*, 8, 1455 - 1463. <https://doi.org/10.1049/cit2.12178>.
- Yu, A., Qiu, D. y Yan, R. (2024). A Quantum-Like Tensor Compression Sentence Representation Based on Constraint Functions for Semantics Analysis. *International Journal of Computational Intelligence Systems*, 17(1). <https://doi.org/10.1007/s44196-023-00380-w>.
- Rai, A., Talekar, A., Khan, J. y Ansari, A. (2022). A Review Article on Quantum Natural Language Processing. *International Journal for Research in Applied Science & En-*

- gineering Technology (IJRASET)*, 10(1).<https://doi.org/10.22214/ijraset.2022.40103>.
- Chen, I. C., Singh, H., Anukruti, V. L., Quanz, B. y Yogaraj, K. (2023). *A Survey of Classical And Quantum Sequence Models*. arXiv. <https://doi.org/10.48550/arXiv.2312.10242>.
- Metawei, M. A., Taher, M., ElDeeb, H. y Nassar, S. M. (2023). A topic-aware classifier based on hybrid quantum-classical model. *Neural Computing and Applications*, 35, 18803 - 18812. <https://doi.org/10.1007/s00521-023-08706-7>.
- Li, Q., Wang, B., Zhu, Y., Lioma, C. y Liu, Q. (2023). *Adapting Pre-trained Language Models for Quantum Natural Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.2302.13812>.
- Mansky, M. B., Wörle, F., Stein, J. K., Müller, R. y Linnhoff-Popien, C. (2023). Adapting the DisCoCat-Model for Question Answering in the Chinese Language. *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 591 - 600. <https://doi.org/10.1109/QCE57702.2023.00073>.
- Yu, Y., Qiu, D. y Yan, R. (2021). An efficient framework for sentence similarity inspired by quantum computing. *2021 IEEE International Conference on Big Knowledge (ICBK)*, 157 - 163. <https://doi.org/10.1109/ICKG52313.2021.00030>.
- Ganguly, S., Morapakula, S. N., Bertel, L. G. A. (2022). *Coded Leadership* (Cap. An Introductoin to Quantum Natural Language Processing (QNLP). CRC Press.
- Murauer, J. (2023). *Analyzing word predictions by quantum natural language processing* [Tesis Doctoral, Universidad de Múnich (LMU)]. <https://www.nm.ifi.lmu.de/pub/Diplomarbeiten/mura23/PDF-Version/mura23.pdf>.
- Abbaszade, M., Salari, V., Mousavi, S. S., Zomorodi, M. y Zhou, X. (2021). Application of Quantum Natural Language Processing for Language Translation. *IEEE Access*, 9, 130434 - 130448. <https://doi.org/10.1109/ACCESS.2021.3108768>.
- Yu, W., Yin, L., Zhang, C., Chen, Y. y Liu, A. X. Application of Quantum Recurrent Neural Network in Low-REsource Language Text Classification *Quantum Information*, 5, 1 - 13. <https://doi.org/10.1109/TQE.2024.3373903>.

- Stein, J., Christ, I., Mansky, M. B., Müller, R. y Linnhoff-Popien, C. (2023). *Applying QNLP to sentiment analysis in finance*. arXiv. <https://doi.org/10.48550/arXiv.2307.11788>.
- Atker, M. S., Shahriar, H. y Bhuiya, Z. A. (2023). Automated Vulnerability Detection in Source Code Using Quantum Natural Language Processing. En: Wang, G., Choo, KK. R., Wu, J., Damiani, E. (eds) *Ubiquitous Security. UbiSec 2022. Communications in Computer and Information Science, 1768*. Springer. https://doi.org/10.1007/978-981-99-0272-9_6.
- Toumi, A. (2022). *Category Theory for Quantum Natural Language Processing* [Tesis Doctoral, Universidad de Oxford]. https://www.researchgate.net/publication/366247347_Category_Theory_for_Quantum_Natural_Language_Processing.
- Wang, Z., Ren, M., Tian, X. y Liang, X. (2019). Character-Level Quantum Mechanical Approach for a Neural Language Model. *International Journal of Computational Intelligence Systems*, 12(2), 1613 - 1621. <https://doi.org/10.2991/ijcis.d.191114.001>.
- Hamilton, K., Shekar, M. C., Gounley, J., Bharadwaj, D., Date, P., Coello-Pérez, E. A., Suh, I. S. y Tourassi, G. (2023). Characterizing Quantum Classifier Utility in Natural Language Processing Workflows. *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2, 369 - 370. <https://doi.org/10.1109/QCE57702.2023.10286>.
- Peral-García, D., Cruz-Benito, J., García-Peñalvo, F. J. (2024). Comparing Natural Language Processing and Quantum Natural Processing approaches in text classification tasks. *Expert Systems with Applications*, 254(124427). <https://doi.org/10.1016/j.eswa.2024.124427>.
- Zhao, Q., Hou, C., Xu, R. (2020). Context Based Quantum Language Model with Application to Question Answering. En: Yang, Y., Yu, L., Zhang, L.J. (eds) *Cognitive Computing – ICC 2020. ICC 2020. Lecture Notes in Computer Science, 12408*. Springer. https://doi.org/10.1007/978-3-030-59585-2_3.
- Bruhn, S. (2021). *Density Matrix Methods in Quantum Natural Language Processing* [Trabajo Fin de Master, Universidad de Osnabrück]. PICS - Publications of the Institute of Cognitive Science. <https://osnadocs.ub.uni-osnabrueck.de/bitstream/ds->

202205026790/7/pics-2022-02-Bruhn_Masterthesis_Density-Matrix-Methods-QNLP_2022.pdf.

Wright, M. (2022). *Design and Implementation of a Quantum Kernel for Natural Language Processing* [Tesis Doctoral, Universidad de Queens]. https://www.researchgate.net/publication/360618378_Design_and_Implementation_of_a_Quantum_Kernel_for_Natural_Language_Processing.

Zheng, J., Gao, Q. y Miao, Z. (2023). Design of a Quantum Self-Attention Neural Network on Quantum Circuits. *2023 International Conference on Systems, Man, and Cybernetics (SMC)*, 1058 - 1063. <https://doi.org/10.1109/SMC53992.2023.10393989>.

Peral-García, D., Cruz-Benito, J., García-Peñalvo, F.J. (2023). Development of Algorithms and Methods for the Simulation and Improvement in the Quantum Natural Language Processing Area. En: García-Peñalvo, F.J., García-Holgado, A. (eds) *Proceedings TEEM 2022: Tenth International Conference on Technological Ecosystems for Enhancing Multiculturality. TEEM 2022. Lecture Notes in Educational Technology*. Springer. https://doi.org/10.1007/978-981-99-0942-1_130.

Toumi, A., Yeung, R., Poór, B. y de Felice, G. (2023). *DisCoPy: the Hierarchy of Graphical Languages in Python*. arXiv. <https://doi.org/10.48550/arXiv.2311.10608>.

Srivastava, N., Belekhar, G., Saumya, S. y Babu H, A. (2023). *Enabling Quantum Natural Language Processing for Hindi Language*. arXiv. <https://doi.org/10.48550/arXiv.2312.01221>.

Bouakba, Y., Belhadef, H. (2023). Ensemble Learning Based Quantum Text Classifiers. En: Abelló, A., et al.. *New Trends in Database and Information Systems. ADBIS 2023. Communications in Computer and Information Science, 1850*. Springer. https://doi.org/10.1007/978-3-031-42941-5_35.

Khatri, N. (2022). *Experimental Comparison of Ansätze for Quantum Natural Language Processing* [Tesis Doctoral, Universidad de Oxford]. <https://www.cs.ox.ac.uk/people/aleks.kissinger/theses/khatri-thesis.pdf>.

Pronesti, M. (2023). *Experimental Quantum Natural Language Processing for the Travel Industry* [Trabajo Fin de Master, Universidad Politécnica de Torino]. <https://webthesis.biblio.polito.it/26689/>.

- Belhadef, H., Benchiheb, H., Lebdi, L. (2023). Exploring the Capabilities and Limitations of VQC and QSVC for Sentiment Analysis on Real-World and Synthetic Datasets. En: Abelló, A., *et al.* *New Trends in Database and Information Systems. AD-BIS 2023. Communications in Computer and Information Science, 1850*. Springer. https://doi.org/10.1007/978-3-031-42941-5_36.
- Coecke, B., de Felice, G., Meichanetzidis, K. y Toumi, A. (2020). *Foundations for Near-Term Quantum Natural Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.2012.03755>.
- Frick, R. A., Vogel, I. y Nunes Grieser, I. (2022). Semi-Supervised Ensemble Classification for Detecting Check-Worthy Tweets. *Conference and Labs of the Evaluation Forum 2022, 3180*(39). <https://ceur-ws.org/Vol-3180/paper-39.pdf>.
- Meichanetzidis, K., Toumi, A., de Felice, G. y Coecke, B. (2023). Grammar-aware sentence classification on quantum computers. *Quantum Machine Intelligence, 5*(10). <https://doi.org/10.1007/s42484-023-00097-1>.
- Correia, A. D., Moortgat, M. y Stoof, H. T. C. (2022). *Grover's Algorithm for Question Answering*. arXiv. <https://doi.org/10.48550/arXiv.2106.05299>.
- Coecke, B., Felice, G.d., Meichanetzidis, K., Toumi, A. (2022). How to Make Qubits Speak. En: Miranda, E.R. (eds) *Quantum Computing in the Arts and Humanities*. Springer. https://doi.org/10.1007/978-3-030-95538-0_8.
- Adeshir-Larijani, E. y Nasiri Fatmehsari, M. M. (2024). Hybrid classical-quantum transfer learning for text classification. *Quantum Machine Intelligence, 6*(19). <https://doi.org/10.1007/s42484-024-00147-2>.
- Masum, A. K. M., Maurya, A., Murthy, D. S., Pratibha, Mahmud, N. (2023). *Hybrid Quantum-Classical Machine Learning for Sentiment Analysis*. arXiv. <https://doi.org/10.48550/arXiv.2310.10672>.
- Kang, S. y Kim, J. (2024). *Inverse Design of Metal-Organic Frameworks Using Quantum Natural Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.2405.11783>.
- Kartsakis, D., Fan, I., Yeung, R., Pearson, A., Lorenz, R., Toumi, A., de Felice, G., Meichanetzidis, K., Clark, S. y Coecke, B. (2021). *lambeq: An Efficient High-Level Python Library for Quantum NLP*. arXiv. <https://doi.org/10.48550/arXiv.2110.04236>.

Coecke, B. y Meichanetzidis, K. (2020). *Meaning updating of density matrices*. arXiv. <https://doi.org/10.48550/arXiv.2001.00862>.

Li, A., Hou, Y., Pan, T., Tian, T. y Gao, Y. (2021). Mining Latent Semantic Correlation inspired by Quantum Entanglement. *CSAI '21: Proceedings of the 2021 5th International Conference on Computer Science and Artificial Intelligence*, 323 - 330. <https://doi.org/10.1145/3507548.3507598>.

Gao, Y., Hou, Y., Tian, T., Pan, T. y Li, Z. (2021). Modeling Semantic Inspired by Quantum Composite System. *2021 2nd International Conference on Electronics, Communications and Information Technology (CECIT)*, 755 - 759. <https://doi.org/10.1109/CECIT53797.2021.00137>.

Wu, S., Li, J., Zhang, P. y Zhang, Y. (2021). Natural Language Processing Meets Quantum Physics: A Survey and Categorization. En: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 3172 - 3182. <https://doi.org/10.18653/v1/2021.emnlp-main.254>.

Widdows, D., Alexander, A., Zhu, D., Zimmerman, C. y Majumder, A. (2024). Near-term advances in quantum natural language processing. *Annals of Mathematics and Artificial Intelligence*. <https://doi.org/10.1007/s10472-024-09940-y>.

Macedo, J. B., Lins, I. D. y das Chagas Moura, M. J. (2023). NLP Advances in Risk Analysis Context: Application of Quantum Computing. En: *Proceedings of the 33rd European Safety and Reliability Conference (ESREL 2023)*, 2613 - 2614. https://doi.org/10.3850/978-981-18-8071-1_P235-cd.

Abbaszadeh, M., Mousavi, S. S. y Salari, V. (2021). *Parametrized Quantum Circuits of Synonymous Sentences in Quantum Natural Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.2102.02204>.

Bishwas, A. K., Mani, A. y Palade, V. (2022). Parts of Speech Tagging in NLP - an Investigation on Runtime Optimization with Quantum Formulation and ZX Calculus. *2022 International Conference for Advancement in Technology (ICONAT)*, 1 - 5. <https://doi.org/10.1109/ICONAT53423.2022.9725821>.

Pandey, S., Dadure, P., Nunsaga, M. V. L. y Pakray, P. (2022). Parts of speech tagging to-

- wards classical to quantum computing. *2022 IEEE Silchar Subsection Conference (SILCON)*, 1 - 6. <https://doi.org/10.1109/SILCON55242.2022.10028796>.
- J. Shi, T. Chen, W. Lai, S. Zhang and X. Li (2024). Pretrained Quantum-Inspired Deep Neural Network for Natural Language Processing. *IEEE Transactions on Cybernetics*, 1 - 13. <https://doi.org/10.1109/TCYB.2024.3398692>.
- Ouamane, N.E.H., Belhadef, H. (2024). Proposed Model for QCNN-Based Sentimental Short Sentences Classification. En: Saeed, F., Mohammed, F., Fazea, Y. (eds) *Advances in Intelligent Computing Techniques and Applications. IRICT 2023. Lecture Notes on Data Engineering and Communications Technologies*, 211. Springer. https://doi.org/10.1007/978-3-031-59707-7_19.
- Lorenz, R., Pearson, A., Meichanetzidis, Kartsaklis, D. y Coecke, B. (2023). QNLP in Practice: Running Compositional Models of Meaning on a Quantum Computer. *Journal of Artificial Intelligence Research*, 76. <https://doi.org/10.1613/jair.1.14329>.
- Zhao, Q., Hou, C., Xu, R. (2022). Quantum Attention Based Language Model for Answer Selection. En: Pan, Y., Mao, ZH., Luo, L., Zeng, J., Zhang, LJ. (eds) *Artificial Intelligence and Mobile Services – AIMS 2021. AIMS 2021. Lecture Notes in Computer Science*, 12987. Springer. https://doi.org/10.1007/978-3-030-96033-9_4.
- Correia, A. D., Moortgat, M. y Stoof, H. T. C. (2023). Quantum computations for disambiguation and question answering. *Quantum Information Processing*, 21(126). <https://doi.org/10.1007/s11128-022-03441-9>.
- Omar, A. y El-Hafeez, T. A. (2023). Quantum computing and machine learning for Arabic language sentiment classification in social media. *Scientific Reports*, 13(17305). <https://doi.org/10.1038/s41598-023-44113-7>.
- Zhang, J., He, R., Guo, F. y Liu C. (2024). Quantum Interference Model for Semantic Biases of Glosses in Word Sense Disambiguation. *AAAI Technical Track on Natural Language Processing II*, 38(17). <https://doi.org/10.1609/aaai.v38i17.29927>.
- Chen, Y., Pan, Y. y Dong, D. (2021). Quantum Language Model with Entanglement Embedding for Question Answering. arXiv. *IEEE Transactions on Cybernetics*, 53(6), 3467 - 3478. <https://doi.org/10.1109/TCYB.2021.3131252>.

- Wiebe, N., Bocharov, A., Smolensky, P., Troyer, M. y Svore, K. M. (2019). *Quantum Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.1902.05162>.
- Nahed, A. (2021). *Quantum Machine Learning and Deep Learning Algorithms for Solving the Answer Selection Problem in Natural Language Question Answering Systems* [Tesis Doctoral, Universidad de Bradley]. Bradley University ProQuest Dissertation & Theses. <https://www.proquest.com/openview/69c86733a8ff7d6c6b52dc0d2e87df65/1?pq-origsite=gscholar&cbl=18750&diss=y>.
- Pandey, S., Basisth, N. J., Sachan, T., Kumari, N. y Pakray, P. (2023). Quantum machine learning for natural language processing application. *Physica A: Statistical Mechanics and its Applications*, 627(129123). <https://doi.org/10.1016/j.physa.2023.129123>.
- Hoffmann, T. (2021). *Quantum Models For Word-Sense Disambiguation* [Tesis Doctoral, Universidad Tecnológica de Chalmers]. <https://odr.chalmers.se/server/api/core/bitstreams/72f4d3ec-bb51-4ffe-a903-683c32f51a16/content>.
- Payares, E., Puertas, E. Martinez-santos, J. C. (2023). Quantum N-Gram Language Models for Tweet Classification. *2023 IEEE 5th International Conference on Cognitive Machine Intelligence (CogMI)*, 69 - 74. <https://doi.org/10.1109/CogMI58952.2023.00019>.
- Karamlou, A., Pfaffhauser, M. y Wooton, J. (2022). *Quantum Natural Language Generation on Near-Term Devices*. arXiv. <https://doi.org/10.48550/arXiv.2211.00727>.
- Widdows, D., Aboumradi, W., Kim, D. Ray, S. y Mei, J. (2024). *Quantum Natural Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.2403.19758>.
- Coello-Pérez, E. A., Suh, I. S., Date, P., Gounley, J. P., Shekar, M. C., Hamilton, K. (2023). Quantum natural language processing applications on high-performance computing systems and quantum devices. *Bulletin of the American Physical Society*. <https://meetings.aps.org/Meeting/MAR23/Session/D73.2>.
- Ganguly, S., Morapakula, S. N. y Pozo Coronado, L. M. (2022). Quantum Natural Language Processing Based Sentiment Analysis Using Lambeq Toolkit. *2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T)*, 1 - 6. <https://doi.org/10.1109/ICPC2T53885.2022.9776836>.

- Meichanetzidis, K., Gogioso, S., de Felice, G., Chaippori, N., Toumi, A. y Coecke, B. (2021). *Quantum Natural Language Processing on Near-Term Quantum Computers*. arXiv. <https://doi.org/10.48550/arXiv.2005.0414>.
- Bouakba, Y., Belhadeh, H. (2023). Quantum Natural Language Processing: A New and Promising Way to Solve NLP Problems. En: Salem, M., Merele, J.J., Siarry, P., Bachir Bouiadja, R., Debakla, M., Debbat, F. (eds) *Artificial Intelligence: Theories and Applications. ICAITA 2022. Communications in Computer and Information Science, 1769*. Springer. https://doi.org/10.1007/978-3-031-28540-0_17.
- Neto, J. S., Ardila, L. Q., Nogueira, T. N., Albuquerque, F., Papa, J. P., Guido, R. C. y Fanchini, F. F. (2024). Quantum neural networks successfully calibrate language models. *Quantum Machine Intelligence*, 6, 8. <https://doi.org/10.1007/s42484-024-00139-2>.
- Kartsaklis, D., Fan, I., Yeung, R., Hoffmann, T., Kocijan, V., London, C., Pearson, A., Lorenz, R., Toumi, A., de Felice, G., Meichanetzidis, K., Clark, S., & Coecke, B. (2022). Quantum NLP with lambeq. *Applied Category Theory 2022 (ACT2022)*. https://msp.cis.strath.ac.uk/act2022/papers/ACT2022_paper_7003.pdf.
- Ruskanda, F. Z., Abiwardani, M. R., Al Bari, M. A., Bagaspati, K. A., Mulyawan, R., Syafalni, I. y Larasati, H. T. (2022). Quantum Representation for Sentiment Classification. *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 67 - 78. <https://doi.org/10.1109/QCE53715.2022.00025>.
- Li, G., Zhao, X. y Wang, X. (2023). *Quantum Self-Attention Neural Networks for Text Classification*. arXiv. <https://doi.org/10.48550/arXiv.2205.05625>.
- Han, Y. X., Pan, Y. y Wang, J. T. (2023). Quantum semantic coding for structure extraction and matching applications. *Modern Physics Letters A*, 38(12), 2350069. <https://doi.org/10.1142/S0217732323500694>.
- Zhou, Y. (2021). *Quantum Semantic Textual Similarity for NISQ era* [Trabajo de Fin de Grado, Universidad de Keio]. <https://aqua.sfc.wide.ad.jp/publications/syubthesis.pdf>.
- Aaranya, A. y Widdows, D. (2022). Quantum Text Encoding for Classification Tasks. *2022*

- IEEE/ACM 7th Symposium on Edge Computing (SEC)*, 355 - 361. <https://doi.org/10.1109/SEC54971.2022.00052>.
- Buonaiuto, G., Guarasci, R., Minutolo, A., De Pietro, G. y Esposito, M. (2024). Quantum transfer learning for acceptability judgements. *Quantum Machine Intelligence*, 6(13). <https://doi.org/10.1007/s42484-024-00141-8>.
- Vinay, R. y Nath, B. (2023). Quantum Video Classification Leveraging Textual Video Representations. *2023 4th International Conference on Communication, Computing and Industry 6.0 (C2I6)*, 1 - 6. <https://doi.org/10.1109/C2I659362.2023.10430918>.
- Lloyd, S., Schuld, M., Ijaz, A., Izaac, J. y Killoran, N. (2020). *Quantum embeddings for machine learning*. arXiv. <https://doi.org/10.48550/arXiv.2001.03622>.
- Ruskanda, F. Z., Abiwardani, M. R., Mulyawan, R., Syafalni, I. y Larasti, H. T. L. (2023). Quantum-Enhanced Support Vector Machine for Sentiment Classification. *IEEE Access*, 11, 87520 - 87532. <https://doi.org/10.1109/ACCESS.2023.3304990>.
- Peng, G., Junwei, Z., Yuexian, H., Xiujun, G., Panpan, W. y Yazhou, Z. (2019). Quantum-Inspired DMATT-BiGRU for conversational sentiment analysis. *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI, 2019*(8995180), 1602 - 1606. <https://doi.org/10.1109/ICTAI.2019.00232>.
- Lai, W., Shi, J., Chang, Y. (2023). Quantum-Inspired Fully Complex-Valued Neutral Network for Sentiment Analysis. *Axioms* 2023(12), 308. <https://doi.org/10.3390/axioms12030308>.
- Guo, P. y Wang, P. (2023). QHAN: Quantum-inspired Hierarchical Attention Mechanism Network for Question Answering. *International Journal on Artificial Intelligence Tools*, 32(5), 2360009. <https://doi.org/10.1142/S0218213023600096>.
- Fan, Z., Zhang, J., Zhang, P. Lin, Q., Li, Y., Qian, Y. (2024). Quantum-inspired language models based on unitary transformation. *Information Processing & Management*. <https://doi.org/10.1016/j.ipm.2024.103741>.
- Li, S. y Hou, Y. (2021). Quantum-Inspired Model Based on Convolutional Neural Network for Sentiment Analysis. *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, 347 - 351. <https://doi.org/10.1109/ICAIBD51990.2021.9459058>.

- Hongbin, W. y Minghui, H. (2023). Quantum-like implicit sentiment analysis with sememes knowledge. *Expert Systems with Applications*, 232(120720). <https://doi.org/10.1016/j.eswa.2023.120720>.
- Song, Z., Zhou, X., Xu, J., Ding, X. y Shan, Z. (2024). Recurrent quantum embedding neural network and its application in vulnerability detection. *Scientific Reports*, 14(13642). <https://doi.org/10.1038/s41598-024-63021-y>.
- Vijayakumar, S. y Louis, F. (2023). Revolutionizing Staffing and Recruiting with Contextual Knowledge Graphs and QNLP: An End-to-End Quantum Training Paradigm. *2023 IEEE International Conference on Knowledge Graph (ICKG)*, 45 - 51. <https://doi.org/10.1109/ICKG59574.2023.00011>.
- Walton, R. D. (2023). Self-move and Other-move: Quantum Categorical Foundations of Japanese. *Journal of Interdisciplinary Sciences*, 7(1), 29 - 54. <https://journalofinterdisciplinarysciences.com/wp-content/uploads/2023/04/3-Self-move-and-Other-move-Quantum-Categorical-Foundations-of-Japanese.pdf>.
- Pallapothu, L. K. K., Vulavalapudi, V. M. S., Evuru, P. C. y Medisetty, P., Prakash, K. B. y Swain, G. (2023). Semantic Analysis of Auto-generated Sentences using Quantum Natural Language Processing. *2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, 623 - 628. <https://doi.org/10.1109/IITCEE57236.2023.10091087>.
- Rajashekharaiyah, K. M. M., Chickerur, S., Hegde, G., Bhat, S. L., Sali, S. A. (2023). Sentence Classification Using Quantum Natural Language Processing and Comparison of Optimization Methods. In: Garg, D., Narayana, V.A., Suganthan, P.N., Anguera, J., Koppula, V.K., Gupta, S.K. (eds) *Advanced Computing. IACC 2022. Communications in Computer and Information Science*, 1782. Springer. https://doi.org/10.1007/978-3-031-35644-5_7.
- Harvey, C., Yeung, R. y Meichanetzidis, K. (2023). *Sequence Processing with Quantum Tensor Networks*. arXiv. <https://doi.org/10.48550/arXiv.2308.07865>.
- Ruskanda, F. Z., Abiwardani, M. R., Syafalni, I., Larasati, H. T. y Mulyawan, R. (2023). Simple Sentiment Analysis Ansatz for Sentiment Classification in Quantum Natu-

- ral Language Processing. *IEEE Access*, 11. <https://doi.org/10.1109/ACCESS.2023.3327873>.
- Uotila, V. *SQL2Circuits: Estimating Metrics for SQL Queries with a Quantum Natural Language Processing Method*. arXiv. <https://doi.org/10.48550/arXiv.2306.08529>.
- Díaz-Ortiz, J.I., Villanueva, A., Delgado, F. (2024). Strongly Entangling Neural Network: Quantum-Classical Hybrid Model for Quantum Natural Language Processing. In: Vlachos, D. (eds) *Mathematical Modeling in Physical Sciences. ICMSQUARE 2023. Springer Proceedings in Mathematics & Statistics*, 446. Springer. https://doi.org/10.1007/978-3-031-52965-8_40.
- Katyayan, P., Joshi, N. (2023). Supervised Question Classification on SelQA Dataset Using Variational Quantum Classifiers. In: Gupta, D., Khanna, A., Hassanien, A.E., Anand, S., Jaiswal, A. (eds) *International Conference on Innovative Computing and Communications. Lecture Notes in Networks and Systems*, 492. Springer. https://doi.org/10.1007/978-981-19-3679-1_60.
- Maruyama, Y. (2021). The Categorical Integration of Symbolic and Statistical AI: Quantum NLP and Applications to Cognitive and Machine Bias Problems. In: Abraham, A., Siarry, P., Ma, K., Kaklauskas, A. (eds) *Intelligent Systems Design and Applications. ISDA 2019. Advances in Intelligent Systems and Computing*, 1181. Springer. https://doi.org/10.1007/978-3-030-49342-4_45.
- Di Sipio, R., Huang, J. H., Chen, S. Y. C., Mangini, S. y Worring, M. (2021). *The Dawn of Quantum Natural Language Processing*. arXiv. <https://doi.org/10.48550/arXiv.2110.06510>.
- Abbaszade, M., Zomorodi, M., Salari, V. y Kurian, P. (2023). *Toward Quantum Machine Translation of Syntactically Distinct Languages*. arXiv. <https://doi.org/10.48550/arXiv.2307.16576>.
- Vicente-Nieto, I. (2021). *Towards Machine Translation with Quantum Computers* [Tesis Doctoral, Universidad de Estocolmo]. DiVA. <https://su.diva-portal.org/smash/record.jsf?pid=diva2%3A1592369&dswid=145>.
- Wazni, H., Sardrzadeh, M. *Towards Transparency in Coference Resolution: A Quantum-Inspired Approach*. arXiv. <https://doi.org/10.48550/arXiv.2312.00688>.

- Peral-García, D., Cruz-Benito, J., García-Peñalvo, F.J. (2024). Using Quantum Natural Language Processing for Sentiment Classification and Next-Word Prediction in Sentences Without Fixed Syntactic Structure. In: Lopata, A., Gudonienė, D., Butkienė, R. (eds) *Information and Software Technologies. ICIST 2023. Communications in Computer and Information Science, 1979*. Springer. https://doi.org/10.1007/978-3-031-48981-5_19.
- Chang, D. T. (2023). *Variational Quantum Classifiers for Natural-Language Text*. arXiv. <https://doi.org/10.48550/arXiv.2303.02469>.
- Platonov, A., Bessmertny, I., Koroleva, J., Miroslovskaya, L., Shaker, A. (2021). Vector Representation of Words Using Quantum-Like Probabilities. En: Dolinina, O., *et al. Recent Research in Control Engineering and Decision Making. ICIT 2020. Studies in Systems, Decision and Control, 337*. Springer. https://doi.org/10.1007/978-3-030-65283-8_44.

A. Apéndices

A.1. Diferencias entre SMS y SLR

En la Tabla A.1 se han traducido de Kitchenham *et al.* (2010) las principales diferencias que podemos definir entre una SLR y un SMS.

Paso	SMS	SLR
Preguntas Investigación	Generales	Específicas
Búsqueda	Según el tema	Según las preguntas
Flexibilidad	Menos estricto	Muy estricto
Evaluación de Calidad	No es esencial	Crucial
Resultados	Conjunto de artículos	Respuestas a preguntas

Tabla A.1: Tabla comparativa entre SMS y SLR en función de los distintos pasos a dar y aspectos a tener en cuenta en cada una. Fuente: (Kitchenham *et al.*, 2010).

A.2. *Script de Python*

El *script* automatiza el proceso de extracción, conteo y categorización de características específicas de los datos extraídos de cada una de las contribuciones de la PSL con un enfoque de programación orientada a objetos ¹.

La principal motivación de utilizar la programación orientada a objetos es la escalabilidad semántica de cara a la programación de extracción de nuevos resultados. Para ello, simplemente añadiendo la lógica del cálculo correspondiente como un nuevo método de la clase, manteniendo la relación semántica con el procesador principal (la clase en cuestión).

En primer lugar, *Pandas*² se ha utilizado como librería principal para el manejo de datos tabulares con los que se generan los resultados útiles para la elaboración de resultados.

A continuación se define un método auxiliar **suma_uno** que se encargará de tomar un diccionario y una lista de conceptos e incrementar el conteo de cada concepto dentro del

¹La programación orientada a objetos es un paradigma de programación que organiza el código en objetos, que son instancias de clases que encapsulan datos (atributos) y comportamientos (métodos), facilitando la modularidad, reutilización y mantenimiento del software.

²<https://pandas.pydata.org/>

diccionario. Si no está presente dicho concepto en el diccionario, se añade un nuevo par clave-valor, de lo contrario, simplemente se suma uno.

El núcleo del *script* reside en la clase **Digestor** donde recae toda la lógica programada con el paradigma de la programación orientada a objetos del procesado de los datos crudos (tras evaluar la PSL) hasta la generación de resultados en formato JSON.

Dentro de esta clase, se define el método **preg_inv** que contiene la funcionalidad de cálculo y serialización de los resultados del procesado. Para más detalle sobre el algoritmo programado, en el siguiente Apéndice se incluye el enlace al repositorio que contiene este *script* para poder ver el código directamente.

A.3. Repositorio de *GitHub*

El repositorio en cuestión ha sido desarrollado con el principal objetivo de dar un enfoque organizado y con una estructura centralizada que recoja todos los recursos y herramientas desarrolladas y utilizadas para la realización de este trabajo.

<https://github.com/SergioJF10/qNLP-SMS>

A.3.1. Justificación

La principal motivación para la creación de este repositorio es que permite una organización y estructuración tanto de los recursos para la explicación en esta memoria, herramientas desarrolladas, y resultados obtenidos tras el análisis de la SMS en un mismo contenedor.

Además, da una transparencia y rigor a la hora de documentar y difundir el proyecto, haciendo ver que las herramientas empleadas son correctas y perfectamente utilizables o aplicables a futuros proyectos similares a este.

A.3.2. Contenidos

El repositorio contiene los siguientes directorios y archivos principales:

1. **imgs/**: Este directorio contiene varios diagramas en diferentes formatos (PNG, SVG, Drawio) diseñados y realizados para complementar de forma visual algunas explicaciones y resultados expuestos en este documento.

2. **results/**: Es el directorio más extenso y contiene varias carpetas anidadas con los resultados del proceso de extracción de datos obtenidos de la PSL, utilizados para responder a las preguntas y aportar la información adicional presentada en el Capítulo 5.
3. **datos_sms.xlsx**: Es el archivo tabular con mayor peso del repositorio. Contiene varias hojas donde se recopilan todos los datos tras la etapa de ejecución de la SMS, conteos, tablas y gráficos para responder a las preguntas y dar resultados.
4. **data_extraction.py**: El *script* de *Python* descrito en el Anexo A.2 para la extracción de los resultados numéricos de forma automatizada. Dichos resultados se almacenan en formato JSON en las carpetas del directorio **results/**. Posteriormente, se generaron los documentos tabulares (formato XLSX) mediante una página de generación de Excels a partir de documentos JSON,³ (también almacenados en el mismo directorio).

³<https://www.convertcsv.com/json-to-csv.htm>