



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE MÁSTER

Diseño de un sistema automático de perfilado indirecto de la personalidad en base a datos extraídos de redes sociales

Máster Universitario en Dirección TIC para la Defensa

ALUMNO: Laura Prada Rivero

DIRECTORES: Luis Álvarez Sabucedo
Milagros Fernández Gavilanes

CURSO ACADÉMICO: 2022-2023

Universida_{de}Vigo



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE MÁSTER

Diseño de un sistema automático de perfilado indirecto de la personalidad en base a datos extraídos de redes sociales

Máster Universitario en Dirección TIC para la Defensa
Especialidad de Sistemas y Tecnologías de Telecomunicación

Universida_deVigo

RESUMEN

La personalidad permite inferir comportamientos, ya que personalidad y conducta van de la mano, tal y como lo muestran diversos estudios psicológicos a lo largo de la historia. Estos estudios han modelado la personalidad de los individuos de diferentes formas, realizando agrupaciones de las características o rasgos de la personalidad, tales como los modelos de Eysenck o los Big Five. Conocer la personalidad, y, por tanto, poder inferir la conducta, es de interés para los servicios de información de los diferentes cuerpos de seguridad ya que ello aporta información muy valiosa en las investigaciones sobre sospechosos o captación de fuentes.

El lenguaje es un recurso básico en la comunicación humana y puede ser usado como herramienta para inferir la personalidad de las personas. Las palabras que las personas utilizan muestran sus creencias, emociones, estados de ánimo, relaciones sociales y, por consiguiente, su personalidad, tal como muestran numerosos trabajos en el ámbito de la psicología moderna. Hoy en día las redes sociales suponen unos de los canales habituales de comunicación del grueso de la población, lo cual supone una fuente inmensa de información emitida por los individuos y puede ser tomada de muestra para inferir su personalidad.

Los modelos psicológicos tradicionales se basan en los tests psicométricos habituales y en las entrevistas clínicas para identificar los rasgos de la personalidad de los individuos. Sin embargo, estas formas de estudio directas no son siempre posible. Existen modelos de la personalidad que permiten un estudio indirecto de los rasgos psicológicos que caracterizan a los individuos y lo hacen en base a ciertos indicadores que se extraen observando el comportamiento de los sujetos. Esos indicadores, concretamente en los centrados en los verbales, trasladados a una orientación informática, abren la puerta a poder utilizarlos para el perfilado indirecto de forma automática.

Con este estudio se pretende plantear un diseño de un sistema que, con información descargada en formato texto de las redes sociales de los individuos, permita de forma automática inferir su personalidad en base a unos indicadores extraídos de un modelo de perfilado indirecto.

Para ello, una vez definida unos requisitos básicos que deba cumplir el sistema, se elegirá un modelo de perfilado indirecto de la personalidad, y se extraerán los indicadores del lenguaje necesarios para inferir los rasgos de forma indirecta. Después se hará un recorrido por aquellas tecnologías actuales que proporcionen los medios para realizar un diseño de un sistema que cumpla los requisitos especificados. Se estudiarán tecnologías como el procesamiento del lenguaje natural (PLN), que permite que las máquinas trabajen sobre el lenguaje que utilizan los humanos, permitiendo el estudio y el análisis de los mensajes analizando su semántica profunda, su estructura y su sintaxis; el Machine Learning (ML), que utiliza algoritmos y modelos de aprendizaje que permiten a la máquina aprender de una forma parecida a como lo hace el humano, con la intención de predecir resultados en base a lo aprendido; y diferentes modos de almacenamiento y generación de informes para los usuarios finales del sistema.

El producto de este estudio será un diseño completo de la arquitectura de un sistema de información que permita extraer los rasgos de la personalidad de un individuo a través de textos generados por él, en este caso datos de sus redes sociales. El sistema plantea una apertura en su diseño para otro tipo de preguntas sobre los datos. Este diseño de arquitectura pretende ser el paso previo a una implementación del sistema en fases posteriores, no contenidas en este trabajo.

PALABRAS CLAVE

Personalidad, Perfilado Indirecto, ENCUIST, Redes Sociales, Procesamiento de Lenguaje natural, Machine Learning

AGRADECIMIENTOS

Uno de los aspectos más importantes del arranque de un sistema informático es el *conocimiento experto*. Por mucha técnica que se conozca, si no se entiende perfectamente el contexto del negocio al que se quiere dar servicio, es imposible tener éxito en el diseño y ejecución del sistema.

En este trabajo la orientación en los contenidos psicológicos en los que se basa ha sido fundamental. Para la elección del modelo psicológico adecuado y la extracción de requisitos a partir de este, el conocimiento experto ha sido imprescindible. Ignacio D., psicólogo e ingeniero técnico informático, y Silvia G., psicóloga, han sido los consultores expertos que han dado la orientación necesaria. La relación entre Silvia G. y la doctora Halty, autora del modelo ENCUIST, es continua y fluida, lo que ha permitido también una orientación clara del trabajo en cuestiones sobre ese modelo psicológico en concreto. Tanto a Ignacio como a Silvia les agradezco mucho su tiempo, la sonrisa y la paciencia con la que me han recibido cada vez que les he llamado para una consulta.

También, a la psicóloga Camino U., que me dedicó tiempo de sus vacaciones cuando yo andaba perdida dentro de mi mundo de ingeniera, lleno de ángulos, y me suavizó el camino, diciéndome que todo era posible siempre que se tuvieran las cosas claras.

Por su paciencia infinita y su juicio crítico a Ismael M., experto técnico y gran compañero.

A mis tutores por su orientación y buenos consejos que me han llevado finalmente a buen puerto.

Un agradecimiento especial a mis compañeros de máster, me llevo amigos que espero que sean para siempre.

Y, sobre todo, muchísimas gracias a mis dos peques a los que les he robado mucho tiempo de estar juntos, y me han animado en todo momento con sus grandes abrazos y sus sonrisas.

CONTENIDO

Índice de Figuras	3
Índice de Tablas	4
Abreviaturas	5
1 Introducción	6
1.1 Contexto del trabajo	7
1.2 Motivación	8
1.3 Objetivo	9
1.4 Organización del documento	10
2 Recopilación conocimiento experto	13
2.1 Perfilado indirecto de la personalidad	13
2.1.1 Personalidad y comportamiento.	13
2.1.2 Modelo ENCUIST	17
2.1.3 Indicadores para el perfilado	19
2.1.3.1 Indicadores verbales para los rasgos ENCUIST.	19
2.1.3.2 Indicadores verbales para los rasgos del modelo PEN de Eysenck.	21
2.1.3.3 Otros posibles indicadores no verbales.	22
2.2 Redes Sociales objetivo	23
3 Arquitectura del sistema	26
3.1 Análisis del sistema	26
3.1.1 Definición de requisitos de subsistemas	27
4 Estudio de herramientas tecnológicas relacionadas	34
4.1 Extracción de información de RRSS	34
4.2 Machine learning	36
4.2.1 Modelos de aprendizaje de Machine Learning	37
4.2.2 Etapas del Machine learning	37
4.2.3 Tipos de problemas a resolver con Machine learning	38
4.2.4 Algoritmos más comunes del Machine Learning	39
➤ Conclusión sobre ML	44
4.3 Procesamiento del lenguaje natural	45
4.3.1 Modelos lógicos: Gramáticas	46
4.3.2 Modelos probabilísticos del lenguaje natural: Machine Learning basado en datos	46
4.3.3 Herramientas clásicas de desarrollo de PLN	46

4.3.4	Aprendizaje profundo o Deep Learning	50
➤	Conclusión sobre PNL	53
4.4	Análisis de sentimientos	54
4.5	Almacenamiento	55
4.5.1	Bases de datos relacionales o SQL	55
4.5.2	Bases de datos no relacionales o NoSQL	56
4.5.2.1	Características de las BBDD NoSQL	56
4.5.2.2	Tipos de BBDD NoSQL	56
➤	Conclusión sobre el Almacenamiento	58
5	Diseño de los subsistemas	60
5.1	Recursos del desarrollo sw	60
5.2	Almacenamiento	61
5.2.1	Colecciones en MongoDB	62
5.2.2	Grafos en Neo4j	65
5.3	Subsistema de extracción de perfiles de RRSS	67
5.4	Subsistema de procesamiento de lenguaje natural y análisis de sentimientos	68
5.5	Subsistema de análisis de rasgos ENCUIST.	70
6	Conclusiones y líneas futuras	75
6.1	Conclusiones	75
6.2	Líneas futuras	75
7	Bibliografía	77
	Anexo I: Test ENCUIST	86

ÍNDICE DE FIGURAS

Figura 1-1 Componentes del diseño propuesto y su relación.....	11
Figura 2-1 Top 10 de las principales redes sociales en España según IAB [20].....	24
Figura 3-1 Análisis de subsistemas y flujos de información.....	27
Figura 3-2 Subsistema de PLN y análisis de sentimientos.....	30
Figura 3-3 Carga de datos iniciales	31
Figura 3-4 Subsistema de análisis de rasgos ENCUIST	32
Figura 4-1 Tipos de Machine Learning.....	37
Figura 4-2 Cálculo de cada rasgo	45
Figura 4-3 Red Neuronal Sencilla	51
Figura 4-4 Neurona	52
Figura 5-1 Jupyter y Python	60
Figura 5-2 Jupyter Notebook[35]	61
Figura 5-3 Almacenamiento e interconexión	62
Figura 5-4 Modelo de grafos	66

ÍNDICE DE TABLAS

Tabla 4-1 Algoritmos más importantes del Machine Learning.....	44
Tabla 4-2 Tipos de BBDD, características y ejemplos.	58
Tabla 5-1 Indicadores ENCUIST de personalidad. Sumatorios de palabras.	64
Tabla 5-2 Rasgos ENCUIST de personalidad	65
Tabla 5-3 Creación de BBDD, tablas y referencias en MongoDB.....	65
Tabla 5-4 Metadatos de RRSS, rasgos calculados y algoritmo de ML	66

ABREVIATURAS

API	Application Programming Interface o Interfaz de Programación de Aplicaciones
CSS	Cascading Style Sheets u Hojas de Estilo en Cascada
ENCUIST	Modelo de la personalidad para la realización del perfil indirecto
FB	Facebook
FFCCSE	Fuerzas y cuerpos de seguridad del estado
HTML	HyperText Markup Language o Lenguaje de Marcas de Hipertexto
JSON	JavaScript Object Notation
ML	Machine Learning
PLN	Procesamiento de Lenguaje Natural
REST	REpresentational State Transfer
RRHH	Recursos Humanos
RRN	Redes Neuronales Recurrentes
RRSS	Redes Sociales
URL	Uniform Resource Locator o Localizador Uniforme de Recursos
Xpath	XML Path Language

1 INTRODUCCIÓN

La personalidad es la tendencia estable de una persona a pensar, sentir y actuar de una determinada manera[6]. Los llamados rasgos de personalidad expresan una tendencia a la hora de procesar e interpretar lo que sucede a cada persona, y con ello la forma en la que sienten[5]. Es decir, los rasgos explican las diferentes conductas que tienen las personas ante las mismas situaciones.

Uno de los temas más debatidos por los expertos acerca de la construcción de la personalidad es conocer si influye más la propia naturaleza de cada individuo (su base biológica o temperamento) o las experiencias vividas [1]. Sin embargo, hoy en día parece haber consenso en que genética y aprendizaje, de forma conjunta, forjan la forma de ser de cada persona. Es por ello, que, aunque marcados de nacimiento con un temperamento, este tiene una influencia en la personalidad, pero no la determina de forma absoluta, ya que existe una diferencia entre lo que se es al nacer y lo que se termina siendo. El carácter de una persona será la suma de su aprendizaje y experiencias[5]. Las experiencias emocionales son filtradas a través de la naturaleza biológica o temperamento y forjarán la personalidad y esta se reflejará en el comportamiento.

La clasificación de la personalidad es objeto de estudio desde tiempos de la antigua Grecia donde se evaluaban más aspectos físicos ligados a los humores, hasta nuestros días donde la clasificación se basa en modelos de conducta [1][2][5][6]. No hay un único modelo, hay varios, todos ellos con bases comunes y que pueden ser utilizados con diferentes fines.

Muchos de estos modelos están basados en rasgos[1][2]. Y junto con cada modelo se crean tests de personalidad que, de forma directa, son utilizados para identificar los rasgos que cada individuo tiene, siendo la personalidad de cada uno un conjunto de varios o todos los rasgos de un modelo, cuantificándolos de forma individualizada.

Los métodos tradicionales que se han utilizado en psicología para clasificar la personalidad están basados en métodos directos: test psicológicos y entrevistas clínicas[2]. En ellos se busca dar respuesta a determinados ítems o cuestiones que permitirán, mediante fórmulas estadísticas, clasificar al individuo objeto del estudio.

Parece claro que estos métodos directos no siempre son aplicables. En particular, no son los más adecuados cuando se está tratando de clasificar masivamente a un grupo de población o a un individuo que no se debe sentir estudiado. Los sujetos de los estudios o investigaciones no tienen siempre por qué conocer que está siendo objeto de estudio de su personalidad (algo que las empresas de marketing realizan de forma permanente y sin realizar test alguno). Por lo tanto, los métodos directos tampoco son los más adecuados en un entorno policial, por ejemplo, donde el agente necesita de forma rápida y sin que el sospechoso se dé cuenta, inferir su personalidad.

La perfilación indirecta surge para dar respuesta a este tipo de necesidades como metodología igualmente válida que la directa para inferir la personalidad de los individuos estudiados [2][3]. Cuando la evaluación directa no se pueda realizar, el perfilado deberá realizarse por medio de la observación de la propia persona, de su entorno, de su comportamiento y de su lenguaje (verbal y no verbal) en lugar de acudir a las pruebas psicométricas y evitando que los sujetos se sepan evaluados.

Dado que la interrogación directa del sujeto sometido a evaluación no siempre es viable, claramente cuando se trata de un sospechoso o de una posible fuente de información que hay que captar, este modo indirecto de operar, es decir, a través de la observación del comportamiento se erige en modo imprescindible para establecer los rasgos que definen la personalidad del individuo [2][3]. Por ejemplo, observación de movimientos, forma de hablar, de caminar, rutinas diarias...y por supuesto, el lenguaje

que utiliza cada individuo. Las investigaciones científicas en psicología de la personalidad han ido asociando cada uno de estos comportamientos con ciertos patrones estables del carácter, de modo que no sean necesarias las pruebas estadísticas directas basadas en tests para tener una idea del tipo de personalidad del individuo observado.

Actualmente las redes sociales (RRSS) constituyen una fuente inmensa de información que puede ser utilizada con distintos fines y por diferentes tipos de usuarios. La mayoría de la población las utiliza con fines lúdicos o comunicativos, pero existen objetivos más prácticos si se habla del estudio de la personalidad. Hay teorías que exponen que los individuos *seleccionan y crean sus ambientes sociales* para encajar y reforzar sus características personales en los mismos [1][10][10]. Es sobradamente conocido que las empresas de marketing estudian los perfiles de los usuarios de RRSS para enfocar más adecuadamente sus campañas.

1.1 Contexto del trabajo

Las Fuerzas y Cuerpos de Seguridad del Estado (FFCCSE) también tienen el foco puesto en estas redes como un input más en sus investigaciones. La extracción, procesamiento y uso de esta ingente cantidad de información se ha convertido en el objetivo de muchos organismos públicos con la intención de dar solución a cuestiones clave en diferentes investigaciones. La importancia que tiene el perfilado indirecto con este objetivo es clara.

Todos estos cuerpos tienen sus servicios de información los cuales necesitan encontrar fuentes, sujetos fiables que les proporcionen información en sus investigaciones. De estos sujetos, se necesita elaborar sus perfiles de personalidad en base a toda la información posible que se disponga de ellos. Buscar a quien captar para tener la información más precisa en las investigaciones es una tarea fundamental. Y, por otra parte, la búsqueda de determinados perfiles con características determinadas en el marco de una investigación resulta de especial importancia por motivos obvios.

Otro uso claro del estudio de la personalidad es entender en que se basan las personas para tomar sus decisiones de actuación para poder influir en esas decisiones y en sus comportamientos.

Esta introducción trata de dejar claro que la personalidad es el conjunto de respuestas conductuales que da el individuo ante diferentes situaciones. La personalidad está determinada por rasgos característicos, siendo estos medibles de forma directa mediante tests psicológicos o indirecta mediante la observación de la propia conducta.

El doctor James W. Pennebaker [12][13], profesor de Psicología del Centenario de Artes Liberales en la Universidad de Texas en Austin, ha centrado sus estudios dentro del campo de la psicología, concretamente en el ámbito del estudio del lenguaje humano. Es autor de la siguiente frase: “Eres lo que hablas”. El lenguaje es vehículo de expresión de la personalidad de los individuos. El lenguaje permite expresar pensamientos, sentimientos y emociones, todo un conjunto de datos que parecen claves para inferir la personalidad. Pennebaker sostiene que no es tan importante el contenido del mensaje expresado como la forma en la que se expresa, es decir, qué palabras son utilizadas. Y no las palabras que dominan semánticamente un texto, palabras «grandes», sino el estilo de un texto que se expresa a través de las palabras «pequeñas», entre otros elementos, ya que resultan poco accesibles a la manipulación consciente. Así, la persona que quiere resultar agradable elige las expresiones adecuadas, pero no controla el número de artículos o pronombres (palabras pequeñas) que utiliza para ello. Da una base científica bastante clara sobre cómo se puede inferir la personalidad a partir del lenguaje. En diferentes artículos del FBI [14][15] se explican cómo se utilizan los estudios de Pennebaker entre otros para las investigaciones que se llevan a cabo. Estas investigaciones tienen las redes sociales como un medio más para obtener información sobre los investigados, ya que tal y como indican en un artículo del sitio web del FBI leb.fbi.gov [14]: “Words provide a window into the minds of criminals, helping to determine whether they fit any particular personality profile, such as psychopathy” (Las palabras aportan una

ventana a la mente de los delincuentes, ayudando a determinar si se ajustan a un determinado perfil de personalidad, como la psicopatía).

ENCUIST[4] es un modelo de personalidad desarrollado por la psicóloga Lucia Halty[3] directamente para trabajar en entornos policiales, ya que observa rasgos de interés en estos entornos. Permite realizar un perfilado indirecto de los individuos buscando y analizando variables que se pueden obtener de la observación del comportamiento. Este modelo fue elaborado por su autora en 2017 tras dos años y medio de investigaciones. Está muy orientado al perfilado indirecto en base a textos extraídos de charlas o entrevistas con los individuos estudiados, por lo que lo hace un buen modelo para trabajar en un sistema automático de perfilado indirecto basado en textos. Existen, además, muchos estudios universitarios relativos a este modelo que permiten tener documentación de base. Además, está centrado en el idioma español.

Se expondrá la base científica necesaria para poder extraer la personalidad de los individuos en base al modelo ENCUIST a partir de su lenguaje escrito. No se pretende entrar en profundidad en los estudios psicológicos sobre el estudio de la personalidad, pero es necesario, en apartados posteriores, hacer una introducción a los diferentes modelos de personalidad sobre los que trabajan los psicólogos para entender por qué se decide utilizar ENCUIST.

Una vez descrito el modelo psicológico, se darán las bases para diseñar un sistema automático que, basado en el procesamiento de la información de las redes sociales, permitirá de forma automática extraer la personalidad de los individuos que están tras estos perfiles. La aplicación de este sistema automático se podrá orientar a un único perfil o a grupos de perfiles concretos.

1.2 Motivación

Se pretende plantear el diseño de una herramienta que sirva, en un futuro desarrollo, para los servicios de información de las FFCCSE españoles donde la mayoría de las investigaciones se realizan en castellano. Es por ello por lo que este estudio, fundamentalmente en lo relacionado con PLN, lo hace circunscrito al idioma español. Ya que ENCUIST es un modelo con resultados sobre población española, únicamente se puede hablar de idioma y cultura españoles. Toda la documentación encontrada al respecto (base para la extracción de requisitos), se centra en estudios sobre población española. Si se trabajara con otras lenguas, habría necesidad de tener disponibles otros perfiles como lingüistas y psicólogos expertos en la cultura a la que pertenecieran los objetivos, para que ayudaran a confeccionar diccionarios en esos idiomas y datos de prueba/entrenamiento cultural y lingüísticamente válidos.

Como se ha indicado, es muy importante en una investigación de FFCCSE determinar la personalidad de los individuos investigados, y en estadios tempranos de la investigación no se cuenta con la participación de los sujetos para que realicen test psicológicos[3]. Encontrar, de forma rápida y efectiva, información sobre su personalidad ayudará en el transcurso de la investigación. Esto es aplicable también al *discovery de objetivos* en búsquedas exploratorias sobre información masiva dentro de RRSS, por ejemplo, buscar al líder dentro de un determinado grupo de personas o aquella que resulte más influenciable. Los perfiles buscados cumplirán unas determinadas características[14].

Para este objetivo, las redes sociales suponen una fuente inestimable de información[1][10] ya que un sector muy amplio de la sociedad hace uso de ellas para expresarse y relacionarse. Es en esta comunicación donde los individuos dejan marcada la impronta de su personalidad y, como se ha comenzado a ver en la introducción, la personalidad y el comportamiento van de la mano. Una determinada personalidad, ante una situación concreta reaccionará de formas determinadas.

Como se ha introducido, el lenguaje utilizado por las personas “exfiltra” características de su personalidad[12]. Por tanto, se buscará extraer los indicadores léxicos y los metadatos que sirvan para inferir los rasgos de los individuos a través de los textos que cuelguen en sus redes sociales (en concreto Facebook y Twitter). Para ello, se han analizado varios trabajos del campo de la psicología y se ha

consultado a expertos en la materia. De esta investigación y de estas consultas se extrae el conocimiento experto y los indicadores necesarios para inferir los perfiles. Sin embargo, al no haberse encontrado una cuantificación directa de los indicadores en los trabajos previos, es necesario que el sistema planteado haga una cuantificación inicial de los indicadores para después poder realizar la clasificación del análisis de perfiles nuevos en sus rasgos de personalidad. Por ejemplo, la tesis doctoral de González Elices[2] habla del número de apariciones de una palabra o de un determinado grupo de palabras, lo hace indicando términos de “Menor/mayor uso de palabras en el discurso...” pero no hay valores numéricos con los que los compara, lo que no es aplicable directamente como forma de medida en un sistema informático. Sin embargo, es información muy valiosa para extraer requisitos de cuáles deben ser los indicadores.

Esta cuantificación inicial o entrenamiento del sistema, se plantea a través de tecnología de Machine Learning (de aquí en adelante, ML).

1.3 Objetivo

El objetivo es plantear un posible diseño de un sistema que permita perfilar a un individuo o a un conjunto de ellos en base a la información de sus redes sociales, utilizando indicadores del lenguaje y metadatos asociados que se pueden deducir del modelo ENCUIST, una vez analizado en detalle.

No forma parte de los objetivos planteados el desarrollo e implementación de los sistemas en sí mismos como modelo final.

El objetivo del sistema a diseñar es el de analizar perfiles de redes sociales para, en base a unos indicadores seleccionados del modelo ENCUIST, utilizado para el perfilado indirecto de la personalidad, clasificarlos en los diferentes rasgos. El objetivo esperado por el sistema será inferir la personalidad de los individuos que hay detrás de esos datos descargados de RRSS, pudiéndose a posteriori realizar informes o análisis sobre esa información. Partiendo de los indicadores textuales extraídos de la información experta, el diseño final deberá cumplir los siguientes requisitos de alto nivel:

1. El sistema deberá ser capaz de descargar en modo texto perfiles de RRSS junto con sus metadatos y almacenarlos.
2. Deberá ser capaz de realizar un análisis de los textos descargados aplicando PLN e identificar y extraer los datos que sean relevantes para realizar un perfilado indirecto de la personalidad calculando los indicadores que hayan sido seleccionados a partir del modelo ENCUIST.
3. Realizará una clasificación de los perfiles en base a los indicadores y hará una asignación a perfiles ENCUIST.
4. Dispondrá de un módulo de informes que podrá ser interrogado por los usuarios para contestar a las preguntas y análisis que demanden.

Para realizar el objetivo planteado se utilizarán los principios de la arquitectura software hasta la finalización de la fase de diseño y propuestas para su continuación en fases posteriores del ciclo de vida que se podrán abordar a futuro.

Ante el arranque o planteamiento de cualquier sistema o proyecto informático se hace necesario su visión a través de la disciplina de ingeniería de software. De esta manera se puede abordar de forma metódica y estructurada las diferentes etapas por las que el proyecto software irá pasando, desde su análisis y diseño, hasta su programación y pruebas, si fuera necesario.

Este trabajo se va a centrar en las etapas de análisis y diseño, dejando para futuros trabajos el desarrollo y las pruebas.

Existen diferentes modelos y metodologías a la hora de gestionar proyectos. De forma muy esquematizada hoy en día se pueden dividir en:

-
- Metodologías ágiles: Según la definición que nos proporciona Atlassian[72] “es un método iterativo de llevar a cabo proyectos que se basa en realizar publicaciones de forma continua y en integrar el *feedback* de los clientes. La posibilidad de hacer ajustes durante cada iteración fomenta la velocidad y la adaptabilidad.” Estas metodologías, por tanto, están hechas para dar resultados útiles y rápidos en entornos que sufren continuos cambios. La gestión ágil de proyectos es también un pilar de las prácticas de *DevOps*, en las que los equipos de desarrollo y de operaciones trabajan en colaboración.
 - Metodologías en cascada: Atlassian[72] define también el modelo de gestión de proyectos en cascada, el cual implica una secuencia de ejecuciones claramente definida, con proyectos que no pasan de fase hasta que la anterior recibe la aprobación final. El modelo de gestión de proyectos en cascada sigue un enfoque lineal y secuencial. Funciona bien para trabajos que implican procesos predecibles y recurrentes”. Es decir, que, si están claramente definidos los requisitos del sistema, y estos son bastante inamovibles, el modelo lineal en cascada es válido.

Por lo tanto, dado que el proyecto software que se va a llevar a cabo está bastante definido y no va a haber cambios sustanciales durante el tiempo de desarrollo de este trabajo, se va a optar por plantear un modelado tradicional en cascada.

1.4 Organización del documento

A lo largo del documento se van a cubrir los aspectos necesarios para realizar una propuesta de diseño del sistema objetivo, realizando un recorrido por los siguientes puntos:

1. Conocer profundamente el **contexto del negocio** para poder identificar sus necesidades y construir soluciones acordes a ellas. Esto permitirá determinar los requisitos funcionales y no funcionales del nuevo software.
2. Diseñar la estructura del sistema **identificando subsistemas**, componentes, interfaces de comunicación.
3. **Conocimiento del mercado tecnológico** para posibilitar la elección de la tecnología adecuada para cada pieza del sistema, de tal manera que cumpla los requisitos planteados. Para ello es necesario **conocer** que **tecnologías** existen en el mercado que puedan ser solución satisfactoria para los requisitos marcados y cuáles son sus ventajas y desventajas.
4. Elección de los **flujos de trabajo** y estándares adecuados para la consecución del desarrollo.

En la Figura 1-1 Componentes del diseño propuesto y su relación se pueden ver gráficamente y en forma secuencial los diferentes aspectos que se van a abordar.

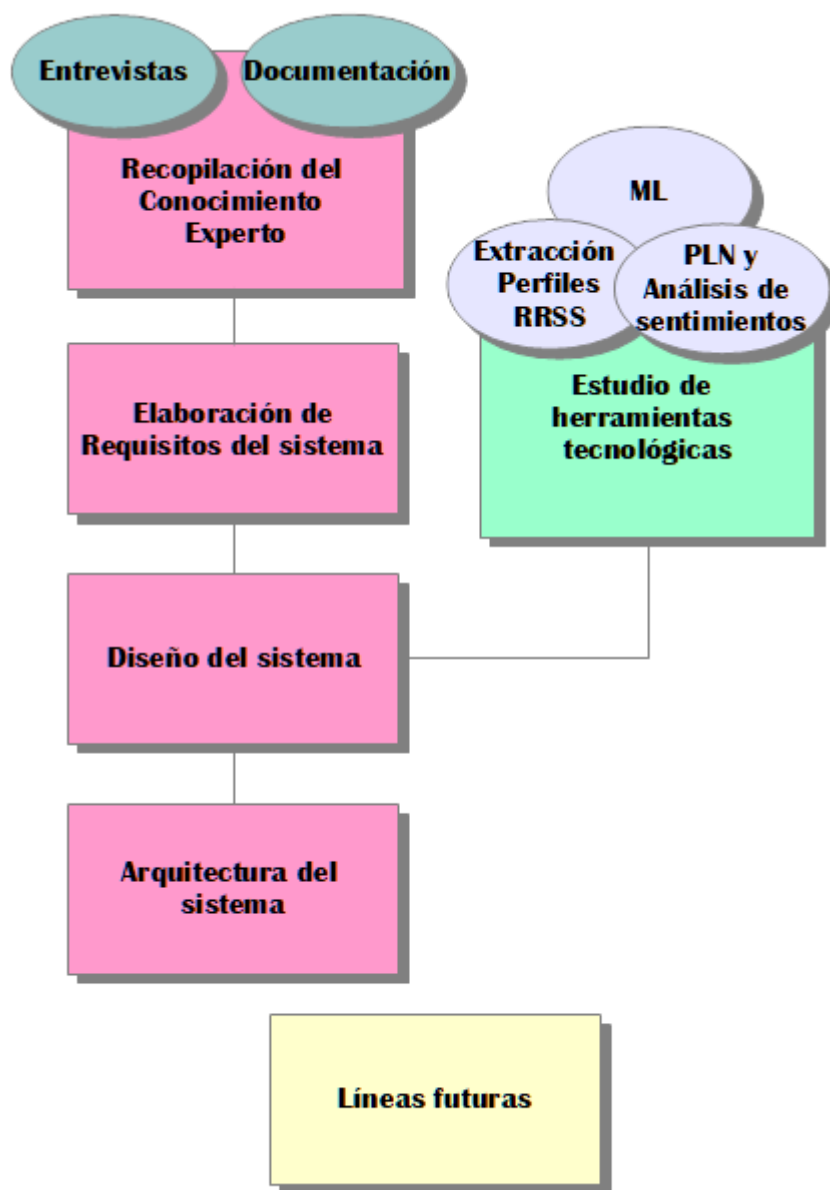


Figura 1-1 Componentes del diseño propuesto y su relación

Explicados uno a uno:

- El **estudio de toda la información experta** necesaria **para definir los requisitos del sistema**. Se pone de manifiesto en este punto la importancia para el personal de perfil técnico el contar con expertos y documentación sobre el campo sobre el que se va a trabajar para implantar el sistema de información adecuado. Si no se conoce el negocio no se puede crear un sistema que le de servicio.
- El **estudio de los diferentes aspectos tecnológicos** que puedan **dar solución a los requisitos**. Para ello se ha hecho un recorrido por diferentes campos de la informática actual que se han visto necesarios para cubrir los requisitos. Esto incluye: el acceso a la información de redes

sociales, el PLN, el análisis de sentimientos que se traslucen de un texto y modelos de ML para este fin.

El análisis de sentimientos se ha considerado como un buen complemento para dar más información sobre los textos que se analizasen, siendo un indicador de las emociones que están detrás de esas publicaciones.

- El **diseño y arquitectura de** los diferentes **módulos que compondrán el sistema** y las librerías y herramientas que serán necesarias para el futuro desarrollo.
- Por último, se plantea la **conclusión y las líneas futuras** que se podrán realizar en ciclos de vida posteriores.

2 RECOPIACIÓN CONOCIMIENTO EXPERTO

Estudiando el conocimiento experto se pretende comprender el contexto del sistema que se va a diseñar. Además, permitirá realizar la captura de requisitos necesarios para la fase de diseño del sistema y el establecimiento de los indicadores del lenguaje que se requieren.

2.1 Perfilado indirecto de la personalidad

2.1.1 Personalidad y comportamiento.

En la introducción se expuso la idea de que la personalidad de los seres humanos se ve reflejada en su conducta.

Una definición bastante completa y llena de significado de la personalidad es “la construcción hipotética que inferimos de la conducta de las personas” [5]. Esta construcción recoge una serie de elementos bastante estables y consistentes en el tiempo llamados rasgos. La personalidad queda marcada por la experiencia vital, el aprendizaje y por la biología de cada individuo. Por tanto, la personalidad es única y característica de cada ser humano. Sin embargo, hay características comunes que permiten a los psicólogos realizar clasificaciones en función de los *tipos de personalidad* [5][6][11].

Las teorías sobre personalidad son muy diversas y se remontan a principios del siglo XX. Las diferentes teorías de la personalidad que se han formulado desde entonces se agrupan en tres modelos[2][16]:

1. Personalista: Los comportamientos dependen de factores internos de los sujetos. Cómo estos factores son estables, los comportamientos son predecibles. **Este planteamiento teórico entiende a la persona** como un ser activo y determinante de su conducta. Por lo tanto, en este modelo lo importante es conocer los rasgos de la personalidad de cada uno de los individuos. **La personalidad se mantendrá estable a lo largo del tiempo y en distintas situaciones.** De esta manera, **si se consigue aislar los rasgos de una persona, se puede prever su comportamiento futuro.** Desde este planteamiento han nacido multitud de modelos con sus respectivos tests que intentan medir la personalidad, como el modelo de los Big Five y el modelo de Eysenck[11], explicados más en profundidad posteriormente.
2. Situacionista: Al contrario que el anterior enfoque, **este entiende a la persona como un sujeto pasivo e influenciado totalmente por el contexto.** En este modelo no importan los rasgos y cualidades de una persona, el mayor peso recae en la situación. Según este enfoque las causas que determinan la conducta de los sujetos son factores externos. Por lo tanto, hay que estudiar los factores ambientales para determinar el comportamiento.

Este modelo **se basa en el supuesto de que toda la conducta es aprendida.** Por ello, deben estudiarse los procesos de aprendizaje por los que adquirimos nuevas formas de actuar.

Este modelo tiene a simplificar bastante ya que **la actitud de un sujeto afecta a su conducta**, si esto no fuera así, todas las personas se comportarían igual en la misma situación.

3. Interaccionista: Este modelo sostiene que el comportamiento de los individuos está determinado por la suma continua de situación y biología. Las variables personales y ambientales son igual de importantes. Es un intento de aunar las dos perspectivas anteriores

y solventar sus problemas. Desde este paradigma se entiende que **la conducta está determinada por la interacción entre las variables personales del sujeto y las variables situacionales**. Por lo tanto, según esta perspectiva la personalidad es producto de la interacción del sujeto con su contexto. Este planteamiento, aunque muy completo, es en realidad muy difícil de abordar y trabajar.

Por lo tanto, los modelos basados en la teoría personalista son aquellos que definen la personalidad como algo estable a lo largo del tiempo y en diferentes situaciones. Deduciendo entonces que las conductas de los individuos son predecibles.

Son estos modelos la base teórica de los requisitos que se necesitan para el sistema a diseñar. Ya que se necesita partir del principio de que la personalidad de un individuo es estable.

No se va a realizar una explicación en profundidad todos los modelos psicológicos que se han teorizado a lo largo de la historia de la psicología, pero sí es necesario introducir los dos modelos antes mencionados y que dan origen al que se va a exponer en los siguientes apartados como herramienta a la hora de perfilar de forma indirecta a los individuos. Los modelos de Eysenck o de tres factores y el de los cinco grandes (*Big-Five*) de McCrae y Costa.

1. **La teoría de la personalidad de Eysenck**[1][2][3][11]: Su enfoque se enmarca en la *teoría de los rasgos*, que asume que la conducta está determinada por unos atributos relativamente estables que son las unidades fundamentales de la propia personalidad, porque predisponen a una persona a actuar de una manera determinada. Esto significa que los rasgos deben ser coherentes a través de situaciones y con el tiempo, pero pueden variar entre los individuos.

Lo que propone Eysenck es que cada persona nace con una estructura específica a nivel cerebral, que causa discrepancias en la actividad psicofisiológica y, por tanto, provoca que el individuo desarrolle diferencias en el mecanismo psicológico, determinando un tipo específico de personalidad.

Como conclusión identificó tres dimensiones independientes: Psicoticismo (P), Extroversión (E) y Neuroticismo (N), por lo que se denomina modelo PEN.

- **Psicoticismo:** Las personas con puntuaciones altas en psicoticismo se caracterizan por ser insensibles, inhumanas, antisociales, violentas, agresivas y extravagantes. Estas puntuaciones altas se relacionan con distintos trastornos mentales, como la propensión a la psicosis. En contraste con las otras dos dimensiones, el psicoticismo no cuenta con un extremo inverso, sino que es un componente presente en distintos niveles en la persona.

Este modelo es personalista, por tanto, Eysenck aporta explicación biológica de psicoticismo, concretamente de las hormonas gonadales como la testosterona y las enzimas como la monoamino oxidasa (MAO). A pesar de no haber gran cantidad de investigaciones sobre el psicoticismo en comparación con la extraversión y el neuroticismo, algunos estudios actuales muestran que las personas con episodios psicóticos presentan niveles altos de testosterona y niveles bajos de la MAO.

Además, en estos estudios, la impulsividad y la agresividad, dos rasgos característicos de los individuos que puntúan alto en psicoticismo, correlacionaron negativamente con la MAO, pues esta enzima desempeña un papel fundamental en la degradación de las monoaminas noradrenalina, dopamina y serotonina. Por lo tanto, se evidencia el papel de la biología en este rasgo.

- **Extraversión (extraversión o introversión):** Los extrovertidos se caracterizan por la sociabilidad, impulsividad, desinhibición, vitalidad, optimismo y agudeza de ingenio; mientras que los introvertidos son tranquilos, pasivos, poco sociables, atentos, reservados, reflexivos, pesimistas y tranquilos. Eysenck postuló que la principal diferencia entre los extrovertidos y los introvertidos se encuentra en el nivel de activación cortical.

La activación cortical de Eysenck propone una explicación biológica de la extraversión teniendo en cuenta el sistema de activación reticular ascendente (SARA). La actividad del SARA estimula la corteza cerebral, que, a su vez, aumenta el nivel de activación cortical.

El nivel de arousal cortical se puede medir a través de la conductancia de la piel, las ondas cerebrales o el sudor. Teniendo en cuenta los distintos niveles de actividad del SARA, los introvertidos tienen niveles más elevados de actividad que los extrovertidos. Algunas investigaciones han demostrado que los extrovertidos buscan fuentes de estimulación externa que les provocan un mayor nivel de estimulación. De nuevo la base biológica queda evidenciada en este rasgo.

- **Neuroticismo (estabilidad o inestabilidad emocional):** Las personas con inestabilidad emocional muestran ansiedad, histeria y obsesión. Con frecuencia tienden a reaccionar de forma emocionalmente exagerada y tienen dificultad para volver a un estado normal después de la activación emocional. En el otro extremo la persona es ecuaníme, calmada y con un alto grado de control emocional.

Eysenck también explica el neuroticismo en términos de los umbrales de activación del sistema nervioso simpático o el *cerebro visceral*. El cerebro visceral también se conoce como el sistema límbico, que consiste en el hipocampo, la amígdala, el septo, y el hipotálamo, y regula los estados emocionales tales como el sexo, el miedo y la agresión. Es responsable de la respuesta de lucha o huida en la cara del peligro.

Para medir los niveles de activación del cerebro visceral se puede utilizar la frecuencia cardíaca, la presión arterial, la conductancia de la piel, sudoración, la frecuencia respiratoria y la tensión muscular (especialmente en la frente). Las personas neuróticas tienen umbrales bajos de activación del cerebro visceral y son incapaces de inhibir o controlar sus reacciones emocionales. Biología una vez más.

2. **La teoría de los Big Five o de los Cinco Grandes [1][2][3][5][6]:** En realidad es una taxonomía de modelos anteriores a él. En 1980 McCrae y Costa clasificaron los rasgos de la personalidad en cinco grupos: Extraversión (E), Neuroticismo (N), Amabilidad (A), Apertura a la experiencia (O), Responsabilidad (R). Las iniciales de estos rasgos en inglés forman la palabra OCEAN, que es el acrónimo en inglés de este modelo. Hoy en día es uno de los modelos más reconocidos en la comunidad internacional del campo de la psicología.

Con anterioridad a los Big Five, Raymond Cattell expuso su teoría de la personalidad basada en 16 factores. Muchos investigadores creyeron que la teoría de Cattell era demasiado complicada y que la de Eysenck tenía un alcance demasiado limitado. Como consecuencia de lo anterior, surgió la teoría de los cinco factores para describir los rasgos esenciales que sirven como componentes básicos de la personalidad.

- **Extraversión:** A través de este rasgo se evalúa la disposición del individuo frente a las interacciones personales, es decir, el nivel de actividad y la estimulación que tiene la persona a la hora de relacionarse con los demás. También está íntimamente relacionado con el grado de placer o disfrute que se adquiere a través de las relaciones sociales. Altas

puntuaciones nos indican que nos encontramos frente a individuos sociables, habladores, orientado a las personas, optimistas, disfrutan ser el centro de atención, les resulta fácil hacer nuevos amigos, son amantes de la diversión y afectuosos. Por otro lado, bajas puntuaciones nos señalarían individuos que les resulta difícil hacer amigos e iniciar conversaciones, reservados, sobrios, no eufóricos, solitarios, callados o retraídos.

- **Neuroticismo:** Con el neuroticismo, lo que se busca evaluar es el ajuste emocional comparado con la inestabilidad emocional. Este rasgo identifica a individuos con tendencia al malestar psicológico, ideas irreales, excesiva rumiación o ansia y respuestas de afrontamiento desadaptativas.

Una puntuación elevada en neuroticismo sería indicadora de personas preocupadas, nerviosas, emocionales, sensibles al estrés, inseguras o hipocondríacas. Por el contrario, aquellos individuos con una baja puntuación se caracterizarían por ser relajados, no emocionales, seguros, estables emocionalmente, resilientes, autosatisfechos, no se preocupan mucho y tienen alta autoestima.

- **Amabilidad:** Este rasgo es útil para conocer la calidad de la orientación interpersonal. La amabilidad se presenta a lo largo de un continuo, que va desde la compasión hasta el antagonismo en los pensamientos, los sentimientos y las conductas. Es decir, tiene que ver con hasta qué punto el individuo es capaz o no de ponerse en el lugar de otro y actuar acorde a sus emociones y preocupaciones.

Si un individuo puntúa alto en amabilidad, será compasivo, afable, confiado, atento, no rencoroso, crédulo o franco. En cambio, una puntuación baja nos muestra una persona cínica, ruda, suspicaz, competitiva, vengativa, despiadada, irritable, indiferente y manipuladora.

- **Apertura a la experiencia:** Este rasgo evalúa hasta qué punto un individuo realiza una búsqueda activa de nuevas experiencias y disfruta con las actividades poco rutinarias. Básicamente, se encarga de medir la tolerancia y la exploración de situaciones no familiares.

Una puntuación elevada en este rasgo muestra personas curiosas, abiertas a probar cosas nuevas, con amplios intereses, creativas, originales, imaginativas o no tradicionales. En contraposición, si la puntuación es baja, serán personas que no les gusta el cambio, no disfrutan de las cosas nuevas, son convencionales, pragmáticas, realistas y con pocos intereses.

- **Responsabilidad:** Se encarga de evaluar el grado de organización, persistencia y motivación para la conducta dirigida a metas. Diferencia a las personas entre aquellas que se dedican a que el resultado de su trabajo sea óptimo y perfecto y aquellas que no se preocupan por los resultados de su desempeño.

Si se puntúa alto en responsabilidad estaremos ante individuos organizados, planificadores, formales, trabajadores, puntuales, ordenados, atentos a los detalles, ambiciosos o perseverantes. Y aquellos que tengan puntuaciones bajas se caracterizarán por ser personas sin objetivos, informales, perezosos, descuidados, indisciplinados, procrastinadores y con poca voluntad.

Merece la pena en este punto recordar un escándalo asociado a los BigFive y a Facebook[17]. En 2013 unos 265.000 usuarios completaron un test psicológico publicado por Facebook que requería permiso para acceder a información personal y de la red de amigos, sin el consentimiento de estos últimos. En 2016, Cambridge Analytica, una empresa de análisis de datos especializada

en elecciones electorales y en marcas, cruzó los datos del test de personalidad con la información de Facebook para inferir perfiles psicológicos de cada usuario. Así, Cambridge Analytica logró saber cuál debía ser el contenido y los temas los mensajes para cambiar la forma de pensar de los votantes de forma casi individualizada. Cuando esta noticia salió a la luz, hizo perder a Facebook 37000 millones de dólares en un día.

Sin embargo, esto evidencia el gran instrumento que supone conocer la personalidad de las personas.

2.1.2 *Modelo ENCUIST*

ENCUIST es un modelo personalista basado en los dos modelos anteriores y muy adecuado para el perfilado indirecto, ya que su diseño tuvo este objetivo como meta [2][3][4].

Al igual que los modelos tradicionales, ENCUIST está acompañado de un test Anexo I: Test ENCUIST. Está compuesto por 137 ítems, 5 relacionados con datos sociodemográficos (edad, sexo, nacionalidad, país de residencia y nivel educativo), 4 relacionados con variables delictivas, victimológicas y de relación, y 128 sobre personalidad, los cuales permiten identificar y medir los siguientes rasgos de la personalidad: Extroversión/búsqueda de sensaciones, impulsividad/agresividad, neuroticismo (ansiedad, ira y asco), insensibilidad emocional y necesidad de cognición. Sin embargo, ENCUIST va acompañado de un procedimiento rápido y ágil de observación basado en entrevistas a los sujetos. Este procedimiento se basa en la relación que existe entre la personalidad y los diferentes canales de comunicación que utilizan los individuos. En las entrevistas ENCUIST se abordan temas cotidianos con el fin de estudiar tanto el lenguaje verbal como el no verbal de población no clínica.

Las entrevistas clínicas estructuradas y los tests psicológicos, métodos tradicionales y perfectamente fiables para elaborar perfiles psicológicos, requieren la participación consentida e informada por parte del sujeto del estudio. Este se sabe estudiado y debe colaborar contestando a las preguntas que le plantea el experto. Sin embargo, esto no es posible en entornos de investigación, ya sean realizados por parte de la Policía u otros FFCCSE. Como se ha introducido, si se necesita entrevistar a un posible sospechoso y hacerse una idea de su personalidad de forma rápida, no se le puede realizar un test. De igual manera no se puede elaborar una prueba escrita consentida en la busca de posibles perfiles peligrosos en RRSS a toda la población objetivo. De ahí que sea necesario el uso de perfilado indirecto.

Se utiliza como requisitos del sistema las conclusiones a las que llega en modelo ENCUIST. Es decir, este modelo es la base psicológica que utiliza para diseñar un sistema que permita realizar un perfilado indirecto de individuos en base a texto descargado de sus perfiles de redes sociales.

Llegado este punto, es necesario explicar los rasgos de personalidad que propone ENCUIST, basándose en los modelos que se han explicado previamente Eysenck y Big Five.

Particularmente, ENCUIST está compuesto por los siguientes rasgos de personalidad:

- **Extroversión/búsqueda de sensaciones (E):** Como se expuso anteriormente, Eysenck indica que la extroversión dependía de los niveles basales de arousal del neocórtex. Ya se vio que el término arousal se refiere al nivel de activación cerebral. Sugiere que los extrovertidos están corticalmente menos activados que los introvertidos. De ahí que tengan ciertas características en las que se demuestra su gusto por estímulos que les hagan elevar el nivel de arousal, como por ejemplo estímulos lumínicos fuertes o sabores picantes.

La emoción de alegría está vinculada al rasgo de extroversión/búsqueda de sensaciones. Es una emoción que genera conductas de aproximación y búsqueda de refuerzos positivos.

Resumen: Sociabilidad y búsqueda de sensaciones.

- **Neuroticismo (ansiedad, ira y asco) (N):** Eysenck indicó que este rasgo estaba asociado al sistema límbico. Las emociones básicas son miedo, tristeza, ira, alegría y asco. Las personas con puntuaciones elevadas en este rasgo tienden a activarse emocionalmente con mucha rapidez y esa activación es más persistente en el tiempo, les cuesta salir de ese estado emocional. La alegría se relaciona más con el rasgo de la extraversión como se ha comentado en el apartado anterior. Tradicionalmente se habla de que los individuos con puntuaciones elevadas en neuroticismo suelen ser susceptibles a problemas basados en la ansiedad, el miedo y la tristeza. Además, se considera que el sujeto tiende a la inestabilidad emocional cuando experimenta elevados niveles de la emoción de asco. Por último, la ira es una emoción universal que todo el mundo experimenta en algún momento, pero algunos con excesiva frecuencia e intensidad, por lo que puede resultar disfuncional.

Resumen: Inestabilidad emocional. Emociones básicas

- **Insensibilidad emocional (CU):** Los sujetos con elevadas puntuaciones en insensibilidad emocional se caracterizan por presentar rasgos de personalidad como crueldad, falta de empatía, ausencia de culpa, falta de remordimientos y sentido desmesurado de autovalía. Son personas que tienen dificultades en el reconocimiento y reactividad de determinadas emociones, en concreto el miedo. A nivel biológico este rasgo se relaciona con un mal funcionamiento de la amígdala, ya que es una estructura implicada en el reconocimiento de expresiones emocionales. Tal y como se indica en el modelo ENCUIST[4] mientras que unos sujetos pueden ser agresivos por una hiperactividad del sistema de amenaza, los sujetos con puntuaciones elevadas en el rasgo de CU pueden ser agresivos debido a una falta de activación por parte de la amígdala, que resulta en una ausencia de miedo y falta de empatía hacia los otros.

Resumen: Ausencia de culpa, falta de empatía, sentido desmesurado de autovalía.

- **Impulsividad/agresividad (I):** En este perfil nos encontramos con sujetos típicamente antisociales, que transgreden normas y tienen dificultades en el control de impulsos ejecutando una respuesta impulsiva agresiva. La ira tiene un importante papel en la agresión. Según los expertos, el córtex prefrontal se encarga de regular las emociones para dar al individuo una respuesta más adaptativa, produciendo una inhibición de la ira.

Resumen: Transgresor de normas, dificultad en el control de impulsos.

- **Necesidad de cognición (NC):** Las personas con alta NC poseen una fuerte motivación intrínseca para realizar tareas mentales, lo cual los lleva a buscar información y a pensar detenidamente sobre la misma. Puesto que las personas con alta NC forman y cambian sus actitudes a través de la elaboración detallada de la información, sus actitudes suelen resultar más fuertes que aquellas modificadas a través de baja elaboración cognitiva.

Resumen: Realización de tareas mentales. Búsqueda de información detallada.

2.1.3 Indicadores para el perfilado

Trabajos realizados sobre ENCUIST han realizado entrevistas y test de personalidad a un conjunto de individuos[2]. Una vez analizados estos datos se han identificado una serie de indicadores léxicos que servirán como base a los requisitos del sistema que pretende diseñar. Se usarán, por tanto, una serie de indicadores que tienen una relación directa con los rasgos del modelo de personalidad ENCUIST.

Una vez entendidos los indicadores y su base científica, se utilizarán técnicas de procesamiento de lenguaje natural para plantear un sistema que extraiga la información necesaria de los perfiles descargados y que permita inferir la personalidad utilizando el modelo ENCUIST.

En realidad, se podrían utilizar otras fuentes de información textual de los individuos, pero como se comentó en la introducción, las redes sociales dan acceso a una ingente cantidad de información de una masa enorme de población que han depositado libremente y que puede ser estudiada sin que ellos se sientan objeto de investigación.

2.1.3.1 Indicadores verbales para los rasgos ENCUIST.

Del estudio sobre ENCUIST realizado a través de test y entrevistas [2] se extraen los siguientes indicadores léxicos relacionados con los rasgos de personalidad de este modelo. Tal y como se explicó en la introducción, la relación entre las palabras de las entrevistas y el número de veces que aparecen es cualitativa, no cuantitativa. Durante la fase de Desarrollo se expondrá un método para conseguir cuantificar de manera automática.

1. Extroversión.

- Mayor uso del yo y menor del él/ella.
- Mayor uso de verbos en 1ª persona
- Mayor uso de palabras de entendimiento (pensar, considerar, etc.)
- Mayor uso de palabras relacionadas con la familia y amigos.
- Menor uso de interjecciones
- Menor uso de palabras de función.

● Búsqueda de sensaciones.

- Menor uso de pronombres y de pronombres personales, concretamente de la 1ª persona del singular.
- Menor uso de verbos, del uso presente, de verbos en 3ª persona (tanto del singular como del plural) y del subjuntivo.
- Mayor uso de preposiciones.
- Menor uso de conjunciones.
- Menor uso de palabras relacionadas con la familia.
- Menor uso de palabras relacionadas con procesos afectivos y de emociones positivas.
- Menor uso de palabras de exclusión (pero, sin, aunque, etc.)

● Apertura a la experiencia:

- Mayor uso de palabras en sus respuestas.
- Menor uso de palabras de función.
- Mayor uso de artículos.
- Mayor uso del tiempo pasado y verbos en 3ª persona del singular.
- Mayor uso de números.
- Mayor uso de emociones positivas.
- Mayor uso de palabras de entendimiento (pensar, considerar, etc.) y de palabras de discrepancia (debería, podría, etc.)

- Mayor uso de palabras relacionadas con procesos biológicos y con la familia ingerir.
- Menor uso de palabras de asentimiento (de acuerdo, ok, vale, etc.).

2. Neuroticismo:

- Mayor uso de adverbios.
- Menor uso de palabras relacionadas con el trabajo.
- Menor uso de interjecciones
- Menor uso de preguntas (interrogativos)

● Ansiedad.

- Mayor uso de pronombres, pronombres personales y 1ª persona del singular.
- Mayor uso de verbos.
- Menor uso de palabras de tentativa (tal vez, supongo, etc.)
- Menor uso de preguntas (interrogativos)

● Ira.

- Menor uso de palabras de función.
- Menor uso de la 2ª persona del singular.
- Menor uso del tiempo presente y de verbos en 2ª persona (tanto en singular como en plural)
- Menor uso de negaciones (no, nunca, jamás, etc.)
- Menor uso de cuantificadores (pocos, muchos, etc.)
- Menor uso de palabras relacionadas con procesos sociales.
- Menor uso de palabras relacionadas con procesos afectivos y con emociones positivas.
- Mayor uso de palabras inclusivas (y, con, incluyendo, etc.)
- Menor uso de palabras exclusivas (pero, sin, aunque, etc.)
- Menor uso de las palabras relacionadas con ver y oír.
- Mayor uso de palabras relacionadas con la familia comer.
- Menor uso de palabras de asentimiento (de acuerdo, ok, vale, etc.)
- Menor uso de preguntas (interrogativos)

● Asco.

- Mayor uso de adverbios.
- Menor uso de verbos en 2ª persona del singular.
- Menor uso de palabras exclusivas (pero, sin, aunque, etc.)

3. Insensibilidad emocional.

- Menor uso de palabras en su discurso.
- Menor uso de palabras de más de seis letras.
- Menor uso de la 2ª persona del singular.
- Menor uso del tiempo en pasado, de verbos en 2ª persona del singular y de verbos en 3ª persona del plural.
- Menor uso de palabras relacionadas con procesos sociales
- Menor uso de palabras relacionadas con la familia y con los seres humanos en general.
- Menor uso de palabras relacionadas con procesos afectivos, tanto con emociones positivas como con negativas)

- Mayor uso de palabras de entendimiento (pensar, considerar, etc.) y de palabras de discrepancia (debería, podría, etc.)
- Menor uso de palabras relacionadas con procesos perceptivos.
- Menor uso de preguntas (interrogantes)

4. **Impulsividad.**

- Menor uso de palabras en el discurso.
- Menor uso de palabras de más de seis letras.
- Menor uso de palabras de función.
- Mayor uso de verbos.
- Mayor uso de palabras relacionadas con los amigos.
- Menor uso de palabras relacionadas con procesos cognitivos, con palabras de discrepancia (debería, podría, etc.) y de tentativa (tal vez, supongo, etc.)
- Menor uso de palabras de asentimiento (de acuerdo, ok, vale, etc.) e interjecciones

Sin embargo, no se localizan indicadores léxicos sobre el rasgo de ENCUIST de **necesidad de cognición** debido a que se centra en rasgos más importantes para el entorno policial, centrándose en la extroversión (apertura y búsqueda de sensaciones), neuroticismo (ira, asco, ansiedad), insensibilidad emocional e impulsividad.

De lo descrito en este apartado sobre el modelo ENCUIST se obtienen requisitos acerca del uso de las palabras que pueden aparecer en el perfil, su frecuencia y tipología. Es decir, que tipo de palabras habrá que tener en cuenta cuando se aplique PLN.

Los rasgos de la personalidad aparecen directamente relacionados con los perfiles de ENCUIST.

Sin embargo, las palabras aparecen contabilizadas cualitativamente, no cuantitativamente, algo necesario para poder hacer la asignación directa a un perfil. Este aspecto conlleva a que el sistema que se diseñe inicialmente deberá realizar esta cuantificación a través de un sistema que clasifique a los individuos que han sido evaluados mediante un test ENCUIST con sus indicadores extraídos de RRSS.

2.1.3.2 **Indicadores verbales para los rasgos del modelo PEN de Eysenck.**

Como se ha expuesto anteriormente, hay relación entre el modelo PEN y el modelo ENCUIST, habiendo una relación directa entre unos rasgos y otros [2][3]. Para tomar más requisitos léxicos se ha utilizado también bibliografía relativa al estudio del modelo PEN y los indicadores verbales que se obtienen de relacionar resultados de los test de rasgos con textos de los sujetos estudiados. El artículo de la SEI [10] utiliza el EPQ de Eysenck para establecer los siguientes indicadores.

● **Extrovertidos**

- Mayor número de referencias a sí mismo y a otras personas (nombres propios, pronombres).
- Mayor número de palabras asociadas a acciones (verbos).
- Contiene más adverbios, modificadores y preposiciones.
- Frases más largas (mayor número de palabras), con pausas más cortas.
- Mayor número de elipses y guiones.
- Mayor número de letras mayúsculas.
- Utilización de signos de exclamación.
- Mayores errores tipográficos.
- Emociones positivas (interjecciones de risas “ja”, palabras de emociones de alegría).
- NO emociones negativas.
- Utilización de palabras más sociales (“nosotros”, “amigos”, “colegas”).
- Menores referencias a problemas.

-
- Términos que indiquen placer y acuerdo, menor uso de negaciones.
 - **Neuroticismo.**
 - Gran número de referencias a sí mismas (mayor uso de la primera persona del singular “yo”).
 - Lenguaje intenso: excesivo uso de adjetivos y adverbios.
 - Lenguaje muy repetitivo.
 - Expresión de emociones negativas, sobre todo, pero también de emociones positivas (aunque de forma significativamente menor).
 - Mayor ansiedad durante la comunicación (“preocupación”, “ansiedad”, “estar mal”, “estar rayado”).
 - Términos que indiquen placer y acuerdo, menor uso de negación.
 - **Psicoticismo.**
 - Pocas referencias a sí mismas y a otros (menor utilización de la primera persona del singular).
 - Lenguaje inusual: uso de palabras poco frecuentes, mayor utilización de construcciones pasivas, vocabulario más variado, colocación diferente de verbos, adverbios y nombres.
 - Evitan la expresión de emociones positivas o negativas, así como de afectividad.
 - Mayor número de referencias temporales y cognitivas.
 - Menor probabilidad de referencias a relaciones íntimas.
 - Mayor número de discrepancias con el interlocutor.
 - Menor ansiedad durante la comunicación (muy pocas referencias a “preocupación”, “ansioso”, “tenso”, “nervioso”).

De lo descrito en este apartado sobre el modelo EYSENCK se pueden extraer requisitos acerca del uso de las palabras que pueden aparecer en el perfil, su frecuencia y tipología. Es decir, que tipo de palabras habrá que tener en cuenta cuando se haga el procesamiento de lenguaje natural.

Sin embargo, las palabras aparecen contabilizadas cualitativamente, no cuantitativamente para poder hacer la asignación directa a un perfil. Al igual que en el apartado anterior, este aspecto conlleva a que el sistema que se diseñe inicialmente deberá realizar esta cuantificación a través de un sistema que clasifique a los individuos que han sido evaluados mediante un test ENCUIST con sus indicadores extraídos de RRSS.

Hay que tener en cuenta que los rasgos de la personalidad son los de PEN de Eysenck con lo que se pueden utilizar la siguiente correspondencia para ENCUIST[2][3]:

Extroversión en Eysenck con Extroversión en ENCUIST

Neuroticismo en Eysenck con Neuroticismo en ENCUIST

Psicoticismo en Eysenck con la Insensibilidad emocional en ENCUIST

2.1.3.3 Otros posibles indicadores no verbales.

A continuación, se detallan una serie de indicadores no verbales y que pueden extraerse de los perfiles de RRSS. Estos indicadores complementan a los verbales:

- La alta **extroversión** correlaciona con un mayor número de contactos en redes sociales [46].
- La **apertura a la experiencia** aparece asociada al uso de RRSS. El extrovertido buscador de sensaciones es de los perfiles que más van a aparecer en RRSS [47].

Personas emocionalmente estables (bajo neuroticismo) harán menor uso de las redes sociales.

- La alta **extroversión** correlaciona con el uso de las RRSS como una extensión social, el alto psicoticismo usa las RRSS como un sustituto social, mientras que el alto neuroticismo se asocia con estos dos diferentes usos sociales [48].
- Las personas prefieren interfaces diseñadas de acuerdo con su personalidad [49].
- El **alto psicoticismo** se relaciona con un mayor uso de distintas herramientas y aplicaciones dentro de las redes sociales[50].

El **alto neuroticismo** se correlaciona con el uso de Internet y las redes sociales como una herramienta para disminuir la soledad y crear una sensación de pertenencia al grupo [50].

- Los usuarios populares y los influencers suelen ser **extrovertidos y estables** [51].
- Personas con **alta amabilidad** tenderán a aparecer más a menudo etiquetadas en fotos[52].
- Personas con alta apertura a nuevas experiencias ofrecerán un mayor patrón de añadir y cambiar fotos [53].
- La **apertura a la experiencia** aparece positivamente relacionada con el número de likes del usuario, la asociación a grupos y los cambios de estado. Las personas **extrovertidas** comparten más a menudo sus sentimientos y vivencias y pertenecerán a grupos [54].
- Personas **extrovertidas** con alto **neuroticismo** utilizan más la mensajería instantánea [55].
- La **extroversión** aparece asociada a un uso habitual de las RRSS [56].
- El alto **neuroticismo** aparece asociado al uso del muro, mientras que el bajo **neuroticismo** se asocia con la publicación de imágenes en detrimento del uso del muro [57].
- La apertura a la experiencia se correlaciona con fotografías de Instagram en las que predomina el verde y que tienen menos brillo, son imágenes más saturadas, de colores más vivos. La **extroversión** se correlaciona con fotografías de Instagram en las que predominan tonos verdes y azules, con menos naranjas y rojos. El **neuroticismo** parece correlacionarse con fotografías de Instagram con mayor brillo [58].

De estos artículos que tratan sobre la personalidad y su reflejo en los perfiles de RRSS se puede extraer requisitos sobre metadatos de interés que se pueden extraer de los perfiles relacionados con los diferentes rasgos de la personalidad. Por lo tanto, es importante recoger la siguiente información:

Número de amigos

Número de likes

Número de fotos

Frecuencia de actualización

Número de grupos o foros a los que pertenece

Solo se van a tomar como requisitos aquellos que no tengan que ver con el estudio de la fotografías de los perfiles, que pueden ser del ámbito de otros trabajos.

2.2 Redes Sociales objetivo

Para la obtención de datos de sujetos en sus perfiles de redes sociales se necesitan básicamente textos que tengan suficiente contenido para inferir su personalidad. De ahí que no todas las plataformas sociales que se utilizan sean aptas para extraer información usable.

El *Interactive Advertising Bureau* (IAB) [20] es el principal organismo a nivel global que representa a la industria de la publicidad online en el mundo. Su objetivo es permitir a las industrias de comercialización y medios prosperar en la economía digital. Qué mejor que la mayor de las asociaciones

de marketing para hablar sobre el uso de las redes sociales por los usuarios, ya que, hoy en día, son su principal plataforma de trabajo.

En España, según el “*Estudio anual de redes sociales*” de IAB Spain [20], los medios sociales más utilizados son por este orden: **WhatsApp, Facebook, Instagram, YouTube y Twitter.**

Aunque otras redes sociales como Spotify, Telegram, Tiktok, LinkedIn o Pinterest también se encuentran entre el top 10.

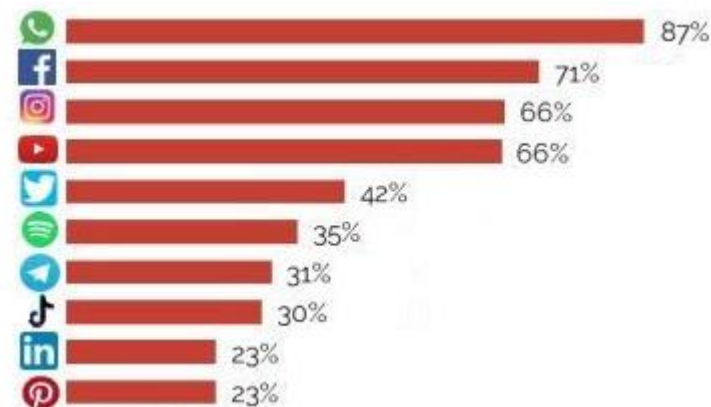


Figura 2-1 Top 10 de las principales redes sociales en España según IAB [20]

En los siguientes puntos se explican las cinco redes sociales más usadas en nuestro país y por qué unas son aptas para el estudio de la personalidad basado en indicadores textuales y otras no.

- **WhatsApp** es la red social más utilizada en España con 30,5 millones de usuarios sólo en nuestro país. Es una aplicación de chat para smartphones que sirve para enviar mensajes de texto y multimedia entre sus usuarios. Su funcionamiento es similar a los programas de mensajería instantánea para ordenador más comunes, aunque enfocado y adaptado al móvil. WhatsApp permite llamadas y videollamadas. La aplicación es utilizada por todos los rangos de edad de la población ya que permite hacer llamadas, enviar y recibir mensajes, documentos, fotos y videos.

Puede haber controversia en si se trata o no de una red social. Tiene características comunes con las redes sociales más convencionales y hace tiempo que dejó de ser una simple herramienta de mensajería. Sin embargo, no es necesario profundizar más en el tema, la organización IAB la toma como red social para su comparativa, que es la que se está utilizando en este apartado para elegir las redes sociales objeto de este estudio.

Conclusión: Sin embargo, a pesar de ser la red social más utilizada y que disponga de mucha información textural en sus comunicaciones, no se puede acceder a los textos de los mensajes de los perfiles objetivos ya que los canales son privados y cifrados entre los extremos de la comunicación.

- **Facebook** es la segunda red social más utilizada entre los españoles, sobre todo entre los mayores de 40 años. En España tiene una penetración enorme, ya que el 87% de la población cuenta con un perfil en Facebook. Entre sus funcionalidades destaca el uso de Facebook Messenger frente a la publicación de contenido o el consumo de contenido multimedia.

Conclusión: Gran cantidad de los contenidos de los perfiles de Facebook son públicos y tienen texto suficiente para su estudio. Al ser una red social de bastante difusión hay altas

probabilidades de que los objetivos tengan perfil. Por lo tanto, Facebook es una red social adecuada para el estudio.

- **Instagram**, en tercera posición, es la red social preferida por los jóvenes españoles, ya que más del 80% de sus usuarios tienen edades comprendidas entre 14 y 25 años. En 2022 ha alcanzado los 24 millones de usuarios y se ha posicionado como la red social con mayor índice de crecimiento en el estudio de este año, con un 20% más de usuarios en comparación con el año pasado. Esta red social fue pensada en un principio para compartir fotos y tiene más funcionalidades como los formatos de vídeo corto llamados “stories” y los “reels”.

Conclusión: El enfoque de esta plataforma es básicamente visual, tiene poco texto, con lo que no es una red apropiada para hacer un estudio de personalidad basándonos en indicadores léxicos. Si los indicadores fueran sobre las imágenes publicadas por los objetivos sería sin duda la red objeto principal del estudio.

- **YouTube** se sitúa en cuarto lugar en frecuencia de uso. Según un estudio elaborado por IAB, su público objetivo es el más variado de todas las redes sociales. Los usuarios se encuentran entre 25 y 49 años, siendo los hombres de clase social media los más activos.

Al igual que con WhatsApp la controversia está servida. Pero, así como con WhatsApp puede haber alguna duda sobre si es o no es una red social, con YouTube no hay ninguna si se estudian las características que deben tener las redes sociales: Viralidad e influencia. YouTube es la red social de video por excelencia.

Conclusión: Al igual que sucede con Instagram esta red no es adecuada por la poca cantidad de textos que posee, sin embargo, si se tratase de buscar la personalidad a través de imágenes habría que tenerla en cuenta.

- **Twitter** es la quinta red social más popular en número de usuarios con unos 7 millones en España. De esta red social cabe destacar el aumento del uso en jóvenes de entre 16 y 24 años, justo al contrario de lo que sucede con Facebook. Los hombres son los usuarios mayoritarios de Twitter tanto en España como en el resto del mundo. En España suponen el 63% frente al 37% de mujeres, mientras que en el mundo la diferencia se acrecienta con un 68% de hombres. Sus usuarios destacan la rapidez para informarse y compartir opiniones sobre temas de actualidad.

A diferencia de otras de las redes sociales más utilizadas, Twitter ofrece un lugar ideal para aquellos usuarios que desean expresar ideas de todo tipo, de forma original y con claridad.

Conclusión: Twitter es ideal para este estudio ya que justamente su intención es la de mostrar la opinión de los usuarios en formato texto.

3 ARQUITECTURA DEL SISTEMA

Gracias al estudio del conocimiento experto se conoce el contexto, la problemática y las necesidades del ámbito del sistema de información a desarrollar, es decir, toda la información necesaria para poder diseñar un sistema que cumpla los requisitos planteados.

Con toda esta información ya se está en disposición de plantear la especificación de requisitos del sistema, análisis de los diferentes módulos que cubrirían esas necesidades y una solución de arquitectura con una propuesta de herramientas interconectadas entre sí para cumplir los requisitos.

La arquitectura planteada no pretende ser única, si no una de las múltiples posibilidades que pueden elaborarse y que den solución a la problemática planteada.

3.1 Análisis del sistema

La Figura 3-1 Análisis de subsistemas muestra los subsistemas y flujos de información que se han considerado necesarios para cumplir los requerimientos del sistema.

- *Subsistema de extracción de perfiles de RRSS*: Se encargará de recoger la información de los perfiles de interés de las RRSS Facebook y Twitter. Enviará la información a un repositorio de *Perfiles*.
- *Subsistema de PLN y análisis de sentimientos*: Procesará la información de los perfiles de RRSS extrayendo los indicadores que se consideren necesarios. También realizará un análisis de sentimientos de los textos para complementar la información. Almacenará los resultados en un repositorio de *Indicadores y Sentimientos*.
- *Subsistema de análisis de rasgos ENCUIST*: Utilizará los indicadores calculados previamente para inferir los rasgos de la personalidad de los sujetos. Almacenará la información en un repositorio de *Rasgos*.
- *Subsistema de generación de informes*: Este subsistema permitirá realizarle preguntas y generar información que permita la toma de decisiones por parte de los usuarios del sistema.
- *Almacenes*: Tal y como se ha indicado existirán tres almacenes, el de *Perfiles*, el de *Indicadores* y el de *Rasgos*.

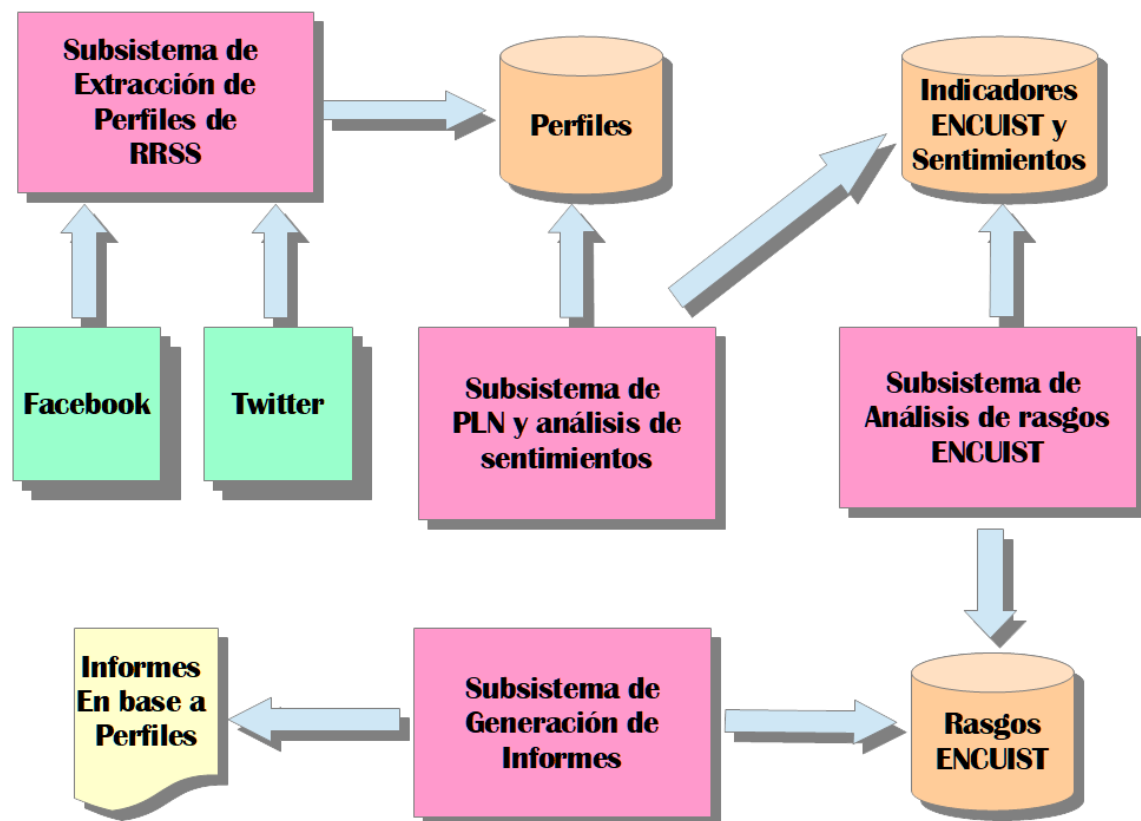


Figura 3-1 Análisis de subsistemas y flujos de información

3.1.1 Definición de requisitos de subsistemas

- *Subsistema de extracción de perfiles de RRSS*
 - El sistema será capaz de descargar perfiles de usuario de las redes sociales Twitter y Facebook.
 - Como parámetro de entrada se indicarán los perfiles que se quieren descargar.
 - Los perfiles serán descargados y se almacenarán en formato textual junto con los metadatos que acompañen al perfil.
 - Los identificadores de los perfiles junto con la metainformación de descarga serán también almacenados acompañando a la información de descarga.
 - Metadatos que deben ser descargados de cada perfil:
 - Nombre del perfil
 - Fecha de descarga
 - Identificador del grupo o del foro del que nos lo estamos descargando
 - Red social de la que se hace la descarga
 - Número de amigos
 - Número de grupos a los que pertenece (en el caso de Facebook)
 - Número de foros a los que pertenece (en el caso de Twitter)
 - Número de fotos descargadas
 - Número de likes que tiene el perfil

- Número de veces que publica
- Frecuencia media de publicación
- *Subsistema de procesamiento de lenguaje natural y análisis de sentimientos*
 - Los diferentes perfiles obtenidos serán procesados para extraer los indicadores marcados por el método ENCUIST. Se realizará un etiquetado morfológico, sintáctico y semántico de los perfiles.
 - Para poder realizar la obtención de los indicadores será necesario realizar un preprocesamiento previo que consistirá en:
 - Tokenización de los textos.
 - Normalización de los textos.
 - Los indicadores ENCUIST que serán necesarios analizar y almacenar de cada perfil serán los siguientes:
 - Número total de palabras en el perfil
 - Media de palabras por frase
 - Palabras mayores de 6 letras
 - Palabras de función: N° total
 - Conjunciones: N° total
 - Inclusivas: N° total
 - Exclusivas: N° total
 - Negación: N° total
 - Negaciones: N° total
 - Adjetivos: N° total
 - Clasificación de las palabras en las categorías:
 - Dimensiones lingüísticas
 - Procesos psicológicos
 - Asuntos personales
 - Lenguaje oral
 - Pronombres personales
 - Primera persona del singular: N° total
 - Segunda persona del singular: N° total
 - Tercera persona del singular: N° total
 - Primera persona del plural: N° total
 - Segunda persona del plural: N° total
 - Tercera persona del plural: N° total
 - Pronombres impersonales: N° total
 - Artículos: N° total
 - Números: N° total

- Preguntas: N° total
- Verbos: N° total
 - Indicativo: N° total
 - Subjuntivo/Condicional: N° total
 - Pasado: N° total
 - Presente: N° total
 - Futuro: N° total
 - Primera persona singular: N° total
 - Segunda persona singular: N° total
 - Tercera persona singular: N° total
 - Primera persona plural: N° total
 - Segunda persona plural: N° total
 - Tercera persona plural: N° total
- Se realizará un análisis de sentimiento sobre los perfiles.
 - Sentimientos: positivo/negativo/neutro
- Toda la información obtenida en este sistema se almacenará en una estructura de almacenamiento llamada *Indicadores ENCUIST* y *sentimientos* que irá acumulando toda la información obtenida del procesamiento de los perfiles junto con el identificador de cada perfil.
- La clasificación del sentimiento extraído del perfil también se almacenará en una estructura llamada *Indicadores ENCUIST* y *sentimientos*.

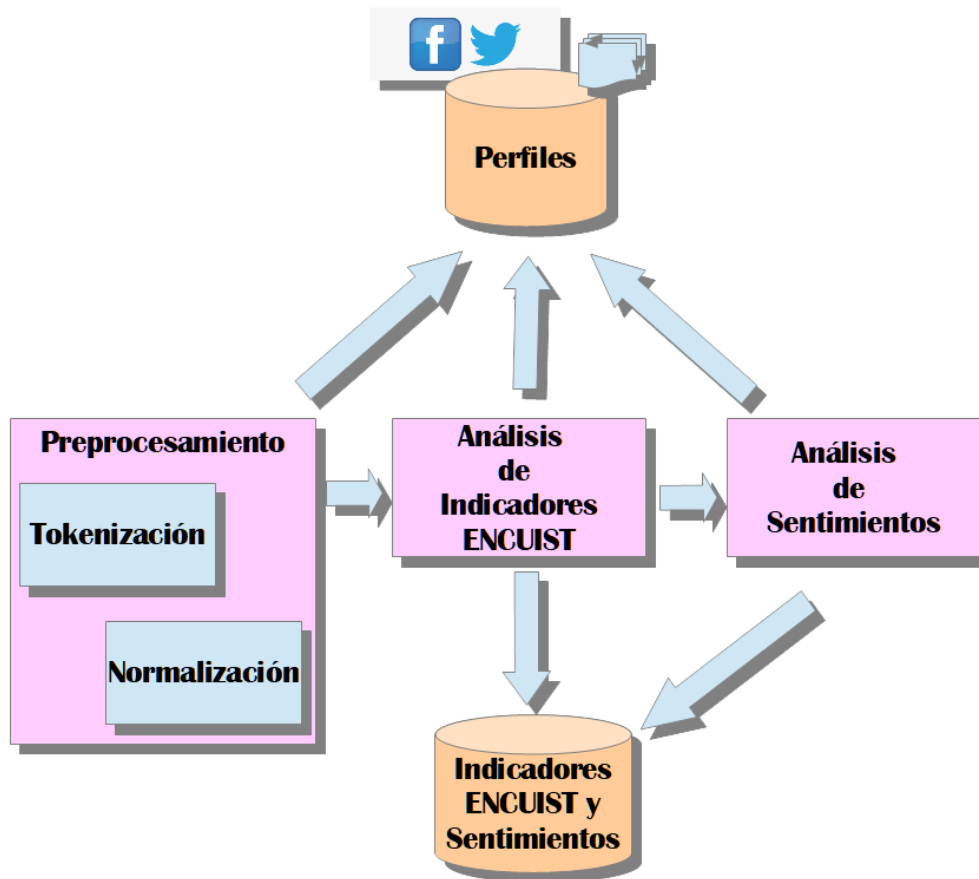


Figura 3-2 Subsistema de PLN y análisis de sentimientos

- *Subsistema de análisis de rasgos ENCUIST.*

- Inicialmente es necesario realizar una cuantificación de los indicadores ENCUIST para poder realizar la automatización del análisis de perfiles (como se ha comentado en apartados anteriores, todos los documentos estudiados sobre los indicadores léxicos de ENCUIST están medidos de forma cualitativa pero no cuantitativa).
- A partir del perfil tokenizado y normalizado almacenado en *Perfiles* el subsistema deberá ser capaz de rellenar de forma automática la estructura de almacenamiento de *Indicadores ENCUIST*.
- Inicialmente este sistema deberá tener un proceso de entrenamiento.

Data set de entrenamiento:

- Serán necesarios *n* individuos que realicen un test ENCUIST. Dichos tests serán evaluados por expertos en psicología y sus resultados almacenados en *Rasgos ENCUIST*.
- De estos individuos se necesitan sus perfiles de Facebook y Twitter. Estos perfiles se procesarán a través del *Subsistema de extracción de perfiles de RRSS* que almacenará los resultados en *Perfiles*.
- Se ejecutará el *Subsistema de procesamiento de lenguaje natural y análisis de sentimientos* generando la información actualizada en *Perfiles* e insertando información en *Indicadores ENCUIST y Sentimientos*.

- Los pares de vectores de entrada y salida para entrenar el sistema estarán almacenados en *Indicadores ENCUIST – Rasgos ENCUIST*, relacionados a través del *idPerfil*.

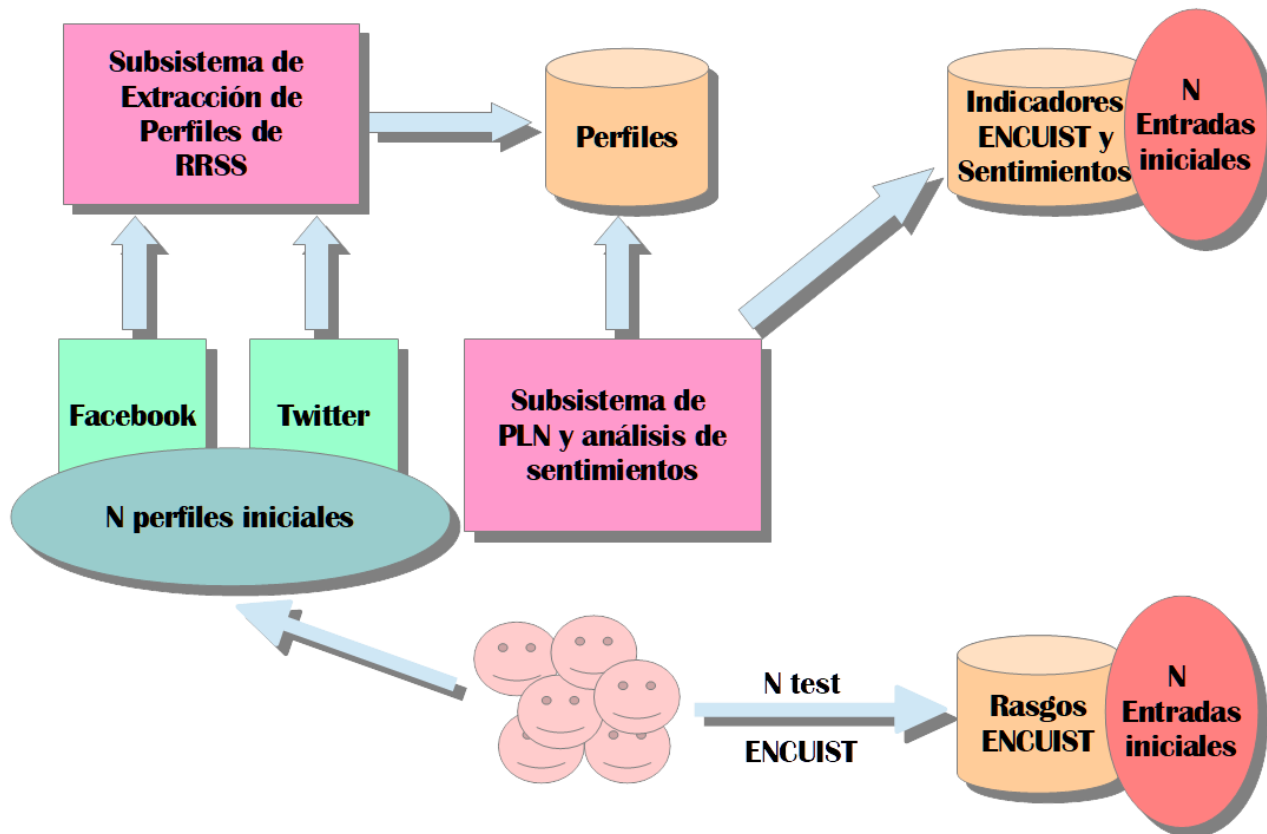


Figura 3-3 Carga de datos iniciales

- Una vez entrenado el modelo debe ser capaz de indicar cuales son los rasgos ENCUIST para los indicadores calculados a partir de un perfil del almacén *Perfiles*.

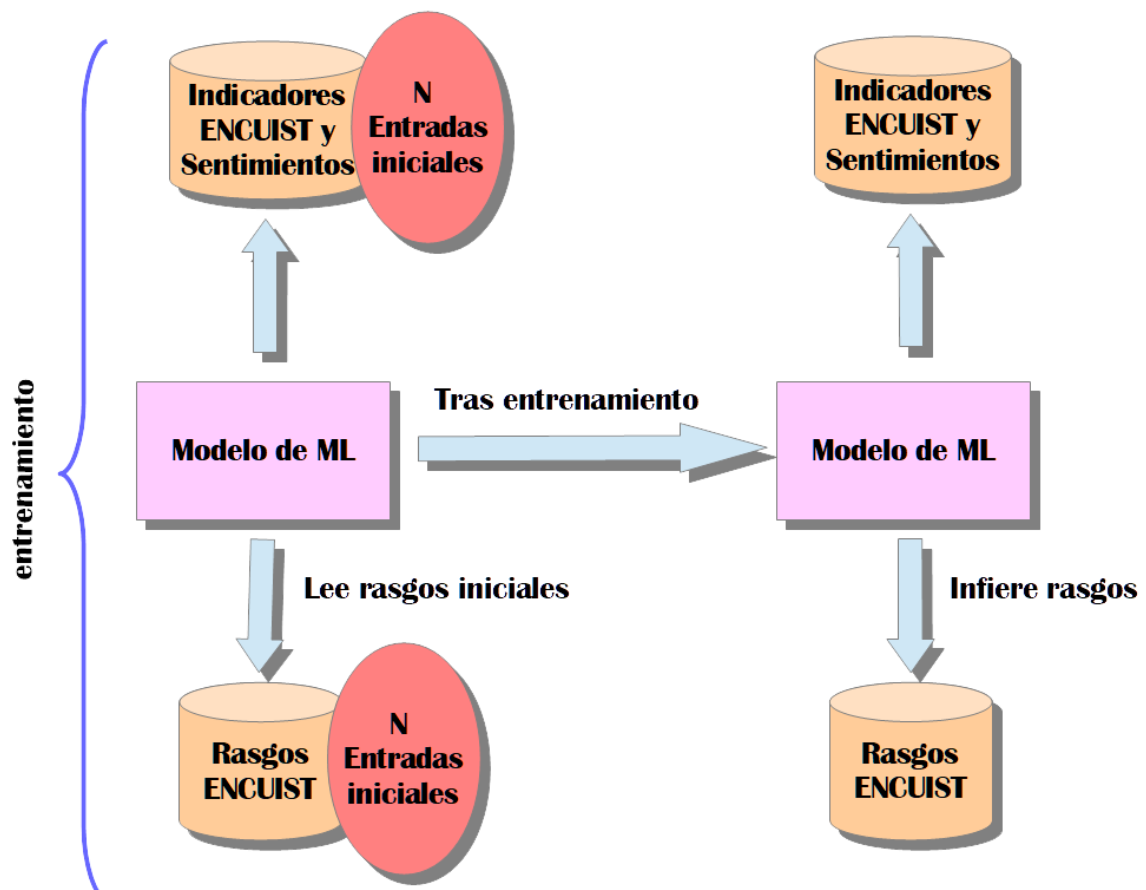


Figura 3-4 Subsistema de análisis de rasgos ENCUIST

- *Almacenamiento*
 - Será necesario almacenar en el sistema:
 - Los perfiles extraídos de RRSS junto con sus metadatos
 - Indicadores ENCUIST y análisis de sentimientos inferidos de cada perfil
 - Rasgos de la personalidad calculados a partir de los indicadores
 - El subsistema se relacionará entre los diferentes elementos a través del *idPerfil* descargado.
 - El subsistema deberá ser **flexible** a la hora de almacenar información ya que el modelo puede cambiar a lo largo de la vida del sistema por diversos motivos:
 - Las RRSS están en continua evolución y puede ser necesario almacenar más información que se considere relevante.
 - El modelo ENCUIST es muy moderno y está siendo objeto de continuo de cambios y evolución. Pueden aparecer más indicadores del modelo que sea necesario reflejar.
 - El almacenamiento debe ser capaz de almacenar toda la **información** necesaria y sus **relaciones** para dar soporte al subsistema de generación de informes.
- *Subsistema de generación de informes*

- Los informes deberán ser dinámicos, flexibles e interactivos, para que el usuario del sistema no tenga que limitarse a los listados predefinidos que pueden no responder a todas las dudas que se le planteen a lo largo de la vida del sistema.
- El sistema estará basado en dashboards capaz de mostrar los principales indicadores del sistema y permitir su análisis por parte del usuario que no debe porqué tener conocimientos técnicos.
- Deberá ser configurable por el usuario para permitirle realizar análisis adecuados.
- Debe ser flexible a la hora de mostrar y configurar la información mostrada.
- Como mínimo permitirá realizar análisis sobre los perfiles individuales o agrupados teniendo en cuenta la metainformación que los acompaña.

4 ESTUDIO DE HERRAMIENTAS TECNOLÓGICAS RELACIONADAS

Para poder dar soluciones técnicas a los diferentes módulos que componen el sistema se va a realizar un recorrido por las diferentes herramientas y tecnologías relacionadas con los diferentes módulos en los que se ha propuesto dividir el sistema.

En cada apartado se explicará cuál es la decisión tomada para el diseño en función de la información estudiada.

4.1 Extracción de información de RRSS

El *web scraping* es el proceso de recopilar datos web estructurados de forma automatizada. El software que se desarrolla se comporta simulando la navegación de un humano y están especializadas en sitios en concreto. En general, la **extracción de datos web** es utilizada por personas y empresas que desean hacer uso de la gran cantidad de datos web disponibles públicamente.

Es extremadamente simple y normalmente sus funciones se dividen en dos partes: un rastreador web (*crawler*) y un raspador web (*scraper*).

Un rastreador web, al que generalmente se le denomina “araña”, es una inteligencia artificial que navega por Internet para indexar y buscar contenido siguiendo enlaces y explorando.

En muchos proyectos, primero se “rastrea” la web o un sitio web específico para descubrir las URL que luego pasa al *scraper* para su descarga.

Los *web scrapers* varían ampliamente en diseño y complejidad, dependiendo del proyecto. Una parte importante de cada *scraper* son los localizadores de datos (o selectores) que se utilizan para encontrar los datos que se desean extraer del archivo HTML.

Para este proyecto será únicamente necesario el diseño de la parte *scraper*, ya que el *crawler* no es necesario porque se sabrá a priori cuáles son los perfiles que se desean descargar.

El *web scraping* consta de varios pasos:

- Realizar una solicitud a las URLs que se desea descargar para obtener el HTML de las páginas.
- Utilizar localizadores para encontrar los datos en el HTML. Por lo general, se aplica XPath, selectores CSS, expresiones regulares o una combinación de todos ellos.
- Una vez obtenidos los datos los guarda en un archivo JSON o CSV o en algún otro formato estructurado.

Existen librerías tanto de código abierto como de pago especializadas en el tipo de información que se necesita descargar, en este caso, Facebook y Twitter con librerías de Python. El principal problema de mantener los *scrapers* de estas dos RRSS es tener que mantener continuamente su código ya que estas redes cambian constantemente sus APIs.

Para la programación de *scrapers* especializados en Python utilizando código abierto es especialmente conocido el *Framework Selenium*[59]. El *web scraping* con Python y Selenium es una de las combinaciones más utilizadas para una recopilación de datos de RRSS e internet en general.

Selenium

Se trata de un framework de software libre que fue diseñado para realizar test automatizados a aplicaciones web[59]. Es decir, inicialmente se desarrolló para probar páginas webs y apps, pero el WebDriver de Selenium también puede usarse con Python para realizar *scraping*. Selenium es un

proyecto que encapsula una variedad de herramientas y bibliotecas que permiten la automatización de la navegación web.

Selenium WebDriver permite interactuar directamente con los navegadores a través de scripts. Java, PHP, C#, Python, Ruby, Perl y Javascript son algunos de los lenguajes de programación que admite. Los navegadores compatibles con Selenium WebDriver incluyen Mozilla Firefox, Google Chrome versión 12.0.712.0 y superior, Internet Explorer, Safari, Opera versión 11.5 y superior y HtmlUnit versión 2.9 y superior. En cuanto a los sistemas operativos, Selenium WebDriver es compatible con Windows, Linux, Mac OS y Solaris.

Selenium WebDriver también se conoce como Selenium 2 y se compone de 4 componentes principales:

- **Selenium Client Libraries:** Estas librerías actúan como intérpretes de los scripts escritos entre cualquiera de un amplio conjunto de lenguajes de programación y Selenium. De esa manera, Selenium puede seguir las instrucciones dadas independientemente del lenguaje de programación que se utilice para programar los scripts. Los lenguajes que traducen estas librerías son entre otros: Java, Javascript, Python, C#, Ruby o PHP.
- **Cliente JSON Wire Protocol OverHTTP:** Se puede interactuar directamente con todos los principales navegadores, como Google Chrome, Mozilla Firefox, Safari, Microsoft Edge, Opera y más. Cada navegador consta de un controlador de navegador que tiene su propio servidor HTTP. JSON Wire Protocol es el responsable de comunicarse con los controladores del navegador a través de su servidor HTTP. Obtiene la información de las bibliotecas de clientes de Selenium y luego la transmite al controlador del navegador respectivo.
- **Browser Drivers:** Cada navegador tiene un controlador que se encarga de controlar las acciones realizadas dentro de ese navegador. Después de que JSON Wire Protocol transmite la información a un browser driver, éste controla el navegador para ejecutar los scripts de Selenium automáticamente y envía la respuesta en protocolo HTTP a través de un servidor HTTP. Los siguientes son los controladores de navegador para los principales navegadores web: Chrome Driver, FirefoxDriver, OperaDriver, SafariDriver, Ruby Selenium Testing, EdgeDriver.
- **Browsers:** Permite la comunicación entre otros con Google Chrome, Mozilla Firefox, Safari, Opera, Microsoft Edge.

Las herramientas utilizadas para descargar perfiles de Facebook se denominan Facebook scraper y las de Twitter, Twitter *scraper*.

Facebook scraper

Se pueden encontrar múltiples desarrollos gratuitos ya implementados para descargar datos públicos de Facebook. En cualquier caso, es necesario recordar que las APIs de las RRSS cambian constantemente y hay que reajustar los *scrapers*.

Una de los Facebook scraper más conocidos es el que se puede encontrar en <https://pypi.org/project/facebook-scraper/> o bien directo en el perfil de GitHub de su creador: <https://github.com/kevinzg/facebook-scraper>.

La información que FB devuelve depende muchas veces de estar logeado, por lo que se recomienda siempre estarlo.

Existen herramientas de pago que liberan de la tarea de estar manteniendo código y que se pagan por uso como:

- Brightdata Twitter scraper: <https://brightdata.es/products/data-collector/website/facebook>

-
- Apify Twitter scraper: <https://apify.com/pocesar/facebook-pages-scraper>
 - Phantombuster: <https://phantombuster.com/>
 - Octoparse: <https://www.octoparse.es/product>

Twitter scraper

Al igual que para FB, Twitter tiene múltiples implementaciones Opensource de scrapers. En Github se puede encontrar una de las herramientas más conocidas desarrolladas en Python <https://github.com/n0madic/twitter-scraper>.

También existen herramientas, muchas de ellas, que se pagan por uso y liberan de la tarea de estar manteniendo código, algunos ejemplos son las mismas que se han propuesto para Facebook ya que tienen sus propias implementaciones para Twitter:

- Brighdata Twitter scraper: <https://brighdata.es/products/data-collector/website/twitter>
- Apify Twitter scraper: <https://apify.com/quacker/twitter-scraper>.
- Phantombuster: <https://phantombuster.com/>
- Octoparse: <https://www.octoparse.es/product>

Para Twitter en concreto existe una opción para el ámbito académico: Twitter Research Academic Access (<https://developer.twitter.com/en/products/twitter-api/academic-research>). Permite la descarga de miles de tweets mensuales siempre que el perfil dado de alta pertenezca al ámbito universitario o de investigación y tenga un objetivo definido.

Una vez estudiadas los métodos para extraer información de las RRSS no se ve necesario realizar más recopilación de datos sobre posibles *Frameworks* o herramientas para realizar *Facebook scraping* o *Twitter scraping* porque los desarrollos actuales con Selenium copan todo el mercado especializado en tareas de *scraping*.

Por lo tanto, se elige para el diseño la utilización de Selenium.

La dificultad estará en mantener el código que se desarrolle, debido a la actualización de las APIs de las RRSS, especialmente, FB. Sin embargo, esta decisión supondrá que el desarrollo sólo costará el esfuerzo del mantenimiento. Aprender a utilizar Selenium es muy sencillo, y, además, al tratarse de código abierto sobre Python, existen mucha documentación en la red.

4.2 Machine learning

Machine Learning (ML) es un método de análisis de datos que automatiza la construcción de modelos analíticos. Es una rama de la inteligencia artificial basada en la idea de que los sistemas pueden aprender de datos, identificar patrones y tomar decisiones con mínima intervención humana[21] .

Es un componente importante del incipiente campo de la ciencia del dato. Con la ayuda de métodos estadísticos, los algoritmos se entrenan para hacer clasificaciones o predicciones, y descubrir nueva información. Esta información facilita la toma de decisiones dentro de los sistemas y de las organizaciones. Las aplicaciones de ML mejoran con el uso y se vuelven más precisas a medida que tienen acceso a más datos.

Actualmente el ML está en pleno auge al igual que la minería de datos y análisis Bayesiano. Esto es debido a los volúmenes y variedades crecientes de datos disponibles, al abaratamiento del procesamiento que cada vez es más potente, y a las diferentes tecnologías de almacenamiento que permite tener grandes de volúmenes de datos accesibles. Todos estos factores significan que es posible producir modelos de manera rápida para analizar datos más grandes y complejos, produciendo resultados más rápidos y precisos [22].

4.2.1 Modelos de aprendizaje de Machine Learning

Existen tres modelos de aprendizaje de ML, la elección de uno u otro dependerá de la naturaleza de los datos y el resultado que se espera obtener. Estos modelos son: supervisado, no supervisado o de refuerzo. En cada modelo se puede utilizar una o varias técnicas algorítmicas. Los algoritmos pueden utilizarse uno a la vez o combinarse para lograr la mayor precisión posible cuando se trata de datos complejos y más impredecibles.

- **Aprendizaje supervisado:** La máquina aprende mediante ejemplos, los cuales están etiquetados como entrada y salida esperada para esa entrada. Mediante un algoritmo, el sistema compila los datos de los entrenamientos a lo largo del tiempo y comienza a determinar similitudes correlativas, diferencias y otros puntos de lógica. La aplicación Waze, para el cálculo de rutas de tráfico, se basa en el uso de algoritmos supervisados.
- **Aprendizaje no supervisado:** Se proporcionan datos de entrada no etiquetados con la respuesta esperada. Muchas veces se trata de datos no estructurados, y el sistema comienza a identificar patrones y correlaciones. Los algoritmos no supervisados empiezan a categorizar la información en base a la propia experiencia del algoritmo, mejorando la precisión con el uso, siendo la experiencia para el algoritmo la cantidad de datos procesados. Muchos sistemas de ciberseguridad se basan en este tipo de algoritmos.
- **Aprendizaje por refuerzo:** Se le proporcionan al algoritmo un conjunto de acciones permitidas, reglas y estados finales potenciales. En los modelos de aprendizaje por refuerzo la recompensa es numérica y se programa dentro del algoritmo como algo que el sistema busca recopilar. Por ejemplo, este tipo de algoritmos es utilizado en negociación bursátil de alto riesgo.

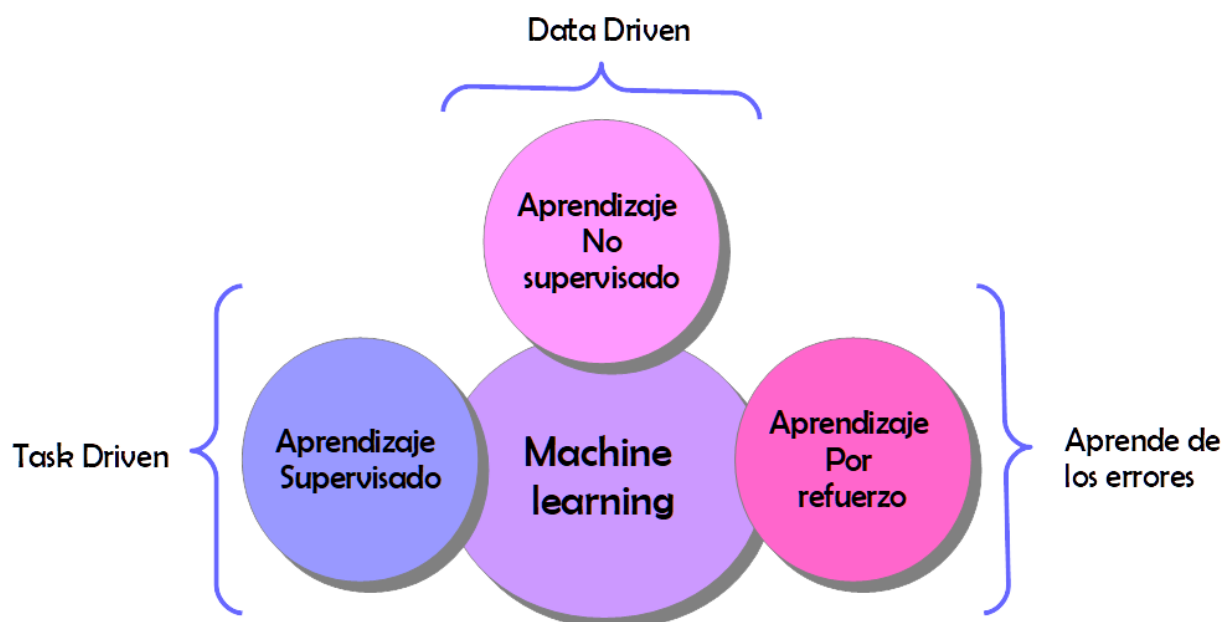


Figura 4-1 Tipos de Machine Learning

4.2.2 Etapas del Machine learning

En el desarrollo de un modelo de Machine Learning hay cuatro etapas y normalmente será *un data scientist* o científico de datos el que gestiona y supervisa el proceso de desarrollo de dicho modelo [22].

1. Inicialmente es necesario seleccionar y preparar un conjunto de datos de entrenamiento. Estos serán los datos que se utilizarán inicialmente para que el modelo aprenda a resolver el problema para el que ha sido diseñado.

En el aprendizaje supervisado los datos estarán etiquetados, como se ha explicado anteriormente, mientras que en el no supervisado no lo estarán y será el modelo el que deba identificar las características que recurrentemente le llegan como entrada.

Es fundamental la preparación de los datos de entrada, su organización y limpieza para que el entrenamiento sea exitoso y no esté sesgado pudiendo fallar en predicciones futuras.

2. El segundo paso será la selección de un algoritmo que será el corazón del modelo. Dependiendo de los datos (tipo y volumen) y del problema a resolver, se elegirá un algoritmo determinado.
3. A continuación, se procederá al entrenamiento del algoritmo, lo que se trata de un proceso de repetición. Se ejecuta el algoritmo con los datos y un conjunto de variables necesarias para la ejecución. Los resultados serán comparados con los que se deberían haber producido.

El algoritmo entrenado es el modelo de Machine Learning.

4. Finalmente, se utilizará el modelo sobre nuevos datos. El procesamiento de nueva información provocará su mejora continua.

4.2.3 Tipos de problemas a resolver con Machine learning

Una vez que se tiene claro en qué consiste el ML es necesario definir los tipos de problemas que puede resolver para ser capaces de identificarlos, aplicando así la técnica y el algoritmo más adecuado para su resolución. Los problemas más típicos que se pueden presentar son: clasificación, regresión y clusterización.

1. **Problemas de regresión:** La salida de estos algoritmos es **cuantitativa**, y depende directamente de la entrada. El análisis de regresión es un proceso estadístico que permite analizar la relación que existe entre dos o más variables, siendo una de ellas dependiente al resto de variables. Es decir, un análisis regresivo hace posible comprender cómo las variables independientes afectan directamente a otra variable que dependa de ellas. Los análisis de regresión se emplean con gran frecuencia en el mundo corporativo, por ejemplo, para predecir la eficacia de un cambio realizado en un proceso comercial. De esta manera las empresas pueden comprender mejor cuáles son los elementos que tienen un mayor impacto en los resultados, cuales afectan a otros elementos de la compañía o cuales se pueden ignorar.

Por lo tanto, es necesario tener claro cuáles son las diferentes variables que intervienen en este tipo de problemas:

- Las variables dependientes: Las que se buscan estudiar utilizando la regresión para comprender cómo se adapta al modificar las variables independientes.
- Las variables independientes: aquellos factores que influyen y afectan directamente a las variables dependientes que se quieren estudiar.

2. **Problemas de clasificación:** El resultado en estos algoritmos es caracterizado por ser **cualitativo**, denominándose en algunos contextos como variables categóricas. Al hecho de predecir una respuesta cualitativa de una observación se le llama clasificar dicha

observación, es decir, predecir la categoría o clase de dicha observación. Para clasificar de forma automática se crea un **modelo predictivo**, al que entregando las variables de entrada genere como salida las etiquetas de clase correspondientes, siendo estas clases un conjunto discreto. Por ejemplo, se puede predecir el rango de categorías al que pertenece un cliente o si una imagen es de comida o de un animal. Los modelos de clasificación requieren grandes conjuntos de datos. Más datos significa más precisión. Por lo tanto, el conjunto de datos de entrenamiento debe ser lo suficientemente grande y de calidad. Los conjuntos de datos de entrenamiento deben parecerse lo suficiente al problema y tener tantos ejemplos de cada clase como sea posible.

3. **Problemas de clusterización:** Los algoritmos de clustering o agrupamiento sirven para encontrar subgrupos o clústeres en los datos. Cuando se agrupan datos lo que se quiere es encontrar particiones que dividan los datos en grupos distintos pero homogéneos. Es decir, los grupos deben ser lo más distintos posibles entre ellos pero que las muestras dentro de cada grupo sean similares. Por lo tanto, inicialmente hay que definir qué se entiende como muestras similares y distintas.

La clasificación y la clusterización tienen ciertas similitudes, pero la diferencia está en el hecho de que la clasificación se sirve de unas clases predefinidas en las que se asignan los objetos, mientras que la clusterización identifica similitudes entre objetos, que agrupa según esas características en común y que les diferencian de los otros grupos de objetos. Por lo tanto, en la clasificación las clases están predeterminadas mientras que en la clusterización los clústeres no lo están, será el sistema el que los identifique en base a las características comunes entre las entradas.

4.2.4 Algoritmos más comunes del Machine Learning

Existen muchos diversos algoritmos de ML. A continuación, se va a realizar una revisión de los más utilizados, haciendo una separación entre los más comunes para modelos supervisados, para los no supervisados y para los de aprendizaje por refuerzo.

Modelos supervisados

1. **Árboles de decisión:** Es un algoritmo de aprendizaje supervisado que gráficamente se puede representar con una estructura de árbol jerárquica, que consta de un nodo raíz, ramas, nodos internos y nodos hoja. Se puede entender como un diagrama de flujo donde cada rama supone una decisión, cada nodo interno es una característica o atributo y cada hoja un resultado. Este tipo de modelos ofrecen gran fiabilidad, estabilidad y facilidad de interpretación. Permiten la construcción de modelo predictivos de clasificación o de regresión.

Los árboles de decisión son **algoritmos fácilmente entendibles** y que funcionan bien en modelos analíticos basados en clasificación o regresión para obtener resultados a un problema.

El aprendizaje del árbol de decisiones emplea una estrategia de divide y vencerás mediante la realización de una búsqueda codiciosa (eligiendo la solución óptima en cada decisión esperando que la solución final sea también la mejor) para identificar los puntos de división óptimos dentro de un árbol. Este proceso de división se repite de forma recursiva de arriba hacia abajo hasta que todos o la mayoría de los registros se hayan clasificado bajo etiquetas de clase específicas.

Que todos los puntos de datos se clasifiquen o no como conjuntos homogéneos depende en gran medida de la complejidad del árbol de decisión. Los árboles más pequeños son más

fáciles de obtener nodos hoja puros, es decir, puntos de datos en una sola clase. Sin embargo, a medida que un árbol crece en tamaño, se vuelve cada vez más difícil mantener esta pureza y, por lo general, da como resultado que haya muy pocos datos dentro de un subárbol determinado. Cuando esto ocurre, se conoce como fragmentación de datos y, a menudo, puede resultar en sobreajustes (mala generalización o pocos datos de entrada). Como resultado, los árboles de decisión tienen preferencia por los árboles pequeños, lo cual es consistente con el principio de la Navaja de Occam, los árboles de decisión deben agregar complejidad solo si es necesario, ya que la explicación más simple suele ser la mejor. Para reducir la complejidad y evitar el sobreajuste, generalmente se emplea la poda. Este es un proceso que elimina las ramas que se dividen en características con poca importancia.

Otra forma en que los árboles de decisión pueden mantener su precisión es mediante la formación de un conjunto a través de un algoritmo de bosque aleatorio o *random forest*. Este clasificador predice resultados más precisos, particularmente cuando los árboles individuales no están correlacionados entre sí.

2. **Random Forest:** Aunque se trata de un algoritmo basado en árboles de decisión, por su importancia y uso extendido merece un tratamiento individualizado.

Es uno de los modelos más utilizado tanto para problemas de clasificación como de regresión lo que le hace ser muy versátil. Este método mantiene la precisión cuando falta una gran proporción de los datos.

Así como todos los modelos, un árbol de decisión también sufre de los problemas de sesgo y varianza. Es decir, ‘cuánto en promedio son los valores inferidos diferentes de los valores reales’ (sesgo) y ‘como de diferentes serán las predicciones de un modelo en un mismo punto si se tomaran muestras diferentes de la misma población’ (varianza). Árboles pequeños darán baja varianza y alto sesgo. Mientras más se complique el árbol el sesgo irá disminuyendo, sin embargo, cuando mayor sea el árbol se irá sobreajustando y el modelo empezará a sufrir una varianza mayor. El modelo óptimo deberá tener un equilibrio entre los dos errores. Los modelos ensambladores, como *random forest* están desarrollados para buscar este equilibrio.

En este modelo se generan diferentes subconjuntos de datos de entrada, se utilizan (se ensamblan) árboles pequeños con esos conjuntos de datos y finalmente se combina la salida. Ninguno de los árboles que componen el bosque podrá ver el conjunto de datos de entrada en su totalidad. Cada árbol de decisión da una clasificación y finalmente la decisión con mayor “votos” es la predicción del algoritmo.

3. **Clasificación de Naïve Bayes:** Estos algoritmos se basan en una técnica de clasificación estadística llamada teorema de Bayes. Naïve en castellano significa inocente. En ellos se asume que las variables predictoras son independientes entre sí. En otras palabras, que la presencia de una cierta característica en un conjunto de datos no está en absoluto relacionada con la presencia de cualquier otra característica. Son algoritmos muy sencillos de utilizar y funcionan muy bien con grandes cantidades de datos.

Lo que hacen es calcular la probabilidad de que ocurra un evento futuro dadas algunas probabilidades de ocurrencia de eventos anteriores.

4. **Algoritmos de regresión lineal:** Este tipo de algoritmo permite comprender las relaciones que existen entre los datos. La regresión lineal se utiliza para predecir el valor de una variable dependiente en función del valor de una variable independiente. Por ejemplo, se utilizan en problemas como el de predecir el consumo energético de diferentes modelos de coches en función de su peso y su potencia.

Las *variables independientes o características* son variables que se manipulan para determinar el valor de una variable dependiente. La *variable dependiente u objetivo*, depende de los valores de la variable independiente. En pocas palabras, es la característica que se está tratando de predecir. Lo más importante en estos algoritmos es la selección de variables independientes que puedan ayudar a predecir el valor de la variable dependiente. Un número elevado de variables independientes no quiere decir necesariamente una mejor salida del algoritmo, a veces da lugar a sobreajustes.

Este tipo de algoritmos se suele utilizar para predecir la cantidad de visitas de una tienda online o la previsión de consumo de un determinado producto.

5. **La regresión logística:** son algoritmos similares a los de regresión lineal, pero con variables binarias o dicotómicas. Toma valores de un determinado par como si/no, hombre/mujer, etc. Es fácil de implementar y se puede usar como línea de base para cualquier problema de clasificación binaria. Describe y estima la relación entre una variable binaria dependiente y las variables independientes. Como todos los análisis de regresión, la regresión logística es un análisis predictivo.

Un caso de uso puede ser la predicción de una determinada característica en un producto, o que un pasajero sobreviva o no en un accidente de avión dependiendo del asiento que ocupe en un determinado modelo de avión.

A pesar de que se trata de un modelo antiguo, la regresión logística es ampliamente utilizada porque es muy eficiente y no requiere demasiados recursos computacionales para su ejecución.

Por lo tanto, parece fácil ver la diferencia entre una regresión logística y una lineal, la primera tiene una salida discreta mientras que la segunda es continua.

6. **Algoritmos de regresión de soporte vectorial o máquinas de soporte vectorial (SVM):** El objetivo de estos algoritmos es encontrar un hiperplano (divide el espacio en dos mitades) que separe de la mejor forma posible dos clases diferentes de datos.

El algoritmo solo puede encontrar este hiperplano en problemas que permiten separación lineal. Los vectores que definen el borde de esta separación son los vectores de soporte, es decir, los datos más cercanos a la línea de separación. Por lo tanto, estos valores son los más importantes a la hora de definir el hiperplano.

Los mejores hiperplanos serán aquellos que maximice la distancia al elemento más cercano. El algoritmo irá clasificando los datos de entrada entre las dos mitades del espacio.

Son muy utilizados en sistemas de visión artificial o en PLN.

Modelos no supervisados

1. **K-Means:** Se trata de un algoritmo de clasificación no supervisada (clusterización) que agrupa objetos en k grupos basándose en sus características. El agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo o clúster. Se suele decir que el centroide es el representante de cada grupo. Se suele usar la distancia cuadrática.

El algoritmo consta de tres pasos:

- Primero se realiza la inicialización del algoritmo: una vez escogido el número de grupos, k , se establecen k centroides en el espacio de los datos, por ejemplo, escogiéndolos aleatoriamente.
- Después se realiza la asignación objetos a los centroides: cada objeto de los datos es asignado a su centroide más cercano.
- Por último, se realiza una actualización centroides: se actualiza la posición del centroide de cada grupo tomando como nuevo centroide la posición del promedio de los objetos pertenecientes a dicho grupo.

Los pasos dos y tres son ejecutados tantas veces como sea necesario hasta que los centroides permanecen inalterados o por debajo de un umbral de distancia.

2. **K-Nearest Neighbours o KNN:** también se trata de un algoritmo no supervisado que puede ser utilizado para resolver tanto problemas de clasificación como de regresión, sin embargo, lo normal es utilizarlo para problemas de clasificación. La “K” significa la cantidad de “puntos vecinos” que tenemos en cuenta en las cercanías para clasificar los “n” grupos que ya se conocen de antemano.

Se asigna una etiqueta de clase sobre la base de un voto mayoritario, es decir, se utiliza la etiqueta que se representa la mayor frecuencia alrededor de un punto de datos determinado. Esto técnicamente se considera "voto por mayoría", es decir, requiere una mayoría superior al 50 %, cuando solo hay dos categorías. En el caso de que haya varias clases, por ejemplo, cuatro categorías sería una etiqueta de clase con un voto superior al 25 %. Se toma, por tanto, se toma el promedio de los k vecinos más cercanos para hacer una predicción sobre la clasificación de la clase que se está tratando de predecir.

Un ejemplo de uso de este algoritmo es el de reconocimiento de texto manuscrito o dígitos que se pueden encontrar en formularios o sobres de correos.

3. **Mean-Shift Clustering:** La agrupación Mean Shift es un algoritmo basado en ventanas deslizantes que intenta encontrar áreas densas de puntos de datos. Es un algoritmo basado en el centroide, lo que significa que el objetivo es localizar los puntos centrales de cada clúster, lo que funciona actualizando a los candidatos para que los puntos centrales sean la media de los puntos dentro de la ventana deslizante.

Estas ventanas candidatas son filtradas en una etapa de post procesamiento para eliminar los duplicados cercanos, formando el conjunto final de puntos centrales y sus correspondientes grupos.

El algoritmo se compone de los siguientes pasos:

- Primero se realiza la inicialización del algoritmo: Se comienza con una ventana circular basada en un punto seleccionado aleatoriamente y con un radio también establecido inicialmente y que se denomina núcleo.
- En las siguientes iteraciones el centro se va desplazando hacia regiones de datos de mayor densidad. El cálculo de la nueva localización del centro se hace conforme a la media.
- La ventana corregida se seguirá moviendo hasta que no haya ninguna dirección en la que se puedan acomodar más puntos dentro del núcleo.

- Los pasos anteriores se realizan con varias ventanas hasta que todos los puntos se encuentran dentro de una ventana. Cuando se superponen varias ventanas se conserva la ventana que tiene mayor cantidad de puntos.
4. **Clúster por jerarquías:** La agrupación es la extracción de agrupaciones naturales de objetos de datos similares. Se basa en el uso técnicas de agrupación para encontrar una jerarquía de agrupaciones, donde esta jerarquía se asemeja a una estructura de árbol, llamada dendrograma. Existen dos categorías para este tipo de clustering:
- **Aglomerante:** Es una aproximación de abajo hacia arriba (bottom-up) donde se dividen los clústers en subclusters y así sucesivamente. Inicialmente se asigna cada muestra simple a un clúster y en cada iteración sucesiva va aglomerando (mezclando) el par de clústers más cercanos satisfaciendo algún criterio de similitud, hasta que todos los elementos pertenecen a un solo clúster. Los clústers generados en los primeros pasos son anidados con los clústers generados en los siguientes pasos.
 - **Divisivo:** Este tipo de clustering se lleva a cabo con un enfoque de arriba hacia abajo (top-down), Se inicia con todos los elementos asignado a un solo clúster y sigue el algoritmo hasta que cada elemento es un clúster individual.

A diferencia del enfoque de abajo hacia arriba donde las decisiones para generar los clústers se basan en los patrones locales sin tomar en cuenta la distribución global, el enfoque de arriba hacia abajo se beneficia de la información completa sobre la distribución global al ir haciendo las particiones.

Modelos de aprendizaje por refuerzo

Los conceptos del AR (aprendizaje por refuerzo) [24] son:

- *Agente:* es el modelo que se quiere entrenar. El que debe aprender a tomar decisiones.
- *Ambiente/Entorno:* dónde se va a ejecutar el agente.
- *Estado:* todos aquellos indicadores que contextualicen el estado.
- *Recompensa:*
 - a. Positiva: Si la acción realizada obtiene una bonificación.
 - b. Negativa: Si la acción realizada obtiene una penalización.
- *Acción:* posibles acciones que puede realizar el agente.
- *Política:* que es la que determina la acción a realizar, dependiendo del estado en el que se encuentra el entorno. Se representa como un mapeado de estados a acciones.

Los pasos en los que se resume el proceso de aprendizaje son los siguientes:

- I. El agente recibe un estado s , y una recompensa r
- II. El agente escoge una acción a del conjunto de acciones disponibles y la envía al entorno
- III. El entorno ejecuta la acción a y su estado pasa, entonces a $s+1$
- IV. El intérprete recibe, entonces una observación $o+1$, interpreta el nuevo estado $s+1$ y calcula la recompensa r_{t+1} correspondiente a la transición $(s; a; s+1)$

La política dará la probabilidad de que el agente tome una acción determinada cuando el entorno se encuentra en un estado determinado. Finalmente, **el agente debe aprender cuál es la política adecuada para obtener la mayor recompensa posible.**

Es un proceso de aprendizaje por condicionamiento similar a la realidad de un ser vivo, donde a través de la observación y la toma de decisiones en un entorno se puede comprobar los efectos que se producen. Es decir, el objetivo en el aprendizaje por refuerzo es que el agente consiga aprender una política óptima para interactuar con el entorno.

1. **Q-Learning** [24]: El algoritmo Q-Learning intenta aprender cuánta recompensa obtendrá a largo plazo para cada pareja de estados y acciones (s, a). A la función mencionada se le denomina de acción-valor. Esta función devuelve la recompensa que el agente recibirá al ejecutar la acción a desde el estado s. Estos elementos se almacenan en una tabla de recompensas: Cada estado podrá tener una serie de acciones permitidas y recibirá una determinada recompensa para cada acción. Sin embargo, la tabla solo tendrá rellenas las recompensas finales (a largo plazo), forma parte del algoritmo rellenar todos los estados intermedios y ver cuales es el camino para llegar a la situación final con mayor recompensa.

Supervisados	No supervisados	Aprendizaje por refuerzo
Árboles de decisión	K-Means	Q-Learning
Random Forest	K-Nearest Neighbours	
Naïve Bayes	Mean-Shift Clustering	
Regresión lineal	Clúster por jerarquías	
Regresión logística		
Máquinas de soporte vectorial		

Tabla 4-1 Algoritmos más importantes del Machine Learning

➤ Conclusión sobre ML

En el establecimiento de requisitos del *Subsistema de Análisis de Rasgos* se vio que era necesario inferir los rasgos a partir de los indicadores. Se parte de un conjunto de entrenamiento que servirá para que el sistema aprenda de qué manera afectan los indicadores a los rasgos.

La tecnología de ML se utilizará para la predicción de cada rasgo en base a los indicadores extraídos de los textos utilizados.

De los requisitos del sistema se sabe que cada rasgo de la personalidad viene evaluado por un valor dentro del rango discreto NULO, BAJO, MEDIO y ALTO. Esto se traduce en que el subsistema deberá hacer una clasificación de los indicadores dentro de una de esas cuatro clases. El modelo a desarrollar será de **clasificación supervisado para cada uno de los rasgos**. La característica de supervisado viene de los datos de entrenamiento de los que se dispondrá inicialmente, donde cada perfil de RRSS vendrá acompañado de su test ENCUIST donde cada rasgo vendrá etiquetado adecuadamente dentro de ese conjunto inicial de valores.

Lo normal cuando se habla de Machine Learning es utilizar varios modelos para ver los resultados de cada uno de ellos y escoger el que más se ajuste a lo esperado. Los algoritmos de ML supervisados de clasificación que se proponen para este diseño son *Random Forest* y *Naïve Bayes*. Estos algoritmos son utilizados típicamente para ir afinando modelos sobre todo al inicio de la etapa de desarrollo. Sin embargo, es necesario destacar que simplemente se trata de una propuesta de diseño.

La clasificación de los resultados, mediante parametrización, debería quedar dentro las cuatro clases comentadas anteriormente y relacionadas con los resultados actuales de los test ENCUIST para cada rasgo (NULO, BAJO, MEDIO, ALTO). Será en fases posteriores a la finalización de este trabajo, en el

desarrollo, donde se podrían explorar otras parametrizaciones menos evidentes para generación de resultados en clases no discretas.

Es importante subrayar que se implementaría un modelo por rasgo debido a que cada uno es independiente entre sí, al menos a priori, así se tratan en los test psicológicos de perfilado directo. La forma de representar esta independencia es mediante un modelo individual para cada uno de los rasgos. De esta forma además se consigue que si en algún momento aparecen en el modelo rasgos nuevos o se elimina o modifica alguno ya existente, se podrá añadir o modificar con bastante libertad. Esquemáticamente, se podría representar de la siguiente manera:

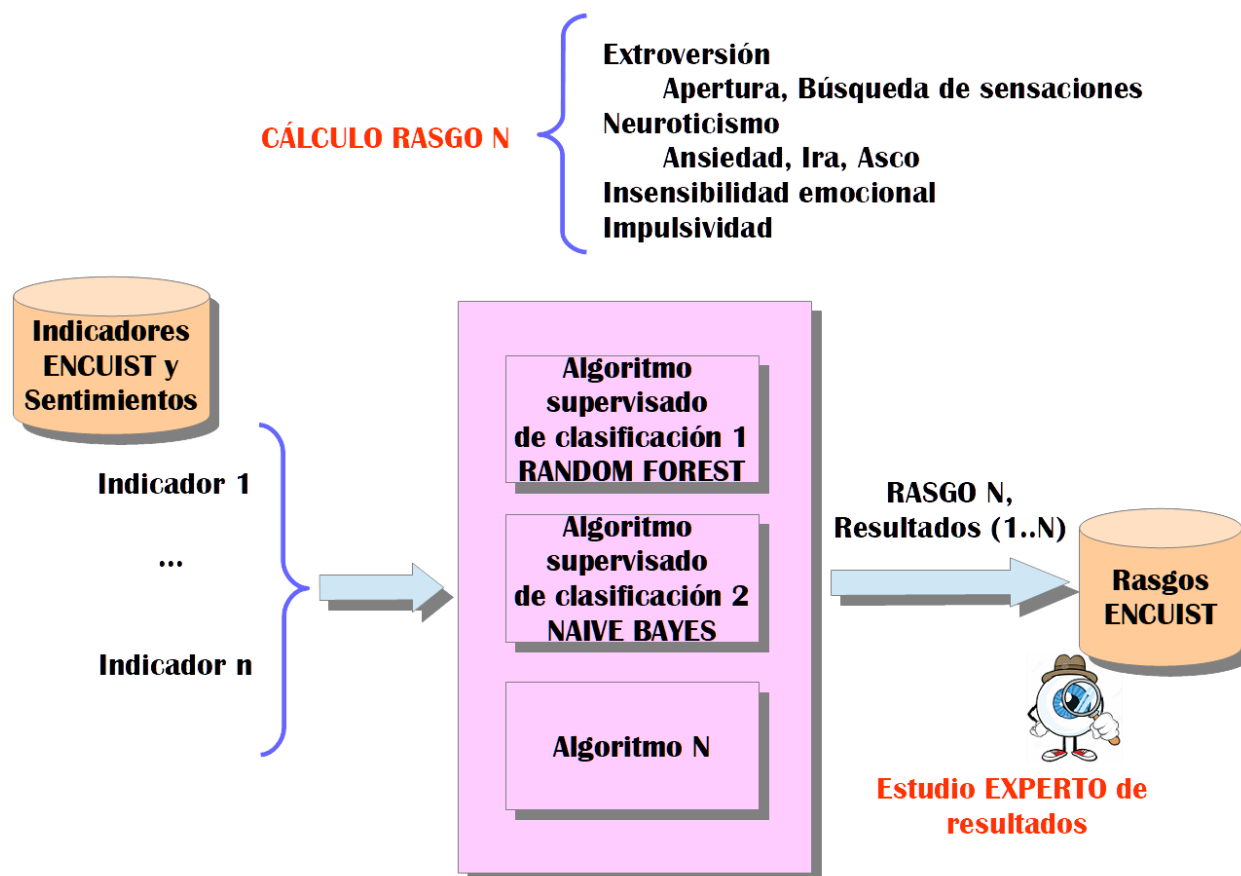


Figura 4-2 Cálculo de cada rasgo

4.3 Procesamiento del lenguaje natural

En la actualidad, el procesamiento del lenguaje natural (PLN) es una rama de la inteligencia artificial que se encarga de estudiar la comunicación de las máquinas con las personas a través de idiomas humanos. Tal y como explica el IIC [8] tratar computacionalmente una lengua implica un proceso de modelización matemática. Existe una figura en este campo que es la de lingüista computacional, que se encarga de modelar un determinado lenguaje para poder ser procesado por un ordenador.

En el sistema de perfilado indirecto que se está diseñando es necesario el PLN para poder establecer los indicadores del lenguaje en base a los cuales se podrán deducir los rasgos. De ahí que sea necesario elegir la mejor manera de conseguir extraer los indicadores mediante PLN.

A continuación, se explica brevemente tres maneras en las que actualmente se realizan modelizaciones de los lenguajes naturales, explicadas desde la más clásica basada en gramáticas, a la

más actual e innovadora que utiliza métodos de *Deep Learning*. Antes de entrar en el apartado de *Deep Learning*, se hará un repaso de las librerías más conocidas de software que permiten hacer desarrollos clásicos de PLN. En *Deep Learning* se hablará en concreto del software que se utiliza en ese tipo de aproximaciones.

4.3.1 Modelos lógicos: Gramáticas

Modelizan la estructura del lenguaje a represar y se basan en las teorías de Chomsky que revolucionaron la psicología de los años 50 afirmando que el lenguaje es el producto de una facultad innata de la mente humana que lo genera con estructuras predefinidas.

4.3.2 Modelos probabilísticos del lenguaje natural: Machine Learning basado en datos

Los lingüistas inicialmente recogen muestras del lenguaje y crean el corpus. Sobre estos ejemplos se aplican fórmulas estadísticas y se calculan las frecuencias y probabilidades de letras, palabras y oraciones. De esta forma se podrán predecir a futuro la aparición de las palabras sin necesidad de utilizar gramáticas.

Los cálculos de aparición se realizan a través de modelos de Machine Learning en base al corpus.

4.3.3 Herramientas clásicas de desarrollo de PLN

Se pueden encontrar múltiples librerías que facilitan la programación al proporcionar funcionalidades comunes a los programadores.

Sin entrar en el campo concreto del Deep Learning que se va a exponer en el siguiente, se detallan a continuación varias de herramientas muy conocidas para el PLN clásico:

1. API GPT3 [9]

GPT3 es una herramienta de OpenAI, empresa fundada por Elon Musk y con un fuerte respaldo de Microsoft. Sus siglas se corresponden con *generative pre-training*. Durante bastante tiempo su API sólo se podía utilizar si la solicitud era aprobada por la compañía, pero desde finales de 2021, siempre que se haga la solicitud desde uno de los países autorizados previamente (España cumple ese requisito), el API se puede utilizar libremente. GPT3 utiliza *Deep Learning* para analizar lenguaje natural y producir contenido textual automáticamente. Puede traducir de un idioma a otro, reconocer entidades nombradas dentro del texto, resumir artículos y componer piezas completas. Este método implica entrenar un modelo con grandes cantidades de datos para mejorar su capacidad de predecir la siguiente palabra más probable en una oración.

Este API inicialmente realiza una tokenización de los textos, es decir, divide los textos en palabras. A continuación, se aplican los modelos.

GPT3 cuenta con varios modelos muy potentes que van organizados desde el más complejo y completo, al más sencillo:

- DAVINCHI: Este modelo es el más completo de todos. Ha sido entrenado con la WIKIPEDIA completa y mucho más contenido. Davinci es el modelo más capaz y puede hacer todo lo que cualquier otro modelo puede hacer, y mucho más, a menudo con menos instrucciones. Davinci es capaz de resolver problemas lógicos, determinar causa y efecto, comprender la intención del texto, producir contenido creativo, explicar los motivos de los personajes y manejar tareas de resumen complejas. Sin embargo, es mucho más lento en sus llamadas al API que el resto de los modelos.
- CURIE: Este modelo es potente y muy rápido a diferencia del anterior. Si bien Davinci es más completo cuando se trata de analizar texto complicado, Curie es

bastante muy eficiente para tareas como la clasificación y el resumen de sentimientos. Curie también es bastante bueno respondiendo preguntas y realizando preguntas y respuestas y como un chatbot de propósito general.

- BABBAGE: puede realizar tareas sencillas como una clasificación simple. También es bastante capaz cuando se trata de la clasificación de búsqueda semántica en qué medida los documentos coinciden con las consultas de búsqueda.
- ADA: suele ser el modelo más rápido y puede realizar tareas como analizar texto, corregir direcciones y ciertos tipos de tareas de clasificación que no requieren demasiados matices. El desempeño de Ada a menudo se puede mejorar al proporcionar más contexto.

Para utilizar el API de GPT3, la descripción completa se puede encontrar en <https://beta.openai.com/docs/api-reference/introduction>. En el sitio web también hay una enumeración de características de los proyectos que utilicen el API y qué están obligados a pasar una revisión por parte de OpenAI. Básicamente aquellos usos de los que se obtenga beneficio económico o utilicen más de 10 usuarios de forma simultánea. También están detallados aquellos usos prohibidos explícitamente por los creadores (uso de lenguaje racista o sexista, contenido de odio, etc.).

Sin embargo, hay que indicar que el uso de GPT3 es de pago, con un máximo de \$120 al mes en el momento de redacción de este documento (octubre 2022).

2. *GPT – NEO y GPT – J* [60]

Como alternativas al uso de GPT3 al ser de pago surgen estas dos librerías. GPT-Neo se lanzó en marzo de 2021 y GPT-J en junio de 2021, como modelos de código abierto, ambos creados por EleutherAI (un colectivo de investigadores que trabajan creando herramientas de IA en código abierto).

EleutherAI usó una biblioteca basada en JAX (una biblioteca de Python que se usa con frecuencia para proyectos de aprendizaje automático).

GPT-Neo tiene 3 versiones: 125 millones de parámetros, 1300 millones de parámetros (equivalente a GPT-3 Babbage) y 2700 millones de parámetros.

GPT-J tiene 6 mil millones de parámetros, lo que lo convierte en el modelo de procesamiento de lenguaje natural de código abierto más avanzado hasta el momento. Este es un equivalente directo de GPT-3 Curie.

El conjunto de datos de entrenamiento que utilizan estas dos librerías se llama "The Pile". Este conjunto de datos comprende datos de fuentes académicas como Pubmed, sitios web comunes como Wikipedia y Github, e incluso subtítulos de películas y programas de televisión. Se puede encontrar información más extensa sobre The Pile en <https://pile.eleuther.ai/>.

Al ser librerías de código abierto pueden descargarse y ser utilizadas por cualquier persona, pero es necesario tener en cuenta los siguientes requerimientos hardware: GPT-J, por ejemplo, necesita alrededor de 25 GB de RAM para funcionar y muchas CPU. Sin embargo, en las CPU, GPT-J es muy lento, por lo que es mucho mejor utilizar GPT-J en una GPU. Como GPT-J necesita alrededor de 25 GB de GPU VRAM, no cabe en la mayoría de las

GPU NVIDIA estándar que existen actualmente en el mercado (que tienen un máximo de 8 GB o 16 GB de VRAM).

Estos requisitos de hardware hacen que sea muy poco práctico probar GPT-J y GPT-Neo, y mucho menos usarlos de manera confiable para la inferencia en producción con alta disponibilidad y escalabilidad en mente.

Por estas restricciones EleutherAI recomienda su uso en Cloud a través de las librerías NLP Cloud o adaptar las librerías GPT-J para el caso de uso concreto para minimizar las necesidades hardware. Sin embargo, NLP Cloud también es una plataforma de pago y sólo se pueden utilizar de forma gratuita para aplicaciones que no estén en producción. Si se puede utilizar para pruebas.

3. *NLP Cloud* [61]

NLP Cloud es una API que da acceso al uso del procesamiento de lenguaje natural. La API se basa en los mejores modelos preentrenados de código abierto. Se pueden usar también modelos propios o entrenar modelos en la plataforma. NLP Cloud permite el uso de funciones de comprensión y generación de texto: extracción de entidades (NER), análisis de sentimientos, clasificación de texto, resumen de texto, respuesta a preguntas, generación de texto y etiquetado de parte del discurso (POS) y otras muchas que se pueden encontrar enumeradas en su sitio web.

Tal y como se indicó en el apartado anterior la API está disponible de forma gratuita para pruebas, hasta 3 solicitudes por minuto. Luego, los primeros planes pagos cuestan \$ 29 por mes (para 15 solicitudes por minuto).

NLP Cloud utiliza GPT-J and GPT-NeoX (continuación del proyecto GPT-Neo) y múltiples librerías de Python como Python's LangDetect library o PyTorch.

4. *NLTK*[62]

NLTK ofrece interfaces fáciles de usar para más de 50 corpus y recursos léxicos como WordNet, junto con un conjunto de bibliotecas de procesamiento de textos. Permite realizar tareas de preprocesado de texto, entre las se puede encontrar: la clasificación, tokenización, lematización o la exclusión de *stop words*, el análisis sintáctico y el razonamiento semántico. Es una de las librerías más utilizadas por la comunidad de desarrolladores por su fácil utilización y gran cantidad de documentación.

5. *LIWC*[18]

Linguistic Inquiry and Word Count (LIWC) es una herramienta desarrollada por Pennebaker junto con otros científicos por encargo del FBI. Posteriormente es utilizada en el ámbito científico y especialmente por el campo de la psiquiatría y la psicología. Tiene diccionarios en castellano, lo que la hace valorada en nuestro país, ya que la mayoría de las herramientas de PNL están muy trabajadas para el idioma inglés. Tal y como indican José Rúas, Mercedes Fernández e Iván Puentes en su trabajo sobre el uso de la herramienta para analizar mítines políticos [13] se utiliza para la evaluación cognitiva y emocional de textos a través de una serie de categorías psicológicas y estructurales. La herramienta analiza textos, palabra por palabra, en una clasificación de 70 variables lingüísticas que incluyen categorías de lenguaje

estándar (artículos, preposiciones, pronombres, etc.), procesos psicológicos (categorías de emociones positivas y negativas, variables cognitivas), palabras relacionadas con la relatividad espaciotemporal, los tiempos verbales y dimensiones tradicionales de contenido, organizadas de forma jerárquica.

Esta herramienta está utilizada dentro del campo de la psicología para el estudio de textos, organizando las palabras en diversas categorías prefijadas[2][19].

El diccionario de LIWC[18] en español cuenta con más de siete mil palabras y raíces de palabras, y con 72 categorías y subcategorías, el mismo número que en inglés. Las categorías se encuentran divididas en: dimensiones lingüísticas, procesos psicológicos, asuntos personales y lenguaje oral. LIWC contiene un grupo que incluye 12 categorías de puntuación tales como punto, coma, exclamación, entre otros.

Palabras como *soledad*, *llorar*, o *feliz* localizan en la categoría de **emociones**, que a su vez se subdivide en **afectos positivos y negativos**. En estas subcategorías se encuentran conceptos relacionados con estados emocionales concretos, como *miedo* o *duelo*. Otras categorías de la herramienta son **palabras de percepción inespecífica** como *ver*, *oír*, o *sentir* y de **procesos cognitivos complejos** como *debería*, *pensar*, *por qué*. Existen 22 categorías generales sobre partes de la oración: pronombres, verbos auxiliares, artículos numerales o conjunciones, es decir, palabras funcionales.

Los datos sobre la librería que provee las funcionalidades de la herramienta se encuentran disponible en Github en la siguiente ruta <https://github.com/chbrown/liwc-python#readme>. Sin embargo, el lexicón es de pago. Para investigaciones científicas se puede utilizar poniéndose en contacto directamente con el creador de la herramienta Dr. James W. Pennebaker. Para otros usos es necesario comprarla a través de Receptiviti, compañía dueña de la licencia.

En el repositorio de Python se puede encontrar un diccionario precargado en un archivo .dic. Pero el diccionario de Github no es un producto oficial de LIWC ni está afiliado de ninguna manera con el equipo de desarrollo de LIWC o Receptiviti. Es una adaptación de la comunidad de sw libre.

6. SpaCy[63]

SpaCy es otra biblioteca de código abierto y, por lo general, comprende modelos estadísticos previamente entrenados y vectores de palabras que admiten más de 60 idiomas.

La principal fortaleza de SpaCy es el reconocimiento de entidades nombradas, el etiquetado de parte del discurso, el análisis de dependencias, la segmentación de oraciones, la clasificación de texto, la lematización, el análisis morfológico y la vinculación de entidades.

7. Apache OpenNLP[64]

OpenNLP admite tareas como tokenización, segmentación de oraciones, etiquetado de parte del discurso, extracción de entidades con nombre, fragmentación, análisis, detección de idioma y resolución de correferencia. Aparte de esto, también incluye la máxima entropía y el aprendizaje automático basado en perceptrones.

8. CoreNLP[65]

Es otra plataforma de código abierto desarrollada por el grupo Stanford NLP como una posible solución para NLP en Java.

Actualmente está centrada en seis idiomas (árabe, chino, inglés, francés, alemán, español).

Las fortalezas de CoreNLP son los límites de las oraciones, las partes del discurso, las entidades nombradas, los valores numéricos y de tiempo, los análisis de dependencias y circunscripciones, la correferencia, el sentimiento, las atribuciones de citas y las relaciones.

9. AllenNLP[66]

Allen es una plataforma de código abierto basada en PyTorch. Es una biblioteca de aprendizaje profundo para PNL que se utiliza para tareas como responder a preguntas, etiquetado de roles semánticos, vinculación textual, texto a SQL.

10. SparkNLP[67]

SparkNLP es una plataforma de código abierto que ofrece más de 200 canalizaciones y modelos previamente entrenados que admiten más de 40 idiomas. SparkNLP admite transformadores como BERT[70][71], XLNet, ELMO y realiza anotaciones precisas y claras para PNL.

11. Gensim[68]

Gensim es una biblioteca de Python gratuita y de código abierto diseñada exclusivamente para textos sin procesar utilizando algoritmos de aprendizaje automático de calidad. Se utiliza para modelado de temas, indexación de documentos. La diferencia principal respecto al resto de librerías de lenguaje natural para Python reside en que Gensim es capaz de identificar automáticamente la temática del conjunto de documentos a tratar. También permite analizar la similitud entre ficheros, algo realmente útil cuando se utiliza la librería para realizar búsquedas.

4.3.4 Aprendizaje profundo o Deep Learning

El *Deep Learning* [37][38] es un tipo de ML que entrena a un ordenador para que realice tareas como las hacen los seres humanos, como el reconocimiento del habla, la identificación de imágenes o hacer predicciones.

En lugar de organizar datos para que se ejecuten a través de ecuaciones predefinidas, el *Deep Learning* configura parámetros básicos acerca de los datos y entrena a la computadora para que aprenda por cuenta propia reconociendo patrones mediante el uso de muchas capas de procesamiento. Se trata de modelos de aprendizaje automático basados en redes neuronales con una gran cantidad de capas no lineales, cada una de las cuales aprende un nivel de abstracción más alto que sus anteriores. Cada neurona es llamada también “perceptrón”.

Las capas de una red neuronal son las siguientes[37]:

- Capa de entrada: responsable de recibir las entradas.
- Capa oculta: procesa los datos de entrada para encontrar información oculta y realiza la extracción de características.
- Capa de salida: devuelve la salida.

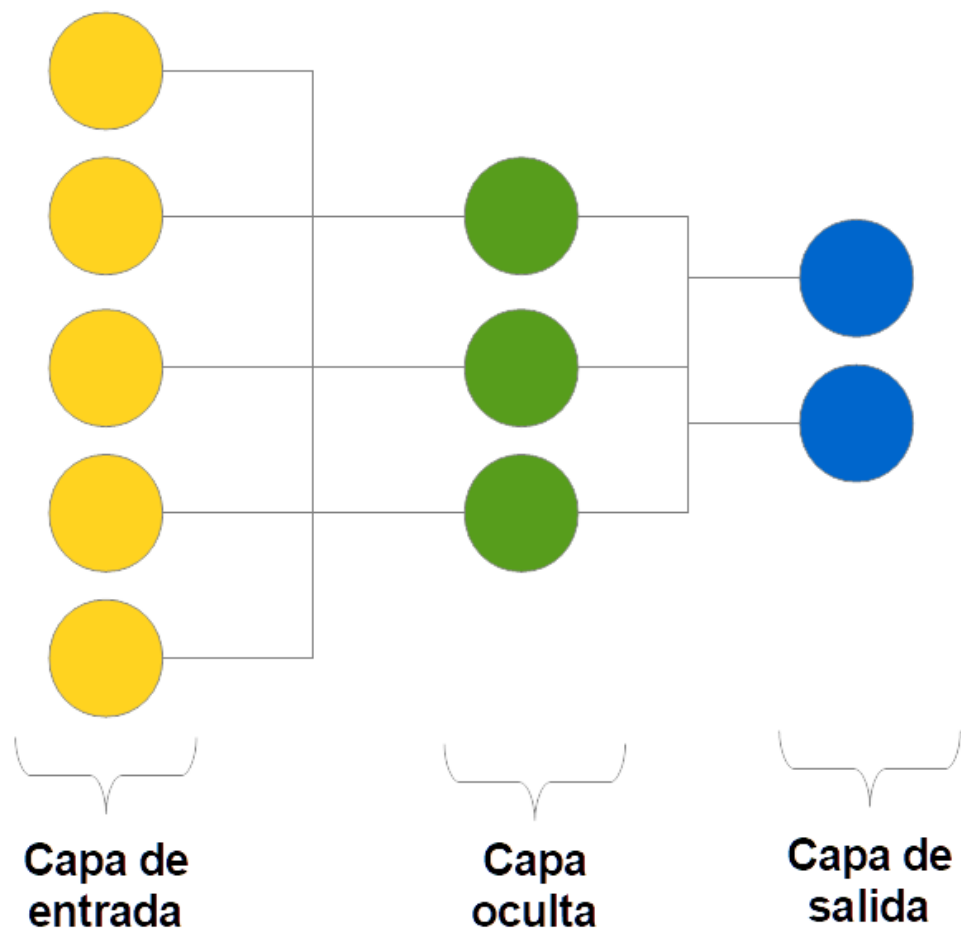


Figura 4-3 Red Neuronal Sencilla

Dentro de cada neurona o perceptrón se realizan las siguientes acciones:

- Producto de cada valor de entrada y el peso del canal por el que ha pasado.
- Cálculo de la suma de los productos ponderados. A esto lo llamamos la **suma ponderada**.
- Agregación de un sesgo exclusivo de la neurona a la suma ponderada.
- A cada suma final se aplica a una función particular llamada **función de activación** que es la que devuelve una salida a partir de un valor de entrada. Hay una serie de funciones de activación disponibles en una red neuronal. Algunos ejemplos de estas funciones son:
 - Sigmoide: Transforma los valores introducidos al rango (0,1).
 - Tangente hiperbólica: en este caso los valores serán transformados dentro del rango (-1,1).
 - ReLU: Anula los valores negativos y deja los positivos iguales a como entran.
 - Leaky ReLU o ReLU rectificado: multiplica los negativos por un coeficiente rectificador y deja los positivos tal cual entran.
 - Softmax: Transforma las salidas en una forma de probabilidades de tal forma que el sumatorio de todas las probabilidades sea 1.

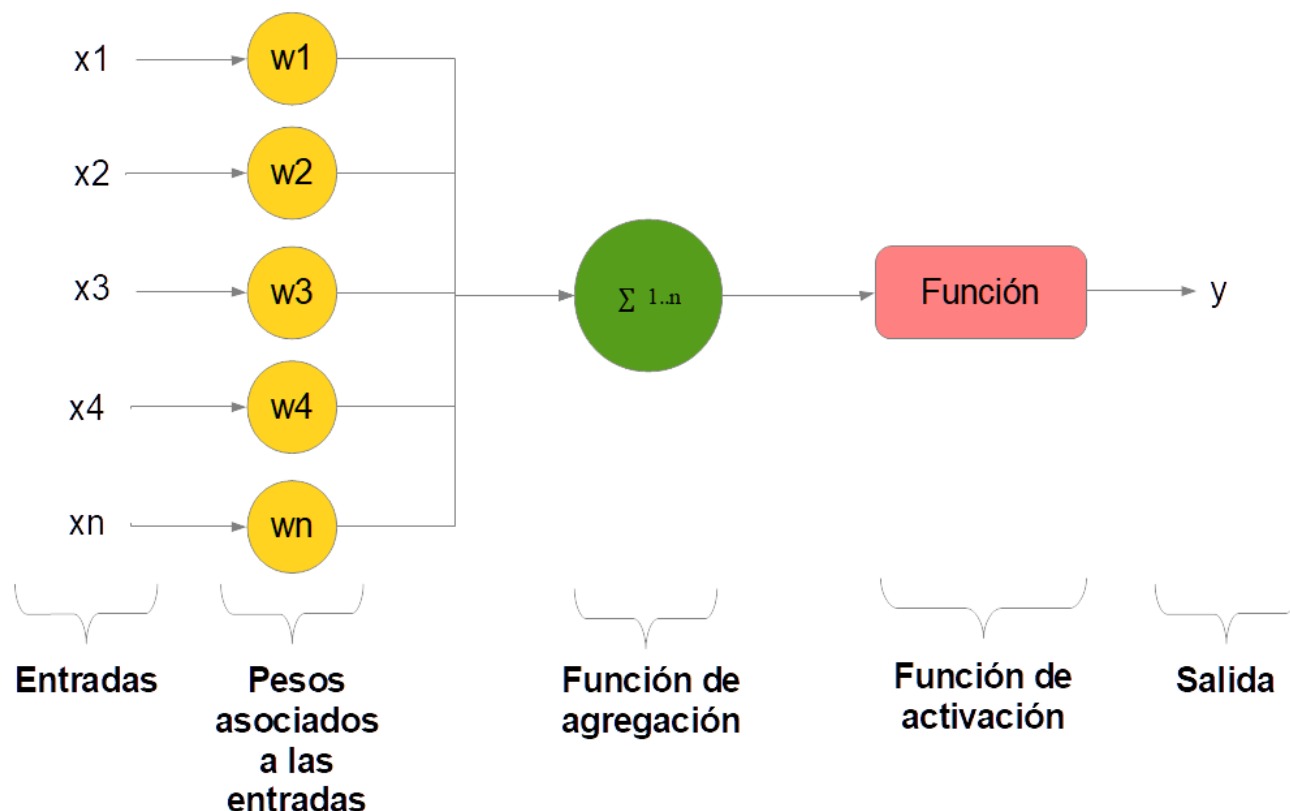


Figura 4-4 Neurona

Hay dos librerías principales para construir redes neuronales: TensorFlow [69] (desarrollada por Google) y PyTorch[43] (desarrollada por Facebook). Pueden realizar tareas similares, pero el primero está más orientado para su uso en proyectos y tiene una curva de aprendizaje más lenta, mientras que el segundo es muy útil para el desarrollo de prototipos rápidos, con curva de aprendizaje más rápida.

Debido a la gran capacidad de cómputo que se tiene hoy en día y a los nuevos métodos de entrenamiento de redes neuronales, este tipo de sistemas tiene gran repercusión y está siendo utilizado en muchos ámbitos, entre ellos a la extracción de semántica en textos.

Sin embargo, hoy en día el *deep learning*, y sobre todo en su uso para PLN ha sufrido una gran evolución con el uso de un nuevo paradigma: los *transformers*[41].

Junto con las redes neuronales aparecieron el uso de los *embeddings*, representaciones matriciales estáticas del texto, en las que cada palabra del vocabulario está codificada en un vector. Con estas matrices se es capaz de tener una idea del significado general de las palabras y de posición en las frases. Ya no se observan las palabras de forma individualizada, sino que se ve su semántica dentro de la frase. El uso de redes neuronales y *embeddings* constituyen lo que se denomina Redes Neuronales Recurrentes. Los textos completos, a través de sus *embeddings* o matrices de representación, van pasando a lo largo de toda la red neuronal para darle un sentido completo. De esta forma, en el PLN se deja de interpretar palabra a palabra para interpretar textos completos.

Partiendo de estas redes neuronales recurrentes, en 2017 en el artículo de Google llamado ‘*Attention Is All You Need*’[40], aparece por primera vez descrito el concepto de *Transformer*. Al igual que las redes neuronales recurrentes o RNN, trabajan con el texto de una sola vez para dar una semántica conjunta. Se utilizan para traducciones simultáneas o resúmenes inmediatos de textos. Este paradigma se basa en un concepto llamado *Attention* en inglés o Atención (de ahí el nombre del artículo de Google). El mecanismo de *atención* observa una secuencia de entrada y decide en cada paso qué otras partes de

la secuencia son importantes. En realidad, funciona como la forma en la que se centra el cerebro humano a la hora de realizar una lectura, aunque se lee palabra a palabra, para entender un texto, se va reteniendo los conceptos más importantes y un significado global de lo que se está leyendo.

Un *Transformer* [41] presenta una arquitectura que permite transformar una secuencia de palabras en otra con ayuda de dos partes: un codificador (*encoder*) y un decodificador (*decoder*). La idea clave es que se utiliza el *encoder* para analizar el contexto de la secuencia de entrada y el *decoder* es el encargado de generar la secuencia del output a partir de este contexto. Inicialmente es necesario calcular el *embedding* del texto sobre el que se quiere trabajar. Esto permite una representación numérica de las palabras, y que, además, las palabras con sentido semántico similar estarán cerca en el espacio vectorial calculado. En las RNN el concepto clave era la recurrencia aplicada sobre la matriz de *embedding*, pero en un *Transformer*, la importante la tiene la *atención*. Este mecanismo permite saber que otra palabra de la secuencia está relacionada con la palabra que se procesa en cada instante. Tanto el *encoder* como el *decoder* tienen implementada una red neuronal llamada *Feed Forward* y aplican funciones matemáticas como *Softmax* para realizar el cálculo de probabilidad de las palabras. No se va a profundizar en cómo funciona internamente un *Transformer*, ya que lo importante aquí es saber que funcionalidad aporta y como se utiliza. Para profundizar en su uso se recomienda la lectura del artículo de Google antes mencionado [40].

Los *Transformers* están revolucionando el mundo del PLN al convertirse en una de las opciones más elegidas por los expertos a la hora de procesar texto. Una de sus grandes mejoras es que reduce considerablemente el tiempo de entrenamiento y que aportan un entendimiento de la gramática del lenguaje no sólo de la semántica de las palabras concretas.

Una vez el modelo del lenguaje está entrenado en el idioma elegido (preentrenamiento), la última capa de la red neuronal o del Transformer puede ser cambiada para que solucione distintos tipos de problemas (*fine-tuning*).

Un punto muy interesante de los *Transformer* es que ya existen modelos que se pueden utilizar directamente en los proyectos que se vayan a realizar. Uno de los modelos más importante que existe actualmente es BERT[70][71] o Representación de Codificador Bidireccional de Transformadores, desarrollada por Google, el cuál lo desarrolló para comprender mejor las búsquedas de los usuarios del buscador. Este modelo ya ha sido preentrenado con un corpus gigantesco, en 102 idiomas diferentes. Además, se pueden encontrar muchas librerías de código abierto que implementan BERT para diferentes tipos de aplicaciones del NLP, como, por ejemplo, RoBERTa[42][44], desarrollado en PyTorch [43] y con diccionarios en castellano o DistilBERT, que utiliza la técnica llamada Distillation. Esta técnica se basa en sustituir la red neuronal de BERT en otra más pequeña cuando el modelo ya ha sido entrenado. Es decir, que en usos más sencillos que el propio entrenamiento del modelo se pueden utilizar redes neuronales más sencillas con el modelo en sí mismo.

➤ Conclusión sobre PNL

En la descripción de los requisitos del *Módulo o Subsistema de PLN y Análisis de Sentimientos* se definieron las necesidades de que el sistema preparase los textos que se descarguen de RRSS para el cálculo de los indicadores adecuados y el cálculo de estos.

Para ello se pueden extraer dos acercamientos y dos propuestas de diseño:

- **Tradicional:** Se propone utilizar NLTK[62]. Esta librería de Python es una de las favoritas de los desarrolladores de lingüística computacional por su flexibilidad y su potencia. Existe mucha documentación y código abierto ya generado que se puede utilizar como referencia y base en los desarrollos. Aunque realmente NLTK es muy amplia y tiene mucha capacidad

aporta funciones que permiten preparar los textos de forma directa para tokenizarlos y normalizarlos, presentando todos el mismo aspecto para el posterior cálculo de los indicadores. Una vez realizadas estas tareas, parece claro, tras el estudio de este capítulo, que LIWC [18] es la herramienta que está más preparada para el estudio de textos en el ámbito de la psicología, ya que permite contar palabras en función de las categorías semánticas necesarias para la evaluación psicológica de los textos, tal y como se ha explicado en el punto del presente apartado que trata en profundidad sobre la librería.

Utilizando estas dos herramientas de forma combinada se plantea un diseño que define el cálculo de los indicadores en función de los textos descargados y almacenados en el sistema.

- **Innovadora:** Tal y como se ha presentado, el *Deep learning*, permite obtener resultados sin necesidad de una preparación tan exhaustiva de los textos, y, además, se aleja de fórmulas tradicionales basadas en el conteo de palabras ya que el software realiza un análisis de los textos en su conjunto. Este diseño permitiría descubrir relaciones entre los rasgos y los textos que probablemente no han sido tenidos en cuenta. Por lo tanto, si se utiliza este tipo de herramientas para el diseño, no será necesario el cálculo de los indicadores, sino que el entrenamiento consistiría en el vector de entrada compuestos por textos descargados de RRSS y el valor del rasgo ENCUIST para ese texto. Los nuevos valores de rasgos calculados se inferirán directamente del análisis de los textos por parte del algoritmo de Deep Learning. Como se ha explicado, el uso de los *Transformers* ya desarrollados, y más en concreto, aquellos que tienen diccionarios en castellano y están basados en BERT, son de las mejores soluciones para resolver las necesidades de este sistema. Siguiendo esta línea, RoBERTa es la herramienta elegida para el diseño propuesto. Anteriormente se ha indicado que una de las grandes mejoras de los *Transformers* es que reducen considerablemente el tiempo de entrenamiento y que aportan un entendimiento de la gramática del lenguaje no sólo de la semántica de las palabras concretas, lo que hace que se hayan convertido en unas de las herramientas favoritas para el procesamiento de texto por parte de los expertos.

4.4 Análisis de sentimientos

El análisis de sentimiento es el proceso de determinar el tono emocional que hay detrás de una serie de palabras. También se le denomina minería de opinión.

Este tipo de análisis se basan en ML y PNL. Entre los usos que tiene realizar este tipo de minería de textos está la de saber qué opinión tienen los autores de los textos o bien sus estados de ánimo. Es decir, las emociones que están detrás de un texto. En general permiten clasificar un texto en positivo, negativo y neutro.

Hoy en día, las empresas de marketing utilizan mucho este tipo de procedimientos para hacer seguimiento de las campañas que lanzan en redes sociales.

La mayoría de las librerías descritas en el apartado anterior permiten realizar un análisis de sentimientos, entre ellas NLTK[62], en concreto, con su librería *SentimentIntensityAnalyzer*.

La propia herramienta LIWC también permite hacer un análisis de sentimientos ya que contiene un diccionario con las emociones clasificadas en positivas y negativas.

Y por supuesto, existen librerías basadas en BERT que permiten realizar también análisis de sentimientos, como el propio RoBERTa[42][44].

Tal y como se ha descrito la mayoría de las herramientas de PNL se pueden utilizar para realizar un análisis de sentimientos.

Por lo tanto, no se ve necesario utilizar soluciones diferentes a las ya planteadas para el *Módulo de Procesamiento de Lenguaje Natural y Análisis de Sentimientos*. La propia herramienta NLTK permite

realizar dicho análisis y se propone para formar parte del diseño en el cálculo de los sentimientos que se podrán inferir de los textos.

4.5 Almacenamiento

En este sistema de información, como en los demás, la información es el activo más valioso. Tener un almacenamiento adecuado de estos datos es esencial para adaptarse rápidamente a los cambios que pueda sufrir el entorno al que da servicio el sistema y también del que informativamente se nutre. Más si estamos hablando de una fuente de información tan cambiante y activa como las redes sociales. Para sacar el máximo provecho posible a la información el diseño del almacenamiento es fundamental.

El escenario actual del almacenamiento se divide básicamente entre BBDD relacionales y no relacionales, es decir, SQL o NoSQL. Por lo tanto, es necesario conocer las diferencias entre ambas y qué prestaciones dan cada una para realizar la mejor elección para el almacenamiento de los datos necesarios para el sistema de perfilado indirecto.

El teorema de CAP es bien conocido el mundo del diseño sw y de necesaria aplicación a la hora de elegir un almacenamiento SQL o NoSQL. En Ciencias de la Computación se enunció este teorema que indica que un sistema distribuido de almacenamiento de datos no podemos garantizar consistencia y disponibilidad (para actualizaciones), al tiempo cuando el sistema sufre una partición (queda separado en dos o más islas). El Teorema CAP viene de las siguientes características:

- **Consistency (Consistencia):** La consistencia garantiza que una lectura nos retornará la escritura más reciente de un registro dado.
- **Availability (Disponibilidad):** El sistema debe garantizar la entrega de la información solicitada en un periodo razonable de tiempo (Ni error ni timeout).
- **Partition tolerance (Tolerancia a particiones):** El sistema, que estará compuesto por varios nodos, debe seguir funcionando, aunque algunos nodos no se encuentren disponibles ya que la información debe ser consistente en todos los nodos.

Lo que dice el teorema es que sólo se puede garantizar dos de esos atributos de forma simultánea.

4.5.1 Bases de datos relacionales o SQL

Estas bases de datos garantizan la consistencia y la disponibilidad. Los datos están organizados en un conjunto de tablas formalmente descritas. La información se relaciona a través de identificadores de los datos, de ahí su nombre. **La interfaz estándar de programa de usuario y aplicación a una base de datos relacional es el Lenguaje de Consultas Estructuradas (SQL).** A través de los comandos SQL se interactúa con los datos del sistema, tanto para la interfaz de usuario como para la generación de informes o la comunicación entre los diferentes módulos del sistema. Los identificadores estructurados dentro de las tablas permiten dirigir los comandos SQL.

Se llevan utilizando desde los años 80 y su uso es ampliamente extendido en prácticamente todas las organizaciones. Oracle, DB2 o SQL Server son algunas de las BBDD relacionales más utilizadas hoy en día.

La principal ventaja de la base de datos relacional reside en la sencillez del modelo relacional y la integridad que dicho modelo proporciona a los datos. Permite manejar grandes cantidades de datos con puntos de relación entre sí, gestionándolos de forma segura y conforme a unas normas y un modo uniforme.

Sin embargo, su ventaja es también su principal debilidad frente a la rápida evolución a la que se deben enfrentar hoy en día los sistemas. En estas bases de datos existe mucha dependencia del modelo, lo que implica que la evolución continua sea compleja. Un cambio de este implica muchas veces profundos cambios en el diseño y, por tanto, en el código software de los sistemas.

4.5.2 Bases de datos no relacionales o NoSQL

Son las que, a diferencia de las relacionales, no tienen un identificador que sirva de relación entre un conjunto de datos y otros. Con relación a formatos, la información de una base de datos puede ser almacenada en tablas o documentos.

Las bases de datos no relacionales tienen un esquema dinámico, no se requiere la estructura de los datos para su manipulación. Los datos se pueden almacenar de cualquier manera, columnas, documentos, gráficos, etc., y cada documento puede tener su propia estructura, sin afectar los demás, puede agregar más campos a medida que se avanza. Están conformadas por documentos, campos y datos del documento, además, se almacenan por colecciones.

Una posible definición de datos no estructurados, son aquellos datos no almacenados en una base de datos tradicional. La información no estructurada no puede ser almacenada en estructuras de datos relacionales predefinidas.

4.5.2.1 Características de las BBDD NoSQL

Las principales características de los datos no estructurados son las siguientes[25][26]:

- **Volumen y crecimiento:** el volumen de datos y la tasa de crecimiento de los datos no estructurados es muy superior al de los datos estructurados. Por ejemplo, si la fuente de los datos es de RRSS donde la actualización de los posts o de los perfiles es continua.
- **Orígenes de datos:** El origen de los datos es muy diverso. Datos generados en redes sociales, datos generados en foros, e-mails, datos extraídos de la web empleando técnicas de web semántica, documentos internos de la organización (word, pdf, ppt).
- **Almacenamiento:** Debido a su estructura no se puede emplear arquitectura relacional, siendo necesario trabajar con herramientas 'Big Data'. Puede ser incluso necesario pensar en un almacenamiento en Cloud.
- **Idiomas y terminologías de información almacenada:** por ejemplo, que puedan cambiar los idiomas de la información o bien, que, si se almacenan textos, estos puedan contener términos que sean llamados de uno u otro modo.
- **Otros aspectos** tales como la seguridad, donde hay información que incluso podría contener virus y es necesaria ser tratada de forma diferente que el resto de la información almacenada.

4.5.2.2 Tipos de BBDD NoSQL

Se pueden reducir a cuatro tipos de BBDD[26]:

- **Base de Datos Columnar:** Mientras que la mayoría de las bases de datos distribuyen la información en filas, las bases de datos columnares funcionan de la manera contraria: los datos se distribuyen en columnas. Por cada entrada, hay una columna, por lo tanto, los datos de cada entrada están dispuestos uno debajo del otro. Las bases de datos columnares se utilizan cuando hay que analizar grandes cantidades de datos.

Las BBDD columnares están desarrolladas para mejorar el cuello de botella que puede ser el acceso a disco cuando se está tratando con grandes volúmenes de información, es decir, es especialmente adecuada en entornos BigData. Además, como los datos de una columna son siempre del mismo tipo, son susceptibles de ser comprimidos de forma muy eficiente.

Cassandra es un claro ejemplo de este tipo de Bases de Datos, basada en el software libre de Apache Hadoop.

- **Clave-Valor:** Esta base de datos almacena los datos en forma clave valor. La clave tiene como única condición que debe ser única, y normalmente es autogenerada. Son muy utilizadas en el desarrollo rápido de prototipos y sus principales ventajas son su sencillez (básicamente tienen operaciones de lectura, escritura y borrado) y su versatilidad para implementar cualquier modelo. Pero esta sencillez también es su debilidad ya que a veces no disponen de herramientas suficientes para la elaboración de acciones complejas.

Redis es uno de los ejemplos más conocidos y se trata de una BBDD clave-valor de código abierto.

- **Orientadas a Documentos o documentales:** Este tipo de base de datos permite la recuperación, guardado y gestión de documentos que de alguna manera sí tienen cierta estructura. Los documentos que contienen estas bases de datos pueden referirse a documentos de texto, archivos XML o JSON (JavaScript Object Notation) es un formato de texto sencillo para el intercambio de datos ampliamente utilizado hoy en día debido a su potencia, simplicidad y versatilidad. La dinámica que ofrecen estos modelos de bases de datos permite que su modificación sea más sencilla que la de los modelos relacionales. A la hora de modificar e incluir nuevos tipos de datos, no existe la necesidad de reinventar por completo la estructura de la base de datos. Los procesos de modificación y ampliación estructural son simples y basta con solo agregar los objetos a la base de datos. Sobre todo, permiten un modelado flexible.

Los documentos se agrupan en colecciones. Las **bases de datos de documentos** proporcionan mecanismos de consulta para las colecciones de documentos permitiendo el acceso de forma individual o conjunta los atributos particulares o especiales que pueda tener.

MongoDB es un ejemplo de este tipo de BBDD muy utilizado hoy en día.

- **Orientadas a Grafos**[76][77][78]: Las bases de datos NoSQL en grafo permiten representar los datos utilizando estructuras de grafos. Un grafo es una representación abstracta de un conjunto de objetos. Los objetos de los grafos se representan mediante vértices (también llamados nodos) y aristas. El modelo en grafo es útil cuando los datos a almacenar tienen multitud de interrelaciones entre sí, y cuando la importancia recae más en las interrelaciones que se establecen entre los datos, que en los propios datos en sí. Los grafos más utilizados en el modelado de datos son aquellos que no sólo presentan vértices y aristas, si no también, etiquetas y propiedades. Los nodos o vértices representarían conceptos, mientras que las aristas permiten relacionar los nodos, es decir, relaciones entre los conceptos. Las etiquetas aportan semánticas a nodos y aristas.

Este tipo de BBDD son muy flexibles a la hora de asumir cambios en el modelo de datos en comparación a las relacionales y mantienen un alto rendimiento frente a altos volúmenes de datos. Los nodos pueden ser de diferente tipo y se pueden añadir atributos adicionales con bastante facilidad a lo largo de la vida del sistema. Las búsquedas sobre relaciones entre los nodos serán muy rápidas ya que la indexación se hará a través de dichas relaciones.

La principal desventaja de este tipo de modelos es la de que no son fácilmente distribuibles ya que hay muy alta dependencia entre los datos. En este sentido son similares a las BBDD relacionales.

Uno de las BBDD orientadas a grafos más conocidas es Neo4j[29]. Es una herramienta muy utilizada en BigData. Su origen se remonta a 2010, está escrita en JAVA y su lenguaje nativo para la creación de los grafos y consulta sobre ellos es *Cypher*, sin embargo, se integra perfectamente con múltiples lenguajes como Java, PHP, Ruby, .Net y Python, entre otros. Dispone de un *API rest* (interfaz de comunicación basado en transferencia segura a través de internet, por ejemplo, en formato JSON o XML) para recuperar la información. Se distribuye en dos versiones: la *community edition (open source)* y la *enterprise edition*.

BBBB	Características		Ejemplos
Relacionales		Altamente fiables	Oracle, SQL Server
		Muy dependientes del modelo	
No relacionales		Modelado flexible	
	Columnar	Grandes volúmenes de información	Cassandra
		Alto rendimiento	
	Clave - valor	Gran sencillez	Redis
		Difícil manejo de operaciones complejas	
	Documentales	Altamente versátiles	MongoDB
		Simplicidad a la hora de insertar nuevos objetos	
	Orientadas a Grafos	Representación muy potente de relaciones entre los datos	Neo4j
		Difíciles de particionar	

Tabla 4-2 Tipos de BBDD, características y ejemplos.

➤ Conclusión sobre el Almacenamiento

El almacenamiento en este sistema es el corazón y debe estar accesible por todos los módulos que lo componen.

Se opta por almacenamiento NoSQL debido a la flexibilidad a cambios en el modelo que este tipo de BBDD aportan. Tal y como se comentó en el apartado de requisitos del almacenamiento, este debe ser flexible para poder representar cambios en los perfiles descargados de las RRSS y también, en el modelo ENCUIST, que aún es muy reciente y está siendo objeto de avances e investigaciones actualmente.

Para cubrir todos los requisitos del sistema de perfilado indirecto, se ve necesario utilizar dos tipos de almacenamiento diferentes:

- BBDD NoSQL documental: Los datos descargados de RRSS, una vez se hayan convertido en archivos **JSON**, se insertarán en esta BBDD. Para el diseño propuesto se escoge **MongoDB**[27] por varios motivos que han sido descritos en el apartado anterior:
 - Amplia utilización en la comunidad de desarrolladores, lo que implica que exista una gran cantidad de documentación, incluso mayor que para otras BBDD NoSQL.
 - Es una herramienta de código abierto, lo que hace que su coste sea muy bajo.
 - No conlleva mucha necesidad de cómputo, se puede ejecutar con pocos recursos HW.
- BBDD orientada a grafos: Para poder analizar la información basándose en las relaciones entre los datos representadas a través de los grafos. De esta forma se podrán encontrar

relaciones que no se han tenido en cuenta previamente y puede ser fácilmente utilizada para el *subsistema de informes*. Su fácil representación gráfica la hace perfecta para el estudio de los datos y sus relaciones. Para la propuesta del actual diseño se elige el producto Neo4J. Los motivos de su elección son:

- Utiliza grafos de propiedad, con etiquetas, que permite catalogar nodos y aristas de una forma muy sencilla permitiendo desarrollos con poco esfuerzo.
- Es utilizado por una amplia comunidad de usuarios que le dotan de una amplia experiencia de soporte en red y de un gran volumen de documentación.
- La herramienta tiene una gran velocidad de respuesta, aunque este no es un requisito propio del sistema de perfilado indirecto de forma automática.
- Rápida curva de aprendizaje apoyado en la gran cantidad de libros sobre Neo4j[29][30][31] escritos en el seno de la comunidad de desarrolladores.
- Está muy recomendado para el descubrimiento de nuevos patrones. Este último punto va a dotar al sistema de un requisito no expuesto explícitamente pero que lo que lo hace muy atractivo para el usuario, ya que va a poder analizar la información apoyándose en esta funcionalidad para poder descubrir nuevos patrones entre los perfiles y los rasgos, es decir, para realizar *discovery* de nueva información. El descubrimiento de patrones, lo hace un poderoso aliado del ML y de la IA.

5 DISEÑO DE LOS SUBSISTEMAS

Una vez hecho el recorrido por las diferentes herramientas y tecnologías relacionadas, y habiendo concluido en cada una de ellas cuál es la elegida para el sistema, se está en disposición de plantear más en profundidad el diseño propuesto.

Sin embargo, antes de entrar con el diseño detallado de cada módulo, es necesario hacer una propuesta de las herramientas y recursos software que se recomiendan para que, en fases posteriores a la conclusión de este trabajo, se realice el desarrollo, considerándose que la elección de los recursos de desarrollo a utilizar forma parte del diseño.

5.1 Recursos del desarrollo sw

El lenguaje de programación que se propone como base para el desarrollo del sistema de perfilado indirecto automático es **Python**[73]. Este lenguaje de alto nivel es el idóneo para el *data mining*, la inteligencia artificial, el machine learning y otros muchos tipos de proyectos por múltiples motivos:

- Se trata de un lenguaje multiplataforma, al igual que java, pero más sencillo, ya que su diseño fue orientado a la eliminación de estructuras superfluas.
- La curva de aprendizaje es muy rápida ya que es fácil de leer y de escribir.
- Es extremadamente compatible con otros lenguajes de programación como Java o C.
- Se trata de software libre, con lo que puede ser utilizado para cualquier propósito y está dotado de una amplia documentación por parte de la comunidad de programadores.
- La mayoría de los Científicos de datos o especialistas en *DataScience* prefieren este lenguaje.

Como entorno de desarrollo se utilizará **Jupyter Notebook**[74]. Estos *cuadernos (notebooks)* permiten combinar texto y código, organizados en *celdas*, lo cual es más cómodo para desarrollar y documentar al mismo tiempo. Es ejecutable en cualquier navegador web. Existen muchas ventajas para el uso de los notebooks de Jupyter:

- Da soporte a 40 lenguajes de programación, entre ellos, Python.
- Es una herramienta, al igual que Python, Opensource.
- Tal y como se ha adelantado, permite en un único documento combinar el código que se está desarrollando, la salida que da el programa de desarrollado y la documentación, que pueden combinar texto y multimedia.
- Permite la gestión de versiones.
- Su uso está muy extendido en el mundo de la ciencia del dato y del machine learning. Hay amplia documentación al respecto.



Figura 5-1 Jupyter y Python

A continuación, se puede ver una captura de pantalla de un notebook de Jupyter utilizando algunas librerías de Python que contienen las herramientas para utilizar Random Forest[35], mezcladas con entradas y salidas, junto con documentación.

Load packages

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.ensemble import RandomForestClassifier
```

/anaconda3/lib/python3.7/site-packages/sklearn/ensemble/weight_boosting.py:29: DeprecationWarning: numpy.core.umath_tests is a n internal NumPy module and should not be imported. It will be removed in a future NumPy release.
from numpy.core.umath_tests import inner1d

Load Data

```
In [2]: url = 'https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2020/2020-07-07/coffee_ratings.csv'
df = pd.read_csv(url)
```

Look at first few rows

```
In [3]: df.head(3)
```

```
Out[3]:
```

	total_cup_points	species	owner	country_of_origin	farm_name	lot_number	mill	ico_number	company	altitude	...	color	category_two_defects
0	90.58	Arabica	metad plc	Ethiopia	metad plc	NaN	metad plc	2014/2015	metad agricultural developmet plc	1950-2200	...	Green	0
1	89.92	Arabica	metad plc	Ethiopia	metad plc	NaN	metad plc	2014/2015	metad agricultural developmet plc	1950-2200	...	Green	1
2	89.75	Arabica	grounds for health admin	Guatemala	san marcos barrancas "san cristobal cuch	NaN	NaN	NaN	NaN	1600 - 1800 m	...	NaN	0

3 rows x 43 columns

Figura 5-2 Jupyter Notebook[35]

5.2 Almacenamiento

La información descargada de las **RRSS se va a almacenar en ficheros JSON**. Esto implica una necesidad gestión y almacenamiento de ficheros. Además, en esta unidad de datos, se pueden incluir más elementos que sean necesarios con pocos cambios en el código.

El uso de diferentes tecnologías de BBDD se puede conseguir una mejora del sistema, ya que se complementarán las ventajas que da cada tecnología en concreto.

Tal y como se comentó en el apartado dedicado al almacenamiento, para cubrir todos los requisitos del sistema de perfilado indirecto, el almacenamiento se va a dividir en dos BBDD de diferentes naturalezas. Gran parte de la información estará replicada en ambas, pero organizada de diferente manera:

- BBDD NoSQL documental: **MongoDB**[27] para los archivos JSON dará servicio de almacenamiento a todos los subsistemas.
- BBDD orientada a grafos: **Neo4j**[29] replicará en modo grafo los metadatos descargados y los datos calculados facilitando la generación de informes para el usuario y aportando la flexibilidad necesaria para cubrir las necesidades de análisis de resultados. No almacenará los textos que estarán almacenados únicamente en los JSON de MongoDB.

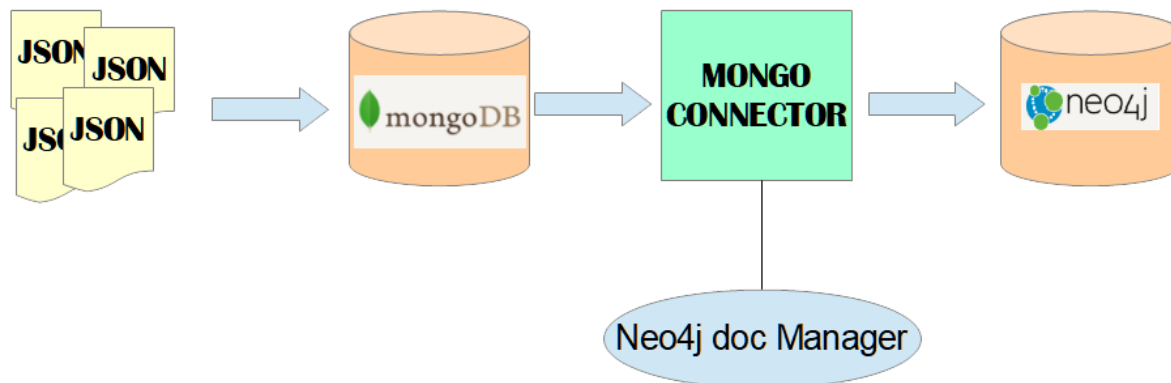


Figura 5-3 Almacenamiento e interconexión

Tal y como se puede ver en la Figura 5-3 Almacenamiento e interconexión, los archivos JSON entrarán directamente en MongoDB, y a través de *mongo-connector*, en este caso *Neo4j doc Manager*, se replicarán con su estructura adecuada en Neo4j[32]. Los desarrolladores de MongoDB han proporcionado el proyecto *mongo-connector* que proporciona un mecanismo para escuchar todas las operaciones de actualización en MongoDB y facilita la duplicación de esas actualizaciones en otro sistema. Para facilitar la sincronización de datos de MongoDB a una instancia de Neo4j, la comunidad ha implementado un administrador de documentos de Neo4j para *mongo-connector*. Está destinado a la sincronización unidireccional en vivo de MongoDB a Neo4j, donde tiene ambas bases de datos ejecutándose para aprovechar las fortalezas de cada base de datos en el sistema.

Toda la información necesaria para su uso e instalación se puede encontrar en Neo4j para desarrolladores [30]. Estas librerías están desarrolladas en Python y son fácilmente usables a través de Jupyter.

5.2.1 Colecciones en MongoDB

En el apartado de Análisis de Subsistemas se detallaron los almacenes de datos que debían existir en el sistema a alto nivel (Figura 3-1 Análisis de subsistemas). Una vez concluido que el tipo de almacenamiento principal que se va a utilizar es MongoDB[28], se detallan a continuación las características de diseño lógico que debe tener:

En Mongo no existen tablas relacionales porque se trata de una BBDD NoSQL basada en documentos, como ya se explicó anteriormente. Existen el concepto de colecciones. Estas colecciones se pueden relacionar entre ellas a través de *referencias*. Es necesario establecer estas referencias para mencionar ellas en el modelado de los subsistemas.

I. Colección Perfiles:

- a. Almacenará en formato JSON los perfiles descargados en raw. Este JSON incluirá toda la metainformación que se detalló en la parte de requisitos.
- b. Almacenará en formato JSON los perfiles una vez que se hayan preparado para el PLN (Tokenización y normalización).
- c. Se utilizarán como referencia el *idPerfil* y el *idUsuario*. Esto es necesario porque es factible descargar n perfiles de un usuario, siendo esta n dependiente del número de

RRSS con las que estemos trabajando (en este caso inicial propuesto únicamente Facebook y Twitter, pero se puede crecer a futuro). También son necesarias estas dos referencias si se quiere descargar a futuro de nuevo el perfil del usuario para, por ejemplo, ver una evolución en tiempo de su perfil, siendo necesario mantener un histórico de las descargas.

II. Colección Indicadores ENCUIST:

- a. Almacenará los indicadores ligados a los textos procesados por el subsistema de PLN.
- b. Estos indicadores irán en formato JSON, lo que le permitirá la flexibilidad de añadir nuevos indicadores si estos son necesarios a lo largo de la vida del sistema y de la evolución de los estudios sobre ENCUIST.
- c. Se utilizarán como referencia el **idPerfil** y el **idUsuario**. Esto permitirá hacer las agrupaciones y búsquedas necesarias por subsistemas de consulta en función del perfil descargado (incluyendo su tipología, fechas, metainformación, etc.), o por el usuario al que se haga referencia y su metainformación.

Indicadores (Sumatorios totales por perfil descargado)

Total de palabras

Media de palabras por frase

Palabras mayores de 6 letras

Palabras de función

Conjunciones

Inclusivas

Exclusivas

Negación

Negaciones

Adjetivos

Dimensiones lingüísticas

Procesos psicológicos

Asuntos personales

Lenguaje oral

Pronombres personales

Primera persona del singular

Segunda persona del singular

Tercera persona del singular

Primera persona del plural

Segunda persona del plural

Tercera persona del plural

Pronombres impersonales

Artículos

Números

Preguntas

Verbos
Indicativo
Subjuntivo/Condicional
Pasado
Presente
Futuro
Primera persona singular
Segunda persona singular
Tercera persona singular
Primera persona plural
Segunda persona plural
Tercera persona plural

Tabla 5-1 Indicadores ENCUIST de personalidad. Sumatorios de palabras.

III. Colección Sentimientos:

- Almacenará el resultado del cálculo de sentimientos de cada uno de los perfiles.
- Estos indicadores irán en formato JSON, lo que le permitirá la flexibilidad antes mencionada. Actualmente los sentimientos se calculan en *Positivo*, *Negativo* o *Neutro*, pero puede que a futuro esto puede cambiar.
- Se utilizarán como referencia el **idPerfil** y el **idUsuario**. Esto permitirá hacer las agrupaciones y búsquedas necesarias por subsistemas de consulta en función del perfil y del usuario, tal y como se ha mencionado anteriormente. Aunque es evidente que se pueden buscar otros datos para realizar las consultas.

IV. Colección Rasgos:

- Almacenará el resultado de haber aplicado el ML a los indicadores anteriores.
- Su formato también será JSON, cuya flexibilidad, permitirá la incorporación de nueva información relativa a los rasgos de ENCUIST, si se producen cambios en la forma de calcularlos o sean necesarios incluir o dejar de calcular rasgos por parte de los expertos en psicología que estén haciendo uso del sistema.
- Al igual que en las anteriores colecciones sus referencias serán el **idPerfil** y el **idUsuario**.
- En cada registro JSON será necesario incluir también **Rasgo** y **Algoritmo** que ha servido para calcular el rasgo en concreto (se explicará en el *subsistema de Análisis de Rasgos*).
- El subsistema de ML que calcule los rasgos puede requerir incluir agregados de los rasgos, quedando esto reflejado en los JSON con las referencias correctas.
- Los rasgos que deben calcularse y almacenarse son los que se recogen en la siguiente tabla y que ya fueron explicados y mencionados en Indicadores para el perfilado:

Rasgo	Subrasgo
Extroversión	Búsqueda de sensaciones
	Apertura a la experiencia
Neuroticismo	Ansiedad

Ira

Asco

Insensibilidad emocional

Impulsividad

Tabla 5-2 Rasgos ENCUIST de personalidad

A continuación, se incluye una tabla con las sentencias básicas de creación de la base de datos de *PerfiladoIndirecto*, y las cuatro colecciones descritas anteriormente junto con sus índices.

MongoDB				
Creación de BBDD	USE <i>PerfiladoIndirecto</i>			
Creación de tabla	db.CreateCollection ("Perfiles")	db.CreateCollection ("Indicadores")	db.CreateCollection ("Sentimientos")	db.CreateCollection ("Rasgos")
Creación de índices o referencias	db.Perfiles.createIndex ({idPeril:1, idUsuario:1})	db.Indicadores.createIndex ({idPeril:1, idUsuario:1})	db.Sentimientos.createIndex ({idPeril:1, idUsuario:1})	db.Rasgos.createIndex ({idPeril:1, idUsuario:1, Rasgo:1, Algoritmo:1})

Tabla 5-3 Creación de BBDD, tablas y referencias en MongoDB

5.2.2 Grafos en Neo4j

La representación en Neo irá directamente relacionada con las preguntas que se vayan a realizar sobre el modelo. Estas preguntas pueden evolucionar a lo largo de la vida del sistema, ya que los grafos son fácilmente configurables y se necesitan pocos cambios en el código.

Las consultas iniciales que se van a poder realizar sobre el sistema de informe son, por ejemplo, las siguientes:

- Devolver todos los patrones de rasgos que hay en función de su algoritmo. Este informe permitirá a los expertos poder comparar los resultados de los diferentes modelos de ML.
- Búsqueda de un determinado patrón de rasgos. De esta manera los expertos podrán buscar un determinado perfil que tengan identificado en base a un determinado patrón de rasgos. Esto se podría aplicar por ejemplo a los líderes de un grupo, ya que estarán almacenados también los foros o grupos a los que pertenecen; a los perfiles más influenciados para buscar fuentes; a individuos que cumplan un perfil que encaje en un patrón criminal.
- Al estar almacenando la fecha de descarga se puede ver la evolución de los rasgos a lo largo del tiempo. Esto permitirá ver cómo evolucionan los individuos y analizar sus RRSS para ver que ha sucedido a lo largo de ese tiempo.

Para responder a estas preguntas iniciales el tipo de información que almacenarán los grafos serán los metadatos descargados de las RRSS y los rasgos calculados por cada algoritmo.

Datos para representar en los grafos

Nombre del perfil

Fecha de descarga
Identificador del grupo o del foro del que nos lo estamos descargando
Red social de la que se hace la descarga
Número de amigos
Número de grupos a los que pertenece (en el caso de Facebook)
Número de foros a los que pertenece (en el caso de Twitter)
Número de fotos descargadas
Número de likes que tiene el perfil
Número de veces que publica
Frecuencia media de publicación
Rasgo
Algoritmo

Tabla 5-4 Metadatos de RRSS, rasgos calculados y algoritmo de ML

Una modelización en grafo enfocada a responder preguntas como las anteriores quedaría de la siguiente manera:

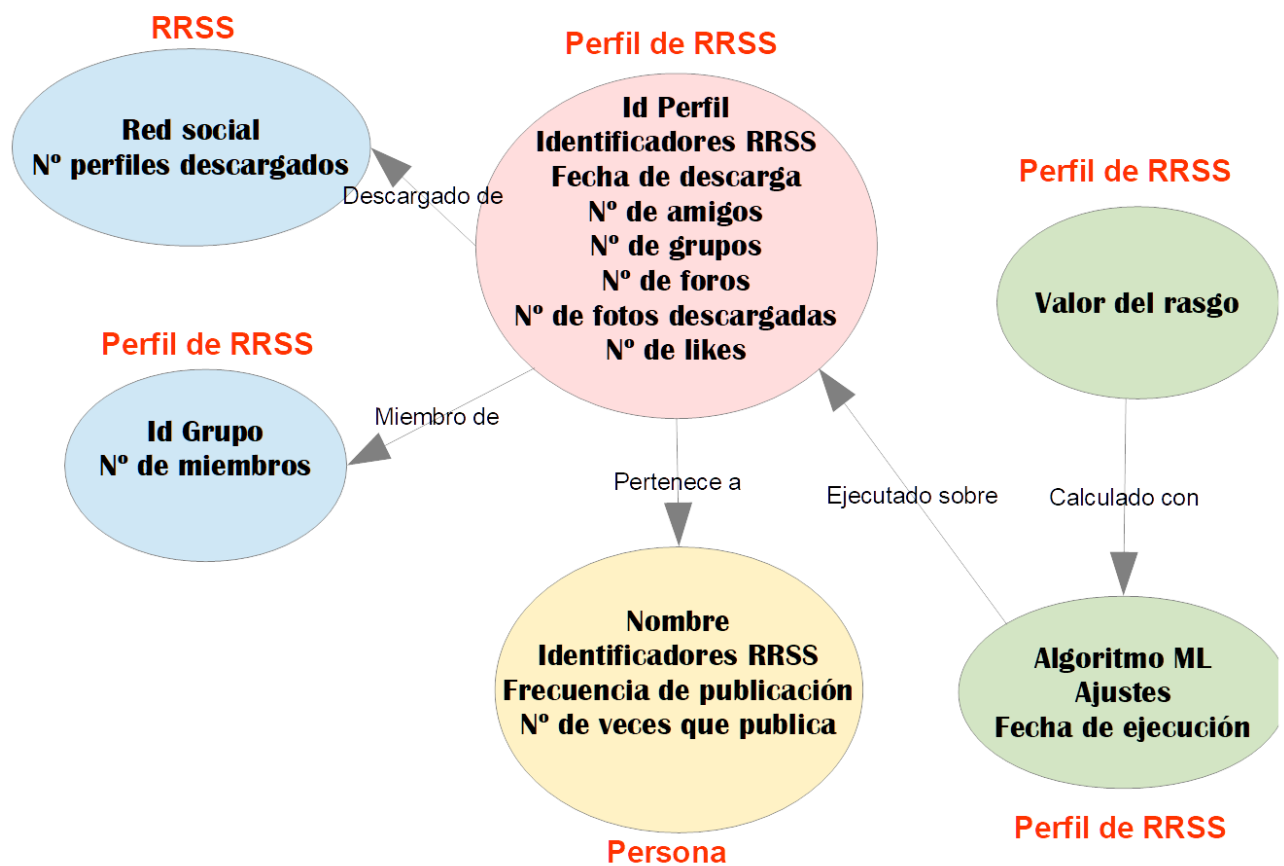


Figura 5-4 Modelo de grafos

Para la creación de los nodos y relaciones se utilizará Cypher. Se podrá crear nuevos tipos de nodos y relaciones en cualquier momento a lo largo del ciclo de vida del sistema. Por ejemplo, dos nodos muy

sencillos de crear pero que ilustran perfectamente la creación de nuevos elementos son los de las RRSS. El número de perfiles descargos comenzará siendo 0 en el inicio de la ejecución.

```
CREATE (FB:RedSocial {nombre:'FaceBook', numeroPerfilesDescargados: 0})
CREATE (TW:RedSocial {nombre:'Twitter', numeroPerfilesDescargados: 0})
```

Un ejemplo de perfil descargado sería de la siguiente manera:

```
CREATE (id1:PerfilRedSocial {idPerfil:'1', IdRedSocial:'angelito1898',
FechaDescarga:'09/04/2022', NAmigos:'17', NGrupos:'0', NForos:'24', NFotos:'4', NLikes:'109'})
```

La relación entre un perfil y su red social se crearía con la siguiente sentencia:

```
(id1)-[:DESCARGADO_DE]->(TW)
```

5.3 Subsistema de extracción de perfiles de RRSS

Este es el subsistema que se encargará de extraer de RRSS los datos de los usuarios necesarios para realizar la tarea de perfilado indirecto.

Como se ha avanzado en el capítulo anterior la descarga se realizará en JSON, formato muy adecuado para el almacenamiento de metadatos y de textos, justo el tipo de datos que son necesarios almacenar en este sistema.

El *Framework* elegido para el desarrollo de este módulo es Selenium[59], tal y como se describió en apartado de Extracción de información de , es la herramienta que se propone para desarrollar en Python los dos módulos de extracción, el de Facebook y el de Twitter.

Para ambos módulos es necesario indicar cuáles serán los identificadores de los perfiles que se van a descargar. Estos perfiles irán en un array de entrada y se irán procesando uno a uno.

I. Facebook *scraper*:

A continuación, se incluye código de ejemplo sobre la carga inicial de las librerías Selenium que hacen falta para descargar el perfil de FB y los metadatos que se necesitan. Tal y como se describió en la introducción a Selenium se va a utilizar la herramienta *WebDriver*. Esta herramienta facilita mucho las descargas. Con el método *get* del *driver*, directamente se posicionará en la página que le entre como parámetro y se podrán ir haciendo las cargas en variables de los diferentes elementos que se encuentren en dichas páginas. Es mejor siempre utilizar un usuario de FB logado.

```
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.common.by import By
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.keys import Keys

#Logado en FB
password = driver.find_element_by_name('session[password]')
password.send_keys('Your password', Keys.ENTER)

link = 'http://www.facebook.com/122595123700563'

#Carga un driver de Firefox
driver = webdriver.Firefox()
driver.get(link)
time.sleep(10)

#Descarga del título
```

```

title_name = WebDriverWait(driver, 20).until(EC.element_to_be_clickable((By.ID,
"pageTitle")))

#Descarga del texto del perfil
String Text = driver.findElement(By.id('Texto')).getText();

#Descarga del número de likes
likes = WebDriverWait(driver, 20).until(EC.element_to_be_clickable((By.XPATH,
"//div[@id='PagesProfileHomeSecondaryColumnPagelet']/descendant::div[contains
(@class,'clearfix')]/div[2]/div")))

```

De forma nativa, *webDriver* trabaja en JSON, ya que como se comentó JSON Wire Protocol se comunica entre el servidor y el cliente mediante la transferencia de datos en ese formato de datos. Por lo que finalmente habría que almacenar finalmente en un JSON toda la información descargada (idPerfil, idUsuario, lista de metadatos, texto).

II. Twitter *scraper*:

Seguidamente se incluye una porción de código para descargar los tweets (que constituyen el texto buscado) de un determinado perfil de Twitter.

```

from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.common.by import By
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.keys import Keys

#Driver de Chrome
driver = webdriver.Chrome()
driver.get("http://twitter.com/miPerfil")

#Autenticación
password = driver.find_element_by_name('session[password]')
password.send_keys('Your password', Keys.ENTER)

#Descarga de los tweets que representarán el texto.
tweets = driver.find_elements(By.XPATH, '//div[@data-testid="tweetText"]')
for i in tweets:
    texto = texto + (i.get_attribute('innerText'))

```

Igual que aparece aquí este fragmento de código, habría que hacer el desarrollo para cada metadato que se indicó en los requisitos.

Finalmente, como ocurría con el scraper de FB el resultado final será un JSON formado con toda la información descargada (idPerfil, idUsuario, lista de metadatos, texto). Este JSON se almacenará en la colección Perfiles.

5.4 Subsistema de procesamiento de lenguaje natural y análisis de sentimientos

Este subsistema contendrá tres módulos para trabajar sobre los ficheros textuales almacenados en JSON y descargados de las RRSS.

- I. Módulo de Preparación de los textos: Para poder trabajar con los textos descargados, tal y como se comentó anteriormente, es necesario su homogenización, es decir, un tratamiento previo para que todos tengan una apariencia similar en cuanto a su procesamiento. Para ello será necesario normalizarlos y tokenizarlos.

Para ello se utilizará NLTK[62]. Esta librería de Python es una de las favoritas de los desarrolladores de lingüística computacional por su flexibilidad y su potencia.

Se podrá importar en el Jupyter Notebook haciendo un simple:

```
import nltk
```

- a. Normalización: Consiste en que todos los textos tengan la misma apariencia y se realizará en función de las necesidades de PLN que se vaya a realizar.

- Inicialmente se limpiará el ruido que puedan tener los textos descargados (por ejemplo, palabras propias de las cabeceras de las RRSS de las que se haya realizado la descarga). Puede ser necesario utilizar funciones específicas para cada una de las redes sociales con las que se trabaje. Un ejemplo de limpieza de ruido eliminando todos los caracteres que no sean ASCII:

```
Texto_normalizado=
unicodedata.normalize('NFKD',word).encode('ascii','ignore').decode('utf
-8', 'ignore'))
```

- En la mayoría de las tareas de normalización, se pasan todas las palabras a mayúsculas o minúsculas, se eliminan los signos de puntuación o las palabras de unión, se pasan los números de cifra a palabras. Sin embargo, tal y como se explicó en el apartado de requisitos del sistema, este tipo de símbolos y su formato son necesarios para realizar la evaluación del perfilado indirecto de la personalidad en función del texto. Por lo tanto, no se va a realizar esa parte de la normalización.

- b. Tokenización: Separación en tokens

Realizando ya el sucesivo procesamiento y almacenamiento en la *Colección Perfiles*.

```
Texto_tokenizado = nltk.word_tokenize(texto)
```

- II. Módulo de cálculo de indicadores textuales ENCUIST: Sobre los JSON normalizados y tokenizados se continua con el cálculo de los indicadores textuales.

Para ello se va a utilizar la herramienta LIWC[75], el enlace que se hace mención en la referencia [75] es directamente el de la librería de Github en Python. Sin embargo, tal y como se comentó anteriormente, el diccionario en español tendrá que ser solicitado al profesor [James W. Pennebaker](#) para su utilización para fines académicos.

Después de realizar el estudio comparativo de las librerías más conocidas de PLN que existen en el mercado, la forma más rápida de obtener la funcionalidad que se está buscando para calcular los indicadores, es esta librería junto con su lexicón en castellano.

Una vez calculados los indicadores del texto, se almacenarán en la *Colección Perfiles de ENCUIST*.

- III. Módulo de análisis de sentimientos: Tal y como se expuso en el estudio de la funcionalidad del análisis de sentimientos (Análisis de sentimientos) NLTK es una

herramienta de Python muy valorada para el análisis de sentimientos a través de su librería `SentimentIntensityAnalyzer`.

Previamente es necesario partir del texto tokenizado, pero esta labor ya ha sido realizada en el primer módulo de tratamiento de los textos.

Para poder trabajar con esta librería sería necesario importarla dentro de Jupyter.

```
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from nltk import sentiment
```

Existe mucha documentación sobre su uso, como por ejemplo en <https://www.arsys.es/blog/analisis-sentimientos-python-jupyter-notebooks>. [27].

Una vez calculado el resultado será incluir en la *Colección Sentimientos* junto con el *idPerfil* y el *idUsuario*.

5.5 Subsistema de análisis de rasgos ENCUIST.

Este es el subsistema que realizará la parte de predicción de cada rasgo ENCUIST en base a los textos analizados.

Accederá a la colección de perfiles de ENCUIST para utilizar los indicadores por perfil que se han almacenado tras los cálculos de los sistemas anteriores.

El resultado esperado de este subsistema son los rasgos ENCUIST. Este tipo de resultados será el introducido en el sistema como parte del vector de aprendizaje junto con los textos de RRSS. En el test se puede observar que los rasgos aparecen numéricamente clasificados en ALTO, MEDIO y BAJO e implícitamente un valor de NULO si el resultado numérico está por debajo del umbral. Es decir, a pesar de que los acompaña un valor numérico resultado de la evaluación de los ítems, este valor sirve únicamente para clasificar el resultado en la escala de clases de NULO, ALTO, MEDIO y BAJO. La palabra clave es, por tanto, clasificación.

Lo normal cuando se habla de Machine Learning es utilizar varios modelos para ver qué resultado da cada uno de ellos y escoger los resultados que más se ajusten a lo esperado. Además, en este proyecto se pueden dar enfoques a la solución del problema:

1. **Modelo más conservador: Algoritmo de clasificación supervisado.** Características de esta implementación.
 - a. Cada rasgo que se va a evaluar tendrá su propio modelo o modelos. De esta forma se gana independencia si a futuro se quieren añadir o eliminar rasgos.
 - b. Como ya se ha explicado en el diseño será necesario aplicar inicialmente un entrenamiento con los vectores de ejemplo compuestos de: [Indicadores, Rasgo_n] siendo n el número de rasgo que se está calculando.
 - c. Se entenderá que cada indicador léxico es independiente del resto de indicadores. Tener cantidad de información no supone un problema porque se ha elegido un BigData NoSQL, por lo que es recomendable, como se ha mencionado, utilizar varios algoritmos y tener almacenados los resultados del rasgo en función del algoritmo.
Se necesitará, entonces, una revisión por parte de expertos psicólogos del resultado que mejor se ajuste y se recomienda realizar una batería de pruebas para esta labor.
 - d. Algoritmos que se utilizarían en esta propuesta:
 - i. **Random Forest:**

La principal implementación de árboles de decisión en Python está disponible en la librería scikit-learn a través de las clases `DecisionTreeClassifier` y `DecisionTreeRegressor`.

Algunos de los hiperparámetros que se puede ajustar para configurar el bosque resultante son los siguientes:

- **Criterion:** La función para medir la calidad de la división. Uno de los valores más usados es **Gini** que mide la varianza total entre las clases del nodo, es decir, la pureza del nodo. El bosque aleatorio intenta maximizar la ganancia de información en cada nodo. La ganancia de información se define como la disminución de la entropía, en el caso de random forest una disminución de la entropía se entiende como un aumento de la pureza del nodo. Cuanto mayor sea la pureza del nodo, menor el valor del índice Gini.
- **max_depth:** La profundidad máxima del árbol. Si no se establece valor, entonces los nodos se expanden hasta que todas las hojas son puras.
- **min_samples_split:** El número mínimo de muestras requeridas para dividir un nodo interno.
- **random_state:** La semilla aleatoria utilizada para generar los subconjuntos aleatorios de entidades y datos.
- **n_estimators:** El número de árboles de decisión en el bosque aleatorio.
- **max_features:** El número máximo de características o features a considerar al buscar la mejor división.

Los pasos para programar un random forest serán los siguientes:

1. **Carga de las bibliotecas requeridas:** El primer paso es cargar las bibliotecas requeridas. En este caso el clasificador de bosque aleatorio de scikit-learn.
2. **Importar el conjunto de datos:** En este caso serán los indicadores léxicos indicados en la Tabla 3-1 Indicadores ENCUIST de personalidad. El objetivo es predecir el valor del rasgo_n que se esté calculando en este modelo.
3. **Dividir el conjunto de datos en conjuntos de entrenamiento y prueba:** se dividirá el conjunto de datos en conjuntos de entrenamiento y prueba. Se usará, en este caso, el 70% de los datos para entrenamiento y el 30% para pruebas.
4. **Entrenamiento del modelo en el conjunto de entrenamiento:** A continuación, Se entrenará el clasificador de bosque aleatorio con el conjunto de entrenamiento.
5. **Hacer predicciones sobre el conjunto de prueba:** Finalmente, se harán predicciones sobre el conjunto de prueba y se evaluará la precisión del modelo.
6. **Ajuste de hiperparámetros:** una vez que se tiene un modelo básico funcionando, se puede mejorar su rendimiento ajustando los hiperparámetros.

```
from sklearn.ensemble import RandomForestClassifier

# Creación de una instancia del Clasificador Random Forest
forest = RandomForestClassifier(criterion='gini',
                               n_estimators=5,
                               random_state=1,
                               n_jobs=2)
```

ii. Naïve Bayes:

Tal y como se explicó anteriormente, este algoritmo es de clasificación supervisado está basado en el teorema de Bayes, es decir, es un algoritmo probabilístico. El teorema de Bayes se usa para encontrar la probabilidad de una hipótesis con evidencia dada.

En general hay dos principales implementaciones de Naïve Bayes: Naïve Bayes Gaussiana y Naïve Bayes Multinomial. En este caso se va a proponer trabajar con el primero.

Naïve Bayes Gaussiano asume que los datos siguen una distribución Gaussiana. Lo que hace es suponer una distribución normal para cada variable y clase dado el promedio y desviación típica de los datos obtenidos. En el caso de los rasgos que estamos calculando las clases son cuatro posibles (NULO, BAJO, MEDIO, ALTO). Cada vez que venga un valor de una variable se calculará la probabilidad de que ese valor venga en cada una de las distribuciones. Repetiremos este proceso para cada una de las variables y obtendremos la probabilidad final de que esa observación pertenezca a cada grupo. Por último, aquella clase que obtenga mayor probabilidad será la predicción del algoritmo.

Al igual que con Random Forest se utilizarán ciertos datos para el entrenamiento y otros se dejarán reservados para el test de prueba.

Para el entrenamiento se fijarán los datos con la función *fit()* y para la predicción del valor que se está buscando *predict()*.

Este algoritmo cuenta sólo con dos hiperparámetros pero en general no suele ser necesario tocarlos. Si es necesario, tras realizar los test se pueden ajustar:

- *priorsarray-like* of shape (n_classes): Probabilidades previas de clase.
- *var_smoothingfloat*. Varianza mayor de todas las características que se agrega a las varianzas para la estabilidad del cálculo.

```
from sklearn.naive_bayes import GaussianNB

# Creación de una instancia del Clasificador Naive Bayes Gaussiano
gnb = GaussianNB()
```

A continuación, como ejemplo, se incluye el código de cómo se realizaría el entrenamiento y test de este algoritmo. Los indicadores de entrada irían en el vector *x_test_indicadores* del conjunto de validación se comparan las

clases obtenidas contra los valores correctos de *y_test_rasgo*. El porcentaje de aciertos es la precisión del clasificador.

```
# fit para entrenar al algoritmo
gnb.fit (X_train_indicadores, y_train_rasgo)
tot_Test = np.size(y_test_rasgo)
tot_aciertos = 0
y_pred_rasgo = []
# zip es una función de Python que nos permite recorrer
# for simultáneamente dos listas X_test_indicadores e y_test_rasgo
for x, y in zip (X_test_indicadores, y_test_rasgo):
    y_est = gnb.predict([x])
    y_pred_rasgo.append(y_est)
    if (y==y_est): tot_aciertos +=1
print('Exactitud del clasificador (porcentaje de aciertos) = %s \n'
      %(tot_aciertos*100/tot_Test))
```

Por pantalla pintaría la frase “Exactitud del clasificador (porcentaje de aciertos)” seguido del tanto por ciento de aciertos.

e. El resultado de ambos algoritmos será almacenado en MongoDB en la *Colección Rasgos*

2. Modelo más innovador: Algoritmo de Deep Learning. Hasta ahora se han descrito soluciones de ML basado en el modelado clásico. Inicialmente los textos han sido procesados acompañándolos de un conteo de palabras, especialmente las “palabras pequeñas” del texto, a través del procesamiento realizado con LIWC. Sin embargo, hay otra aproximación para el cálculo de los rasgos buscados basados en Deep learning. Mientras que los algoritmos tradicionales son lineales, los modelos de aprendizaje profundo, basados generalmente en redes neuronales, se apilan en una jerarquía de complejidad y abstracción crecientes.

Después de pasar el módulo de preparación de los textos (tareas de normalización y tokenización), **no se ejecutaría el módulo de cálculo de indicadores textuales**. Como entrenamiento se seguiría incluyendo el mismo conjunto de pruebas, pero como entrada del algoritmo introduciríamos simplemente los textos preparados para que el algoritmo aprendiera directamente de ellos y ver que conclusiones extrae.

Tal y como se comentó, las dos librerías más importantes desarrolladas para la implementación de modelos de DL con Python son PyTorch[43] y TensorFlow[69].

Sin embargo, mucho más sencillo y eficiente que desarrollar una red neuronal partiendo de cero es basarse en un modelo ya implementado como RoBERTa[44]. Su uso se puede encontrar claramente descrito en la propia página de PyTorch [36]. Existen múltiples implementaciones de RoBERTa ya entradas en castellano. Una de ellas es **SpanBERTa[44]: RoBERTa for Spanish**. Se trata de una implementación del transformer RoBERTa ya entrenado en el corpus OSCAR[38] en castellano. OSCAR es un proyecto open source que crea un corpus en 166 lenguas, entre ellas castellano y que sirve correctamente como corpus de entrenamiento de múltiples modelos. Ya con el

modelo preentrenado se puede realizar el FineTuning para realizar la tarea de clasificación que se necesita en nuestro proyecto, acompañando cada texto con el valor del rasgo por perfil que se tiene en el set de entrenamiento. Junto con el código de SpanBERTa, vienen también las librerías necesarias para hacer el FineTuning.

6 CONCLUSIONES Y LÍNEAS FUTURAS

6.1 Conclusiones

Una parte muy importante de investigación previa al desarrollo del presente trabajo fue encontrar un modelo de personalidad que permitiera realizar un perfilado indirecto. ENCUIST, y todos los trabajos publicados sobre este modelo, permite identificar los indicadores necesarios para hacer una perfilación indirecta y poderla automatizar mediante un sistema de información.

A lo largo de este documento se ha planteado cuál ha sido la investigación llevada a cabo para poder hacer una propuesta de diseño del sistema. Inicialmente ha sido indispensable la profundización en el conocimiento experto para entender las necesidades y el carácter de los datos que alimentarían el sistema. Después, las posibilidades tecnológicas que plantea el mercado, para elegir aquellas que pueden dar una respuesta mejor a los requisitos elaborados. Los indicadores que se seleccionaron han sido básicamente textuales, por eso, el ML y concreto el Deep Learning tienen una aplicación directa en este proyecto.

El campo del PLN ha evolucionado enormemente a lo largo de los últimos años y hay múltiples herramientas aplicables y en continua evolución. Por eso, este proyecto plantea una fórmula tradicional basada en algoritmos de ML habitualmente utilizados, pero a la vez abierta para poder evolucionar, con una BBDD NoSQL que almacenaría JSONs y que sería capaz de adaptar rápidamente su modelo sin tener que hacer grandes cambios de diseño. Y un segundo planteamiento, también con ese almacenamiento, pero basado en Deep Learning, con uso de *Transformers*, herramientas punteras en el PLN.

Y es que, el mundo digital evoluciona a una velocidad pasmosa, siendo el motor de múltiples negocios y campos de conocimiento. Los sistemas deben estar diseñados de forma flexible, para que no tengan que ser desechados al poco tiempo de ser implantados. Deben ser capaces de asumir los cambios en los datos de forma fácil y dinámica. Estos datos de entrada evolucionan, crecen y cambian de forma continua. Y la imaginación y necesidades de los usuarios con ellos. Por eso, los sistemas de información deben tener esto en cuenta a la hora de ser diseñados.

6.2 Líneas futuras

El diseño que ha descrito el presente trabajo es la base para desarrollar el software que implemente el sistema. El siguiente paso para comenzar con el desarrollo será la obtención de datos.

Obtener datos válidos para realizar pruebas para el presente trabajo académico resultó muy complicado debido a dos condicionantes:

- Lentitud por parte de los organismos externos en responder a las solicitudes sobre los datos. El tiempo de respuesta ha sido demasiado largo y superó los límites planteados para la realización de este trabajo.
- Los datos a los que se podía tener acceso son de estricto uso interno a la organización y no es posible su utilización para otros fines que no sea un desarrollo propio.

El desarrollo se llevará a cabo internamente con todas las medidas de seguridad sobre los datos necesarias que garanticen el cumplimiento de normas y leyes sobre protección de datos personales, que como ya es sabido son muy complejas y se necesitará el asesoramiento del Departamento Jurídico para su manejo.

La aplicación del sistema una vez desarrollado puede ser varia. Por ejemplo:

- Investigaciones para encontrar determinados perfiles dentro de un grupo. Por ejemplo, identificación de individuos que cumplan características para ser buenas fuentes; señalamiento del posible líder dentro de un conjunto de perfiles; etc.
- Como la BBDD y el tipo de datos almacenados con los que se ha diseñado el sistema son bastante abiertos para crecer, se podría, a futuro, alimentar el sistema con más información, como, por ejemplo, datos de antecedentes penales o tipos de delitos cometidos por los individuos. El sistema podría entrenarse con los perfiles de determinadas personas y sus delitos, para así poder identificar nuevos posibles perfiles de RRSS de individuos susceptibles de cometer delitos.
- RRHH puede utilizar el sistema para analizar los perfiles de los posibles candidatos y ver si su perfil inferido coincide con las necesidades de la organización.

Como se puede ver, el sistema planteado se presta muy versátil y podría ser utilizado para múltiples fines, dependiendo estos principalmente de los datos que se elijan para alimentar el sistema.

7 BIBLIOGRAFÍA

- [1] Francisco Javier Torregrosa López y Rafael Manuel López.
Personalidad y redes sociales: una revisión sistemática, Research Gate Genios, 2016.
- [2] Patricia González Élices.
Perfilación indirecta a través de la comunicación verbal y no verbal.
Tesis Doctoral. Universidad Autónoma de Madrid.
- [3] Lucía Halty, José Luis González, Andrés Sotoca.
Modelo ENCUIST: aplicación al perfilado criminal.
Anuario de Psicología Jurídica, vol. 27, pp. 21-31, 2017. Colegio Oficial de la Psicología de Madrid
- [4] «ENCUIST»
Modelo de personalidad para la realización de perfiles psicológicos
<https://encuist.com/>
[Último acceso: septiembre 2022]
- [5] «Psicología y mente».
<https://psicologiaymente.com>.
[Último acceso: septiembre 2022].
- [6] «Área Humana, investigación, innovación y experiencia en psicología».
<https://www.areahumana.es/que-es-la-personalidad/>
[Último acceso: diciembre 2022].
- [7] «Atlassian»
<https://www.atlassian.com/es/agile/>
[Último acceso: septiembre 2022].
- [8] «Instituto de ingeniería del conocimiento. IIC»
<https://www.iic.uam.es/>
[Último acceso: septiembre 2022]
- [9] «Beta OpenAI»
<https://beta.openai.com/docs/models/gpt-3>
[Último acceso: octubre 2022]
- [10] María de Vicente Guijarro.
Relación entre redes sociales y personalidad.
Drafts of economic intelligence. Volumen 1 nº 3, pp. 23-34. La SEI. UAM.

- [11] «Psicología y Mente»
<https://psicologiaymente.com/personalidad/teoria-personalidad-eysenck-modelo-pen>
[Último acceso: octubre 2022]
- [12] «Investigación y Ciencia»
<https://www.investigacionyciencia.es/revistas/mente-y-cerebro/deconstruccion-de-la-memoria-509/eres-lo-que-hablas-8301>
[Último acceso: diciembre 2022]
- [13] José Rúas, Mercedes Fernández, Iván Puentes.
Aplicación de la herramienta LIWC al análisis del discurso político.
Los mítines de los candidatos en las elecciones al Parlamento de Galicia de 2012.
Simposio Internacional sobre Política Científica en Comunicación.
Vol. 1, 2013 (Comunicaciones 1), ISBN 978-84-616-4124-6, págs. 47-64
- [14] «Federal Bureau of Investigation. FBI»
Reading People: Behavioral Anomalies and Investigative Interviewing
<https://leb.fbi.gov/articles/featured-articles/the-language-of-psychopaths-new-findings-and-implications-for-law-enforcement>
[Último acceso: diciembre 2022]
- [15] «Federal Bureau of Investigation. FBI»
The Language of Psychopaths
New Findings and Implications for Law Enforcement
<https://leb.fbi.gov/articles/featured-articles/the-language-of-psychopaths-new-findings-and-implications-for-law-enforcement>
[Último acceso: diciembre 2022]
- [16] «Psikipedia»
<https://psikipedia.com/libro/personalidad/2345-modelos-teoricos>
[Último acceso: octubre 2022]
- [17] BBC News World
5 claves para entender el escándalo de Cambridge Analytica
<https://www.bbc.com/mundo/noticias-43472797>
[Último acceso: diciembre 2022]
- [18] «LIWC»
<https://www.liwc.app/>
[Último acceso: octubre 2022]

- [19] María del Pilar Salas Zárate.
Detección de Patrones Psicolingüísticos para el Análisis de lenguaje Subjetivo en español.
Tesis doctoral. Universidad de Murcia.
- [20] «iabspain»
<https://iabspain.es/categoria-de-estudio/redes-sociales/>
[Último acceso: octubre 2022]
- [21] Simon Thompson.
Managing Machine Learning Projects From design to deployment Version 4.
Manning Publications
- [22] Carl Anderson.
Creating a Data-Driven Organization.
O'Reilly
- [23] Benjamin Bengfort, Rebecca Bilbro and Tony Ojeda.
Applied Text Analysis with Pythoon.
O'Reilly
- [24] Mikel Vallejo del Moral.
Algoritmo Deep Q-Learning para el aprendizaje por refuerzo de una estrategia de conducción en 2D.
Director: Miguel Pagola Barrio.
Trabajo de fin de grado. UPNA.
- [25] «incibe»
https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_ciberseguridad_almacenamiento_seguro_metad.pdf
[Último acceso noviembre 2022]
- [26] «Bigdata-analytics»
<https://bigdata-analytics.es/>
[Último acceso noviembre 2022]
- [27] «Openwebinars»
<https://openwebinars.net/blog/ventajas-y-desventajas-de-mongodb/>
[Último acceso noviembre 2022]
- [28] «Github MongoDB»
<https://github.com/mongodb/mongo>

- [Último acceso noviembre 2022]
- [29] «Grapheverywhere»
<https://www.grapheverywhere.com/10-razones-para-elegir-neo4j/>
[Último acceso noviembre 2022]
- [30] «Neo4j python»
<https://neo4j.com/developer/python/>
[Último acceso noviembre 2022]
- [31] «Jupyter»
<https://jupyter.org/>
[Último acceso noviembre 2022]
- [32] «Neo4j mongodb»
<https://neo4j.com/developer/mongodb/>
[Último acceso noviembre 2022]
- [33] «Medium Tutorial Python»
<https://medium.com/datos-y-ciencia/preprocesamiento-de-datos-de-texto-un-tutorial-en-python-5db5620f1767>
[Último acceso noviembre 2022]
- [34] «Arsys»
<https://www.arsys.es/blog/analisis-sentimientos-python-jupyter-notebooks>
[Último acceso noviembre 2022]
- [35] «Ciencia del dato»
https://www.cienciadedatos.net/documentos/py07_arboles_decision_python.html
[Último acceso noviembre 2022]
- [36] «Scikit-learn»
<https://scikit-learn.org/stable/modules>
[Último acceso noviembre 2022]
- [37] «SimpliLearn»
<https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-with-python>
[Último acceso noviembre 2022]
- [38] «Towards data science»
<https://towardsdatascience.com/deep-learning-with-python-neural-networks-complete-tutorial-6b53c0b06af0>

- [Último acceso noviembre 2022]
- [39] «iic Instituto de Ingenieria del conocimiento»
<https://www.iic.uam.es/innovacion/transformers-en-procesamiento-del-lenguaje-natural/>
 [Último acceso diciembre 2022]
- [40] «Cornell University»
 Attention is all you need <https://arxiv.org/abs/1706.03762>
 [Último acceso diciembre 2022]
- [41] Guillem García Subies.
 Modelos de Transformers para la Clasificación de Texto.
 Trabajo Fin de máster. Universidad Politécnica de Madrid.
- [42] «MetaAi»
 RoBERTa: An optimized method for pretraining self-supervised NLP systems.
<https://ai.facebook.com/blog/roberta-an-optimized-method-for-pretraining-self-supervised-nlp-systems/>
 [Último acceso diciembre 2022]
- [43] «PyTorch»
https://pytorch.org/hub/pytorch_fairseq_roberta/
 [Último acceso diciembre 2022]
- [44] «GitHub Spanish BERT»
 SpanBERTa: RoBERTa for Spanish
<https://github.com/chriskhanhtran/spanish-bert>
 [Último acceso diciembre 2022]
- [45] «OSCAR Project»
<https://oscar-project.org/>
 [Último acceso diciembre 2022]
- [46] Odekerken-Schroder, K. D. Wulf, and P. Schumacher
 Strengthening outcomes of retailer-consumer relationships.
 The dual impact of relationship marketing tactics and consumer personality.
 Journal of Business Research.
- [47] Correa, T., Hinsley, A. W., & De Zúñiga, H. G.
 Who interacts on the Web? The intersection of users' personality and social media use.
 Computers in Human Behavior

- [48] Tosun, L. P., & Lajunen, T.
Does Internet use reflect your personality?
Relationship between Eysenck's personality dimensions and Internet use.
Computers in Human Behavior
- [49] Golbeck, J., Robles, C., Edmondson, M., & Turner, K.
Predicting personality from twitter.
Privacy, Security, Risk and Trust (PASSAT) and
2011 IEEE Third International Conference on Social Computing (SocialCom)
- [50] Amichai-Hamburger, Y. & Vinitzky, G.
Social network use and personality.
Computers in Human Behavior. 26, 1289-1295.
- [51] Quercia, D., Kosinski, M., Stillwell, D., & Crowcroft, J.
Our twitter profiles, our selves: Predicting personality with twitter.
Privacy, Security, Risk and Trust (PASSAT) and
2011 IEEE Third International Conference on Social Computing (SocialCom)
- [52] Bachrach, Y., Kosinski, M., Graepel, T., Kohli, P., & Stillwell, D.
Personality and patterns of Facebook usage.
Proceedings of the 4th Annual ACM Web Science Conference
- [53] Gosling, S. D., Augustine, A. A., Vazire, S., Holtzman, N., & Gaddis, S.
Manifestations of personality in online social networks:
Self-reported Facebook-related behaviors and observable profile information.
Cyberpsychology, Behavior, and Social Networking
- [54] Ross, C., Orr, E. S., Sisic, M., Arseneault, J. M., Simmering, M. G., & Orr, R. R.
Personality and motivations associated with Facebook use.
Computers in human behavior
- [55] Ehrenberg, A., Juckes, S., White, K. M., & Walsh, S. P.
Personality and self-esteem as predictors of young people's technology use.
Cyberpsychology & behavior
- [56] Jenkins-Guarnieri, M. A., Wright, S. L., & Hudiburgh, L. M.
The relationships among attachment style, personality traits,
interpersonal competency, and Facebook use.
Journal of Applied Developmental Psychology.
- [57] Ross, C., Orr, E. S., Sisic, M., Arseneault, J. M., Simmering, M. G., & Orr, R. R.
Personality and motivations associated with Facebook use.
Computers in human behavior.

- [58] Ferwerda, B., Schedl, M., & Tkalcic, M. (2015, September).
Predicting personality traits with instagram pictures.
3rd Workshop on Emotions and Personality in Personalized Systems
- [59] «Selenium»
<https://www.selenium.dev/>
[Último acceso diciembre 2022]
- [60] «Eleutherai»
<https://www.eleuther.ai/>
[Último acceso diciembre 2022]
- [61] «nlpcloud»
<http://nlpcloud.com>
[Último acceso diciembre 2022]
- [62] «NLTK: Natural Language Toolkit»
<https://www.nltk.org/>
[Último acceso diciembre 2022]
- [63] «spaCy»
<https://spacy.io/>
[Último acceso diciembre 2022]
- [64] «openNLP»
<https://opennlp.apache.org/>
[Último acceso diciembre 2022]
- [65] «coreNLP»
<https://stanfordnlp.github.io/CoreNLP/>
[Último acceso diciembre 2022]
- [66] «allenNLP»
<http://docs.allennlp.org/v0.9.0/index.html>
[Último acceso diciembre 2022]
- [67] «Spark NLP»
<https://nlp.johnsnowlabs.com/docs/en/install>
[Último acceso diciembre 2022]
- [68] «gensim 4.3.0»
<https://pypi.org/project/gensim/>
[Último acceso diciembre 2022]

- [69] «TensorFlow»
<https://www.tensorflow.org/>
[Último acceso diciembre 2022]
- [70] «HumanLevel. BERT»
<https://www.humanlevel.com/articulos/posicionamiento-natural-busadores/google-bert-actualizacion-para-entender-el-lenguaje-natural.html>
[Último acceso diciembre 2022]
- [71] «Xataka. BERT»
<https://www.xataka.com/servicios/como-funciona-bert-inteligencia-artificial-que-google-quiere-conseguir-que-su-motor-busqueda-nos-entienda-mejor>
[Último acceso diciembre 2022]
- [72] «Atlassian»
<https://www.atlassian.com/es/agile/project-management/project-management-intro>
[Último acceso diciembre 2022]
- [73] «Python»
<https://www.python.org/>
[Último acceso diciembre 2022]
- [74] «Python»
<https://jupyter.org/>
[Último acceso diciembre 2022]
- [75] «Github. LIWC»
<https://github.com/chbrown/liwc-python>
[Último acceso diciembre 2022]
- [76] Neo4J: trabajando con grafos
<https://www.paradigmadigital.com/dev/neo4j-trabajando-grafos/>
[Último acceso diciembre 2022]
- [77] Primeros pasos con Neo4j
<https://www.adictosaltrabajo.com/2014/07/07/neo4j-first-steps/>
[Último acceso diciembre 2022]

[78] BD de grafos

<https://www.grapheverywhere.com/indicios-bd-grafos/>

[Último acceso diciembre 2022]

ANEXO I: TEST ENCUIST

ENCUIST 3.0 (Halty, González y Sotoca, 2017)



A continuación aparecen una serie de preguntas sobre rasgos de personalidad que te pueden describir en mayor o menor medida. Señala en qué medida estás de acuerdo con cada afirmación. No hay respuestas buenas ni malas. Los datos de este cuestionario son absolutamente confidenciales y sólo serán utilizados con fines de investigación.

- A.- Edad: _____
- B.- Sexo: Hombre ____ Mujer ____
- C.- Nacionalidad: _____
- D.- País de Residencia _____
- E.- Nivel educativo:
- No escolarizado ____
 - Primaria ____
 - ESO/EGB ____
 - Bachillerato/BUP ____
 - Formación profesional ____
 - Grado/Diplomado/Licenciado ____
 - Postgrado ____
- F.- ¿Has estado alguna vez metido en peleas? Si ____ No ____
- G.- ¿Has estado alguna vez detenido o en prisión (en adultos o menores)? Si ____ No ____
- H.- En general, ¿sueles mantener relaciones de pareja estables (más de dos años)? Si ____ No ____
- I.- ¿Has sufrido algún tipo de maltrato físico, sexual o psicológico? Si ____ No ____

Me considero una persona... (señala una opción de cada línea)

1.- Introvertida	Extrovertida
2.- Empática	Fria/distante emocionalmente
3.- Ansiosa/preocupadiza	Estable emocionalmente
4.- Impulsiva	Reflexiva
5.- Irascible	Calmada
6.- Buscadora de sensaciones	Tranquila

1.- Indique el grado de acuerdo en las siguientes afirmaciones

1	2	3	4
Muy en desacuerdo	Desacuerdo	De acuerdo	Muy de acuerdo
1.- Me gustaría la sensación de bajar esquiando muy rápido por la pendiente de una gran montaña			1 2 3 4
2.- He probado o me gustaría probar alguna droga que produzca alucinaciones			1 2 3 4
3.- Me gusta probar comidas nuevas que no he probado antes			1 2 3 4
4.- Me gusta la rutina			1 2 3 4
5.- Me gustaría probar a lanzarme en paracaídas			1 2 3 4
6.- Me gusta tener sensaciones corporales excitantes			1 2 3 4
7.- Me gustaría realizar actividades que impliquen algo de riesgo			1 2 3 4
8.- Me gusta explorar una ciudad o un barrio desconocido aunque pueda perderme			1 2 3 4
9.- Me aburren las cosas con facilidad			1 2 3 4
10.- No tengo paciencia con las personas grises o aburridas			1 2 3 4
11.- Disfruto mucho hablando con la gente			1 2 3 4
12.- Soy una persona muy activa			1 2 3 4
13.- Disfruto de las fiestas en las que hay mucha gente			1 2 3 4
14.- Huyo de las multitudes			1 2 3 4

ENCUESTA 3.0 (Halty, González y Sotoca, 2017)

2.- Indique como se siente usted en general en la mayoría de las ocasiones

0	1	2	3
Casi nunca	A veces	A menudo	Casi siempre
15.- Me siento bien	0	1	2 3
16.- Me canso rápidamente	0	1	2 3
17.- Siento ganas de llorar	0	1	2 3
18.- Me gustaría ser tan feliz como otros	0	1	2 3
19.- Pierdo oportunidades por no decidirme pronto	0	1	2 3
20.- Me siento descansado	0	1	2 3
21.- Me siento una persona tranquila, serena y sosegada	0	1	2 3
22.- Veo que las dificultades se amontonan y no puedo con ellas	0	1	2 3
23.- Me preocupo demasiado por las cosas sin importancia	0	1	2 3
24.- Soy feliz	0	1	2 3
25.- Suelo tomar las cosas demasiado seriamente	0	1	2 3
26.- Me falta confianza en mi mismo	0	1	2 3
27.- Me siento seguro	0	1	2 3
28.- No suelo afrontar las crisis o dificultades	0	1	2 3
29.- Me siento triste (melancólico)	0	1	2 3
30.- Estoy satisfecho	0	1	2 3
31.- Me rondan y molestan pensamientos sin importancia	0	1	2 3
32.- Me afectan tanto los engaños que no puedo olvidarlos	0	1	2 3
33.- Soy una persona estable	0	1	2 3
34.- Cuando pienso sobre asuntos y preocupaciones actuales me pongo tenso y agitado	0	1	2 3

3.- Indique el grado de acuerdo en las siguientes afirmaciones

1	2	3	4
Muy en desacuerdo	Desacuerdo	De acuerdo	Muy de acuerdo
35.- Cuando pienso que algo desagradable va a suceder, por lo general suelo estar muy agitado e intranquilo	1	2	3 4
36.- Me preocupa cometer errores	1	2	3 4
37.- Cuando quiero algo, por lo general hago todo lo posible para conseguirlo	1	2	3 4
38.- La mayoría de las veces hago cosas con el único motivo de divertirme	1	2	3 4
39.- Las críticas o una reprimenda me desaniman mucho	1	2	3 4
40.- Cuando consigo a algo que quiero, me siento emocionado y lleno de energía	1	2	3 4
41.- Hago un gran esfuerzo para conseguir las cosas que quiero	1	2	3 4
42.- Me gusta sentir emociones fuertes y sensaciones nuevas	1	2	3 4
43.- Me siento muy molesto cuando pienso o sé que alguien está enfadado conmigo	1	2	3 4
44.- Siempre estoy dispuesto a probar algo nuevo si creo que va a ser divertido	1	2	3 4
45.- Cuando hago algo bien, quiero continuarlo	1	2	3 4
46.- Incluso si algo malo me fuera a suceder, rara vez experimento miedo o nerviosismo	1	2	3 4
47.- Por lo general actúo según el momento me inspira a hacerlo	1	2	3 4
48.- Cuando veo la oportunidad de conseguir algo que quiero, la aprovecho de inmediato	1	2	3 4
49.- Cuando me sucede algo bueno, me afecta profundamente	1	2	3 4
50.- Me siento preocupado cuando pienso que no he hecho un buen trabajo	1	2	3 4
51.- Me sentiría emocionado si ganara una competición	1	2	3 4
52.- En comparación con mis amigos tengo muy pocos miedos o temores	1	2	3 4
53.- Si veo una oportunidad de conseguir algo que me gusta, me siento entusiasmado	1	2	3 4
54.- Cuando trabajo para conseguir algo, me entrego por completo a ello	1	2	3 4

ENCUESTA 3.0 (Halty, González y Sotoca, 2017)

4.- Indique el grado en el que le describen las siguientes afirmaciones

0	1	2	3
No es cierto	Un poco cierto	Muy cierto	Definitivamente cierto
55.- Expreso mis sentimientos abiertamente	0	1	2 3
56.- Lo que pienso que está “bien” o está “mal” es diferente a lo que piensan otras personas	0	1	2 3
57.- Me preocupo por mi trabajo y/o mis estudios	0	1	2 3
58.- No me importa a quién le haga daño para obtener lo que quiero	0	1	2 3
59.- Me siento mal o culpable cuando hago algo malo	0	1	2 3
60.- No muestro mis emociones a los demás	0	1	2 3
61.- No me preocupa ser puntual	0	1	2 3
62.- Me preocupan los sentimientos de los demás	0	1	2 3
63.- No me importa si me meto en líos o problemas	0	1	2 3
64.- No dejo que mis emociones me controlen	0	1	2 3
65.- Me resulta indiferente hacer las cosas bien	0	1	2 3
66.- Parezco una persona fría y con falta de interés por los demás	0	1	2 3
67.- Admito fácilmente que estoy equivocado	0	1	2 3
68.- A los demás les resulta fácil decir cómo me siento	0	1	2 3
69.- Siempre trato de hacer las cosas lo mejor posible	0	1	2 3
70.- Pido perdón a quienes he hecho daño	0	1	2 3
71.- Trato de no herir los sentimientos de los demás	0	1	2 3
72.- No siento remordimientos cuando hago algo malo	0	1	2 3
73.- Soy muy expresivo y emocional	0	1	2 3
74.- No me gusta dedicar mucho tiempo a hacer bien las cosas	0	1	2 3
75.- No me importan los sentimientos de los demás	0	1	2 3
76.- Escondido mis sentimientos a los demás	0	1	2 3
77.- Me esfuerzo mucho en todo lo que hago	0	1	2 3
78.- Hago cosas para que los demás se sientan bien	0	1	2 3

5.- A continuación aparecen una serie de afirmaciones que la gente usa para describirse a sí misma. Indique cómo se siente normalmente en la mayoría de las ocasiones.

1	2	3	4
Casi nunca	Algunas veces	A menudo	Casi siempre
79.- Me caliento rápidamente	1	2	3 4
80.- Tengo un carácter irritable	1	2	3 4
81.- Soy una persona exaltada	1	2	3 4
82.- Me molesta cuando hago algo bien y no me lo reconocen	1	2	3 4
83.- Tiendo a perder los estribos	1	2	3 4
84.- Me pone furioso que me critiquen delante de los demás	1	2	3 4
85.- Me siento furioso cuando hago un buen trabajo y se me valora poco	1	2	3 4
86.- Me cabreo con facilidad	1	2	3 4
87.- Me enfado si no me salen las cosas como tenía previsto	1	2	3 4
88.- Me enfado cuando se me trata injustamente	1	2	3 4

ENCUIST 3.0 (Halty, González y Sotoca, 2017)



6.- Lea las siguientes afirmaciones y marque aquella casilla que mejor describa CÓMO REACCIONA O SE COMPORTA CUANDO ESTÁ ENFADADO O FURIOSO

1	2	3	4
No, en absoluto	Algo	Moderadamente	Mucho
89.- Expreso mi ira			1 2 3 4
90.- Hago comentarios irónicos de los demás			1 2 3 4
91.- Hago cosas como dar portazos			1 2 3 4
92.- Discuto con los demás			1 2 3 4
93.- Digo barbaridades			1 2 3 4
94.- Pierdo la paciencia			1 2 3 4

7.- Lea las siguientes afirmaciones y marque aquella que mejor le describa

1	2	3	4	5
Nunca	Raras veces	Algunas veces	Muchas veces	Siempre
95.- Evito las cosas que me dan asco				1 2 3 4 5
96.- Cuando siento repugnancia por algo, me preocupa que pueda desmayarme				1 2 3 4 5
97.- Cuando siento náuseas me asusto				1 2 3 4 5
98.- Creo que las cosas asquerosas pueden causarme alguna enfermedad o infección				1 2 3 4 5
99.- Siento repugnancia				1 2 3 4 5
100.- Las cosas asquerosas me revuelven el estómago				1 2 3 4 5
101.- Suelo poner cara de asco cuando algo me repugna				1 2 3 4 5
102.- Cuando tengo sensación de náuseas, me preocupa vomitar				1 2 3 4 5
103.- Cuando siento asco, suele ser una sensación intensa				1 2 3 4 5
104.- Suelo experimentar asco				1 2 3 4 5
105.- Me asusta la sensación de desmayarme				1 2 3 4 5
106.- Suelo sentir asco más fácilmente que otras personas				1 2 3 4 5
107.- Me preocupa poder tragar algo asqueroso				1 2 3 4 5
108.- Suelo toparme con cosas asquerosas				1 2 3 4 5
109.- Cuando siento asco me resulta embarazoso (me da corte o vergüenza)				1 2 3 4 5
110.- Creo que la sensación de asco es mala para mí				1 2 3 4 5

8.- Por último, se presentan una serie de afirmaciones relacionadas con su forma de ser. Señale en qué medida cree que le define cada una.

1	2	3	4	5
Nada característico para mí				Muy característico para mí
111.- Me atraen más los problemas muy complejos que los sencillos				1 2 3 4 5
112.- Me gusta asumir la responsabilidad de afrontar una situación que requiere pensar mucho				1 2 3 4 5
113.- Pensar no responde a mi idea de la diversión (creo hay muchas cosas más divertidas que pensar)				1 2 3 4 5
114.- Prefiero hacer algo que requiera pensar poco, a algo que sea un reto para mi capacidad intelectual				1 2 3 4 5
115.- Trato de evitar situaciones en las que se requiera pensar y reflexionar profundamente				1 2 3 4 5
116.- Me produce mucha satisfacción pasarme horas y horas reflexionando y pensando				1 2 3 4 5
117.- Prefiero pensar el mínimo necesario en cada caso				1 2 3 4 5
118.- Prefiero pensar sobre pequeños proyectos diarios que en otros a más largo plazo				1 2 3 4 5
119.- Me gustan las tareas que requieren pensar poco una vez que las he aprendido				1 2 3 4 5
120.- Me atrae la idea de tener que pensar mucho para conseguir algo				1 2 3 4 5
121.- Realmente me gustan las tareas que requieren encontrar nuevas soluciones a los problemas				1 2 3 4 5

ENCUESTA 3.0 (Halty, González y Sotoca, 2017)



122.- Aprender nuevas formas de pensar no me atrae demasiado	1	2	3	4	5
123.- Prefiero que mi vida esté llena de desafíos (o rompecabezas) que debo resolver	1	2	3	4	5
124.- La idea del pensamiento abstracto me atrae mucho	1	2	3	4	5
125.- Prefiero una tarea que sea intelectual, difícil e importante, más que una que no requiera pensar mucho, sea o no sea importante	1	2	3	4	5
126.- Completar una dura tarea que requiere mucho esfuerzo mental me produce alivio más que satisfacción	1	2	3	4	5
127.- Para mí es suficiente saber que las cosas funcionan; me importa poco cómo o por qué	1	2	3	4	5
128.- Me paso el tiempo reflexionando sobre cualquier cosa, incluso aunque no me afecte personalmente	1	2	3	4	5

Muchas gracias por su colaboración