



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE MÁSTER

*Análisis y evaluación de sistemas basados en IA para la
detección de Fake News en español / inglés. Una revisión
sistemática de literatura*

Máster Universitario en Dirección TIC para la Defensa

ALUMNO: José Manuel Martínez Sánchez

DIRECTORES: Milagros Fernández Gavilanes
Luis M. Álvarez Sabucedo

CURSO ACADÉMICO: 2023-2024

Universida_{de}Vigo



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE MÁSTER

*Análisis y evaluación de sistemas basados en IA para la
detección de Fake News en español / inglés. Una revisión
sistemática de literatura*

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

Universida_{de}Vigo

RESUMEN

Este trabajo aborda la problemática actual de las fake news y la necesidad de encontrar mecanismos automáticos que ayuden a su detección. Dado el alcance de internet y las redes sociales, su inmediatez y su facilidad de interacción, la mayoría de los ciudadanos ha elegido estos medios como su principal fuente de información. La falta de control sobre la veracidad de estas noticias está provocando altos niveles de desinformación en la población, lo que representa una seria amenaza para la sociedad. Su alto volumen y la velocidad a la que se producen y propagan hacen necesaria la utilización de medios automáticos que sirvan de ayuda a las agencias informativas y al público en general. En este contexto, numerosos estudios abordan el problema desde diferentes perspectivas, lo que puede provocar que un investigador que intente avanzar en este campo se sienta desorientado.

El objetivo perseguido es realizar una revisión sistemática de la literatura publicada hasta la fecha, orientada a noticias en español o inglés y que se basen en la utilización de la inteligencia artificial para conseguirlo. Para ello se ha utilizado la metodología PRISMA que proporciona un marco para identificar, seleccionar y evaluar estudios relevantes en este campo. La selección se realizó a partir de artículos de revistas científicas y actas de conferencias técnicas, publicados en cinco bibliotecas digitales: ACM Digital Library, IEEE Xplore, Science Direct, Scopus y Web of Science.

Como resultado se obtuvo la identificación de las características más frecuentes, las técnicas más utilizadas y los niveles de acierto de cada una. Al final se proporcionan unas conclusiones y posibles áreas de desarrollo que puedan servir como orientación a investigaciones futuras en este campo.

PALABRAS CLAVE

Fake news, detección, inteligencia artificial, machine learning, NLP

AGRADECIMIENTOS

A Mamen, mi querida esposa, por todo el tiempo que le he robado y la infinita paciencia que ha tenido durante la realización de este proyecto. Es tan suyo como mío.

A mis directores del trabajo, Milagros y Luis, sin cuya orientación y ayuda habría sido muy difícil alcanzar la meta propuesta.

Contenido

Índice de Figuras	3
Índice de Tablas.....	5
1 Introducción y objetivos	7
1.1 Introducción	7
1.2 Objetivos	8
1.3 Organización de la memoria	8
2 Contextualización	9
2.1 Fake News.....	9
2.1.1 La escalera de la manipulación.....	10
2.1.2 Los sesgos cognitivos y la difusión en internet	10
2.1.3 Técnicas de manipulación.....	11
2.2 Inteligencia Artificial	12
2.3 Machine Learning	13
2.3.1 Funcionamiento de un sistema de ML	13
2.3.2 Clasificación de algoritmos de ML.....	14
2.3.3 Algoritmos de ML más utilizados	16
2.3.4 Deep Learning	22
2.3.5 Arquitecturas de DL	23
2.4 Procesamiento de Lenguaje Natural	26
2.4.1 Tareas habituales en el NLP	26
2.4.2 Representación del texto en vectores numéricos	27
2.4.3 Técnicas de word embedding	28
2.4.4 Modelos enormes de lenguaje	33
2.5 Detección de fake news.....	35
2.5.1 Combatir las fake news individualmente.....	35
2.5.2 Verificación de hechos o Fact-checking.....	35
2.5.3 Enfoques de detección automática.....	37
2.6 Revisión sistemática de literatura (SLR)	39
2.6.1 Metodología PRISMA.....	39
2.6.2 Herramienta Parsifal	41
3 Método de Revisión.....	43
3.1 Criterios de elegibilidad	43
3.1.1 Criterios de Inclusión:.....	43

3.1.2 Criterios de Exclusión:	43
3.2 Fuentes de información	43
3.3 Estrategia de búsqueda	44
3.4 Proceso de selección	44
3.5 Proceso de extracción de datos	46
3.6 Lista de datos a extraer	49
3.7 Evaluación de calidad	49
4 Resultados	51
4.1 Selección de estudios	51
4.2 Descripción de los estudios	52
4.3 Síntesis de resultados	55
4.3.1 Características estudiadas	55
4.3.2 Métodos de extracción de características	56
4.3.3 Modelos de ML y DL utilizados	57
4.3.4 Técnicas de IA utilizadas	59
4.3.5 Modelos de Aprendizaje y grado de precisión	61
5 Conclusiones y líneas futuras	65
5.1 Discusión	65
5.1.1 PI 1. ¿Cuáles son las áreas de la IA con aplicación en la detección de fake news?	65
5.1.2 PI 2. ¿Qué técnicas se están usando en la detección de fake news y qué resultados?	66
5.1.3 PI 3. ¿Qué retos y limitaciones se están encontrando en la detección de fake news?	66
5.2 Conclusiones	67
5.3 Líneas futuras	67
6 Bibliografía	69

Índice de Figuras

Figura 2-1 Fake News, la fábrica de mentiras. Fuente:[1]	9
Figura 2-2 Campaña presidencial de EEUU en 2016. Fuente:[2]	10
Figura 2-3 Intoxicación, más de lo que podemos procesar. Fuente:[3]	11
Figura 2-4 Imagen falsa generada mediante IA. Fuente:[1]	12
Figura 2-5 Ramas de la Inteligencia Artificial. Fuente:[6]	13
Figura 2-6 Desajuste (<i>underfitting</i>) y sobreajuste (<i>overfitting</i>). Fuente:[8]	14
Figura 2-7 Tipos de algoritmos según supervisión y sus aplicaciones. Fuente:[9]	15
Figura 2-8 Regresión lineal y Regresión logística. Fuente:[8]	17
Figura 2-9 Árboles de decisión y SVM. Fuente:[8]	18
Figura 2-10 Comparación de agrupamiento de k-means con el de DBSCAN. Fuente:[8]	20
Figura 2-11 PCA, selección de hiperplano sobre el que proyectar. Fuente:[7]	20
Figura 2-12 DL, subconjunto del ML. Fuente:[11]	22
Figura 2-13 Unidad Lógica de Umbral TLU. Fuente:[7]	22
Figura 2-14 Perceptrón con dos entradas. Fuente:[7]	23
Figura 2-15 Multi-Layer Perceptron (MLP). Fuente:[7]	23
Figura 2-16 Arquitecturas de DL. Fuente:[12]	24
Figura 2-17 Red neuronal LSTM con detalle de una célula. Fuente:[12]	25
Figura 2-18 Representación vectores <i>one-hot encoding</i> y <i>embedding</i> . Fuente:[15]	27
Figura 2-19 Clasificación de técnicas de <i>word embedding</i> . Fuente:[16]	28
Figura 2-20 Vectores con relaciones sintácticas y semánticas. Fuente:[17]	30
Figura 2-21 Arquitecturas utilizadas en el modelo Word2vec. Fuente:[18]	31
Figura 2-22 Red Neuronal Recurrente desplegada en el tiempo. Fuente:[19]	31
Figura 2-23 Arquitectura <i>transformer</i> . Fuente:[20]	32
Figura 2-24 Modelos enormes de lenguaje (LLM) según parámetros. Fuente:[21]	34
Figura 2-25 Sitio web del International Fact-Checking Network. Fuente:[23]	36
Figura 2-26 EFE Verifica, sitio web de verificación de la Agencia EFE. Fuente:[25]	37
Figura 2-27 Perspectivas de la detección de fake news en su ciclo de vida. Fuente:[26]:	38
Figura 2-28 Pantalla descriptiva de la revisión en Parsifal. Fuente: Autor	41
Figura 2-29 Gráfico de Parsifal con número de artículos encontrados por fuente. Fuente: Autor ..	42
Figura 3-1 Pantalla de resultados de ACM. Fuente: Autor	44
Figura 3-2 Herramienta Parsifal, selección de artículos. Fuente: Autor	45
Figura 3-3 Pantalla de herramienta ChatPDF. Fuente: Autor	46
Figura 3-4 Hoja de cálculo para extracción de datos. Fuente: Autor	49

Figura 3-5 Preguntas de evaluación de calidad en Parsifal. Fuente: Autor.....	50
Figura 4-1 Diagrama de selección de estudios según PRISMA (elaboración propia).	52
Figura 4-2 Tipos de características estudiadas (elaboración propia).....	55
Figura 4-3 Enfoques utilizados en la detección (elaboración propia).	56
Figura 4-4 Técnicas de NLP utilizadas (elaboración propia).	57
Figura 4-5 Tipos de Aprendizaje automático utilizados (elaboración propia).	57
Figura 4-6 Técnicas de ML utilizadas (elaboración propia).	58
Figura 4-7 Técnicas de DL utilizadas (elaboración propia).	58

Índice de Tablas

Tabla 2-1 Algoritmos de ML Supervisados. Fuente:[7].....	19
Tabla 2-2 Algoritmos de ML No Supervisados. Fuente:[7].....	21
Tabla 2-3 Secciones y Temas de la metodología PRISMA. Fuente:[28].....	40
Tabla 3-1 Batería de preguntas realizadas a ChatPdf. Fuente: Autor	47
Tabla 3-2 Preguntas y respuestas de ejemplo en ChatPDF. Fuente: Autor	48
Tabla 4-1 Estudios seleccionados con autor, características y resumen. Fuente: Autor	55
Tabla 4-2 Estudios seleccionados con métodos utilizados. Fuente: Autor	60
Tabla 4-3 Estudios con técnicas, <i>datasets</i> y resultados obtenidos. Fuente: Autor	63

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Introducción

Desde la antigüedad, existen fake news, noticias falsas creadas con el objetivo de sembrar desinformación en la sociedad y obtener un beneficio con ello, pero nunca han supuesto una amenaza tan grave como en la actualidad. La razón se encuentra en su facilidad de creación y su rapidez de difusión mediante internet y las redes sociales. Hoy en día, cualquier persona con acceso a un ordenador o un teléfono móvil puede crear una noticia falsa, cuyas consecuencias y alcance pueden ser inimaginables. Esto hace que, las agencias de información y los medios de comunicación, que hasta ahora ejercían la tarea de control y verificación de la información que llegaba a los ciudadanos, se vean incapaces de acreditar o desmentir el ingente volumen de noticias que se genera diariamente. Para agravar más el problema, cada día más personas utilizan las redes sociales como principal fuente de información para saber qué sucede en el mundo, por su rapidez, economía y posibilidad de expresar su opinión, en comparación con las fuentes periodísticas clásicas.

Para abordar este problema existen varias opciones. La más sencilla y podría ser la más efectiva, es la concienciación de los usuarios de las redes sociales, para que no transmitan aquello de lo que no están seguros que sea cierto. Sin embargo, esta opción es difícil de llevar a cabo porque las fake news están específicamente diseñadas para que los usuarios no puedan resistirse a reenviarlas, pues provocan sentimientos intensos de indignación, sorpresa, afinidad o repulsa. Además, en el caso de que el lector decidiera comprobar si esa noticia es cierta, tampoco es tan sencillo realizar esa comprobación, pues muchas veces están redactadas de forma ambigua y poco clara. Otra opción son las agencias de verificación, que realizan una labor de divulgación de todas las noticias falsas que llegan a su alcance, pero aquí los problemas son que no tienen capacidad para comprobar todas las fake news al ritmo que se generan y que precisan que el usuario vaya a su página web a verificar las noticias.

Por todo lo anterior, se hace necesaria la creación de soluciones tecnológicas que permitan detectar las fake news de forma automática, eficiente y fiable. Con los últimos avances en inteligencia artificial, concretamente en el NLP y el aprendizaje automático, el aumento de la capacidad de cálculo y la capacidad de manejar grandes volúmenes de datos, estas soluciones empiezan a dar buenos resultados en los estudios prácticos que se están realizando.

No obstante, el alto interés de la comunidad científica sobre este asunto provoca que haya gran cantidad literatura técnica y que un investigador que quiera desarrollar un estudio en este sector se sienta abrumado por ella. Este trabajo pretende reducir este problema, proporcionando un estado del arte sobre este tema a la vez que aporta información sobre los métodos y técnicas utilizados.

1.2 Objetivos

Este trabajo tiene como objetivo facilitar una visión de cómo se está aplicando la inteligencia artificial, concretamente el procesamiento del lenguaje natural y el aprendizaje automático, en la detección de fake news en español o inglés. Para ello se plantean las siguientes preguntas de investigación a resolver:

- PI 1. ¿Cuáles son las áreas de la IA con aplicación en la detección de fake news?
- PI 2. ¿Qué técnicas se están usando en la detección de fake news y con qué resultados?
- PI 3. ¿Qué retos y limitaciones se están encontrando en la detección de fake news?

1.3 Organización de la memoria

Este trabajo está organizado en cinco secciones:

1. Introducción y objetivos. Donde se plantea la problemática existente y una motivación del trabajo que se va a realizar. Además, se hace una declaración expresa de los objetivos que se pretenden abordar en la investigación.
2. Contextualización. En esta sección se hace una descripción de los elementos que componen el entorno que se va a investigar. Se pretende que todos los términos necesarios para comprender el trabajo se hayan definido previamente en esta sección.
3. Método de revisión. En este apartado se define la planificación de los pasos que se van a realizar conforme a la metodología aplicada.
4. Resultados. Se detallan los resultados obtenidos, en la investigación, bajo distintas perspectivas y métricas.
5. Conclusiones y líneas futuras. Se exponen las conclusiones obtenidas tras la discusión de los resultados y se proponen líneas futuras de acción.

2 CONTEXTUALIZACIÓN

2.1 Fake News

Las fake news son noticias falsas que se difunden a través de los medios de comunicación, internet o las redes sociales, con la intención de crear un círculo vicioso de desinformación en la sociedad para obtener un beneficio. Sus fines pueden ser muy variados, desde políticos o sociales, hasta simplemente económicos, como el de conseguir más lectores para tener más ingresos por publicidad. Dada esa intencionalidad quizás sería más correcta la traducción del anglicismo por el de noticias falseadas, aunque en este trabajo nos seguiremos refiriendo a ellas como fake news, debido al carácter técnico del mismo y la amplia difusión del término en inglés [1].



Figura 2-1 Fake News, la fábrica de mentiras. Fuente:[1]

Desde la antigüedad han existido noticias falsas creadas con la intención de influir sobre la opinión pública. No obstante, su capacidad de difusión estaba limitada por los medios disponibles en la época. Estos medios al principio eran lentos y sólo estaban a disposición de unos pocos, es el caso de la propaganda política llevada a cabo por los gobernantes como los del Imperio Romano, los reinos medievales o los estados modernos. Con la aparición de los avances tecnológicos, como la imprenta, se fue aumentando su rapidez y alcance, hasta llegar a los medios de comunicación de masas, como los periódicos, la radio o la televisión, donde alcanzaron la inmediatez. Pero estos medios seguían estando controlados por una élite propietaria de ellos que hacía que se mantuviera el sentido descendente de la información.

Sin embargo, con la aparición de internet y las redes sociales, la pirámide se ha invertido. Los usuarios son a la vez productores y consumidores de información, de tal forma que cualquier persona puede crear una noticia falsa, difundirla y acabar haciéndose “viral”, llegando a millones de personas

en segundos, si se dan las condiciones adecuadas. Se crea un círculo vicioso de desinformación y distribución de información falsa que es muy difícil de evitar.

El término de fake news se hizo popular durante la campaña de las elecciones presidenciales de 2016 de los Estados Unidos, en la que el candidato republicano, Donald Trump (Figura 2-2), lo utilizó con frecuencia para referirse a noticias que no le eran favorables, fueran ciertas o no. Casi inmediatamente empezó a ser utilizado por los medios de comunicación internacionales y saltó fronteras incorporándose al vocabulario cotidiano de todo el mundo[1].



Figura 2-2 Campaña presidencial de EEUU en 2016. Fuente:[2]

2.1.1 La escalera de la manipulación

Bajo la idea de información falsa, varios conceptos se entremezclan, siendo el más amplio de ellos el de la desinformación y las fake news serían un subconjunto de ella. Podríamos decir que dentro de la desinformación tenemos siete niveles de degradación de la verdad, según el grado de manipulación que haya sufrido la información. Es lo que se conoce como la escalera de la manipulación [1].

- **Sátira o parodia.** Tiene por objetivo hacer reír o criticar, no engañar, pero puede llevar a error por su formato que imita al de una noticia real.
- **Conexión falsa.** El titular o la imagen de portada, normalmente impactante o tendencioso, no coincide con el contenido de la noticia.
- **Omisión de contenido.** La información no es falsa, pero se muestra de tal forma que induce a conclusiones falsas por faltar partes relevantes de la misma.
- **Contexto falso.** El contenido es cierto, pero se muestra fuera de su contexto y cambia el significado. Puede ser temporal, de ubicación, de actores, etc.
- **Contenido impostor.** Se falsifica la fuente de una información, se utilizan logos falsos o aspecto de los documentos. De esta forma parece que la información procede de esa fuente y le da más credibilidad, o daña la imagen de la fuente suplantada.
- **Contenido manipulado.** Se modifica parte de un contenido existente, como imágenes o audio para que el resultado final sea distinto, y lógicamente falso.
- **Contenido fabricado.** Toda la noticia es inventada, no se ha producido nunca y no tiene ninguna conexión con la realidad.

2.1.2 Los sesgos cognitivos y la difusión en internet

Las personas no percibimos la información que recibimos de la misma forma, todos la vemos distorsionada, influidos por los patrones que tenemos adquiridos de forma inconsciente. Es lo que denominamos los **sesgos cognitivos** [1].

Normalmente, tendemos a aceptar y por lo tanto a difundir, la información más acorde con nuestras convicciones y creencias. Aunque no sea cierta, la preferiremos a otra que, aun siendo verdadera, nos desagrade. Esto es el **sesgo de confirmación**. Además, los algoritmos de las redes sociales y de internet están diseñados para mantenernos el mayor tiempo posible consumiendo

información, para ello nos mostrarán cada vez más información que nos guste, creando una burbuja que nos impide conocer otros puntos de vista.

Hay otros sesgos también muy influyentes en nuestra opinión. Por ejemplo, el **sesgo de anclaje** que se basa en que tendemos a dar por buena la información que primero llega a nosotros, rechazando toda información contraria posterior. Otro es el **sesgo de autoridad**, que nos hace dar por cierto lo que parece proceder de una fuente respetable, aunque la referencia haya sido manipulada. Por último tenemos el **sesgo de repetición**. Cuando una noticia es repetida muchas veces acabamos dándola por cierta sin cuestionarla porque otras noticias falsas previas la validan.

Unido a todos estos sesgos tenemos el concepto de **posverdad** [1], que se produce cuando los hechos en una noticia importan menos que las creencias o las emociones de la opinión pública que la recibe. Ante un contenido nuevo en el que se tuviera que primar la racionalidad o la emoción, normalmente ganarán las emociones.

La motivación para crear fake news muchas veces es económica, precisamente por la forma en que está diseñado internet. Hay una tendencia en la red que consiste en crear unas noticias con unos titulares muy llamativos, pero con un contenido muy pobre, que no corresponde con el reclamo. Sin embargo, si la abrimos consigue su objetivo, que es que hagamos clic y carguemos la página, es un **ciberanzuelo**. De esta forma consiguen ganar dinero simplemente por las visitas recibidas. Otra caso similar es el de las noticias virales. Estas son rentables, pero no por lo interesante que sea su contenido, si no por lo rápida y extensa que sea su difusión. De hecho, las noticias falsas están diseñadas para difundirse mejor y más rápido que las verdaderas.



Figura 2-3 Infoxicación, más de lo que podemos procesar. Fuente:[3]

Otra circunstancia que influye en la difusión de las fake news es la **infoxicación** (Figura 2-3). Recibimos tanta información en nuestro día a día que prácticamente, sólo podemos leer los titulares y algo del contenido, sin poder profundizar debidamente y crearnos una opinión propia al respecto. Esto se agrava con la necesidad de estar conectados con todo y saber todo lo que está pasando en todas partes. Nos convertimos en simples transmisores sin saber muy bien lo que estamos remitiendo[3].

2.1.3 Técnicas de manipulación

Hay muchas formas de manipular el contenido de las noticias. Unas llevan mucho tiempo existiendo, prácticamente han ido surgiendo inmediatamente después de aparecer el medio que las soportan. Pero otras, de reciente aparición, están ligadas a las nuevas tecnologías como la inteligencia artificial.

La manipulación de las imágenes es algo que se viene haciendo desde el mismo momento en que apareció la fotografía. Recortar zonas que no interesan, fusionar dos fotografías, eliminar personas o fondos, es algo que se hacía ya cuando la fotografía aún era en blanco y negro[4].

Lo mismo podemos decir del audio o del vídeo. Cortar trozos que no interesaban o añadir ruido o partes que en realidad no existieron, grabación de maquetas, recrear escenas que no sucedieron y grabarlas como reales, se hacía ya el siglo pasado.



Figura 2-4 Imagen falsa generada mediante IA. Fuente:[1]

Sin embargo, con la aparición de la inteligencia artificial se ha dado un salto cualitativo que nos hace pensar que en poco tiempo no será fácil detectar si estamos ante algo real o algo creado artificialmente. Las herramientas generativas pueden crear imágenes basándose en los datos que le demos con una calidad similar a una fotografía tomada directamente de la realidad, como podemos apreciar en esta imagen falsa del Papa Francisco creada mediante inteligencia artificial (Figura 2-4). El **deepfake** permite combinar la voz y los rasgos de una persona con otra, de tal forma que podemos ver a esa persona diciendo cosas que jamás dijo, o directamente se pueden crear rostros de personas que no existen con una calidad tal que sean muy difíciles de detectar[1].

2.2 Inteligencia Artificial

Según Asunción Gómez-Pérez, podemos considerar la inteligencia artificial (IA) como la tecnología habilitadora cuyo objetivo es dotar a las máquinas de capacidades intelectuales propias de los seres humanos, como razonar, entender el lenguaje, traducir entre idiomas, interpretar documentos o aprender [5].

Si bien es un concepto que se maneja desde hace bastantes años, el término apareció formalmente durante la Conferencia de Darmouth en 1956, donde John McCarthy acuñó la expresión “inteligencia artificial”, y la definió como “la ciencia e ingenio de hacer máquinas inteligentes, especialmente programas de cómputo inteligentes”.

Tras estar un tiempo relegada al ámbito académico, resurgió con fuerza en los últimos años, coincidiendo con avances tecnológicos, como el aumento de la capacidad de computación, el *big data* o el almacenamiento en la nube, que permitieron llevar a la práctica algoritmos y soluciones que se habían ido desarrollando solamente en el plano teórico.

Su evolución estos últimos años ha sido exponencial, irrumpiendo con fuerza en nuestras vidas, de hecho, fue nombrada palabra del año 2022 por la Fundación del Español Urgente (FundeuRAE).

Precisamente por estos cambios, la RAE ha ido adaptando su definición según ha ido evolucionando el concepto, considerando en la actualidad IA como la “disciplina científica que se ocupa de crear programas informáticos que ejecutan operaciones comparables a las que realiza la mente humana, como el aprendizaje o el razonamiento lógico”.

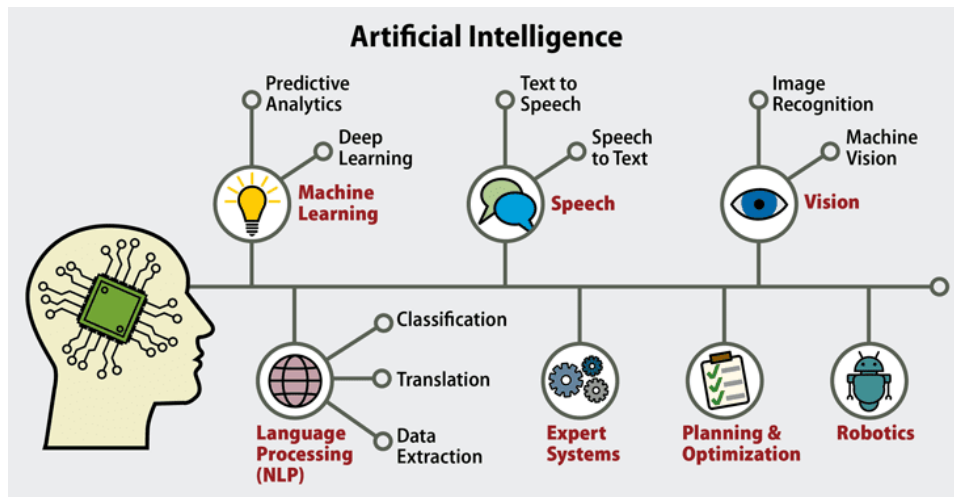


Figura 2-5 Ramas de la Inteligencia Artificial. Fuente:[6]

Dados los múltiples aspectos que abarca, la IA se divide en ramas según la capacidad humana con la que está relacionada como podemos apreciar en la Figura 2-5. Así tenemos: visión, voz, sistemas expertos, robótica, planificación, Machine Learning (ML) o Procesamiento de Lenguaje Natural (*Natural Language Processing* o NLP). Este trabajo se basará principalmente en las dos últimas ML y NLP.

2.3 Machine Learning

El ML o aprendizaje automático, según Aurélien Géron, es “la ciencia (y arte) de programar computadores para que puedan aprender de los datos” [7]. Más concretamente, se refiere a una programación no específica para una tarea concreta, que permita que las máquinas puedan modificar el algoritmo y ajustarlo conforme vayan variando los datos de entrada. De un modo más generalista, pero en el mismo sentido, Arthur Samuel, en 1959, dijo que es “el campo de estudio que proporciona a los computadores la habilidad de aprender sin ser explícitamente programados”. Aquí el matiz lo tenemos en el término “no explícitamente”, que implica que la programación no tiene un objetivo puntual, sino que irá variando conforme lo hagan los datos y los resultados deseados.

Por lo tanto, podríamos decir que ML sí necesita una programación de las máquinas, pero no para realizar una tarea puntual, sino de tal forma que, conforme vayan evolucionando los datos que tratan, se vaya modificando el comportamiento de los programas para proporcionar los resultados deseados, sin necesidad de intervención humana.

Esto tiene gran aplicación en sistemas donde hay que realizar muchos ajustes o aplicar reglas complicadas para llegar a un correcto funcionamiento, por ser un entorno complejo, con gran cantidad de datos o muy cambiante. El caso que nos ocupa cumple todas las condiciones anteriores, pues se trataría de clasificar las noticias que se difunden por medios electrónicos, en falsas o no, sin tener que estar modificando los criterios que las definen constantemente conforme van evolucionando.

2.3.1 Funcionamiento de un sistema de ML

De forma muy resumida, el funcionamiento de un sistema de ML consiste en entrenar un sistema para que aprenda a realizar unas tareas partiendo de unos datos de entrenamiento, y utilizando para

ello algoritmos sobre los que se van variando sus parámetros de configuración hasta obtener los mejores resultados posibles. A continuación, es necesario validar estos resultados con otro conjunto de datos de prueba y por último, aunque no siempre se lleva a cabo, se debería generalizar el modelo para casos en los que nunca ha sido usado.

Un sistema de ML, para funcionar correctamente, tiene que disponer de datos suficientes y de calidad, y un algoritmo adecuado para el problema al que se enfrenta. Respecto a los datos, no sólo importa la cantidad, sino que también tienen que ser representativos, es decir, no tienen que tener sesgo de ningún tipo, ni “ruido” o elementos claramente incorrectos o anormales que distorsionan el sistema. Además, las características de los datos usadas en el modelo deben ser relevantes para el estudio y tener formato y dimensión correctos. Es muy importante hacer una preparación previa de esos datos antes de empezar el entrenamiento para minimizar estos problemas.

En cuanto al algoritmo, no puede ser demasiado simple, pues no podría representar el modelo (*underfitting*) y por lo tanto daría unos valores de acierto muy bajos con los datos de prueba, ni demasiado complejo, pues eso querría decir que está muy bien adaptado a los datos de prueba (*overfitting*), pero generalizaría mal cuando tuviera que tratar con datos nuevos.

Detectar el problema de *overfitting* no es tarea sencilla pues no se suele detectar hasta llegar a la realización de pruebas con datos reales. Para evitarlo se puede utilizar una técnica denominada validación cruzada, que consiste en partir los datos de entrenamiento en subconjuntos o *folds*, y entrenar el sistema con cada subconjunto por separado, validándolo contra los datos de los otros subconjuntos como si fueran los datos de prueba. Por último, se podrán usar los verdaderos datos de prueba para comprobar el modelo final resultante de combinar los resultados de todos los subconjuntos.

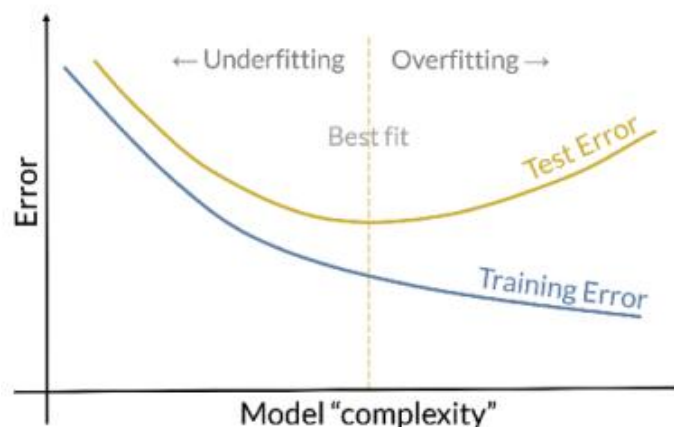


Figura 2-6 Desajuste (*underfitting*) y sobreajuste (*overfitting*). Fuente:[8]

Para el *underfitting* la solución pasa por cambiar a un modelo más potente y seleccionar mejores características, mientras que para el *overfitting* se necesitaría aumentar los datos y reducir el ruido de estos, simplificar el modelo, y utilizar hiperparámetros del algoritmo que mejoren su comportamiento ante el *overfitting*. En la Figura 2-6 podemos observar cómo va variando la tasa de error del resultado sobre los datos de entrenamiento y de prueba, conforme aumenta la complejidad del modelo, y dónde tenemos el punto de equilibrio entre *underfitting* y *overfitting*.

2.3.2 Clasificación de algoritmos de ML

Hay numerosos algoritmos de ML, por lo que es interesante agruparlos para identificar aquellos que reúnan las características que mejor se ajusten al problema que se quiere resolver. Los podemos clasificar atendiendo a su supervisión, según su capacidad de aprendizaje en tiempo real y según compare los datos nuevos con los datos conocidos o generalización[7]. En la Figura 2-7 se puede observar una clasificación de los algoritmos de ML atendiendo a su supervisión.

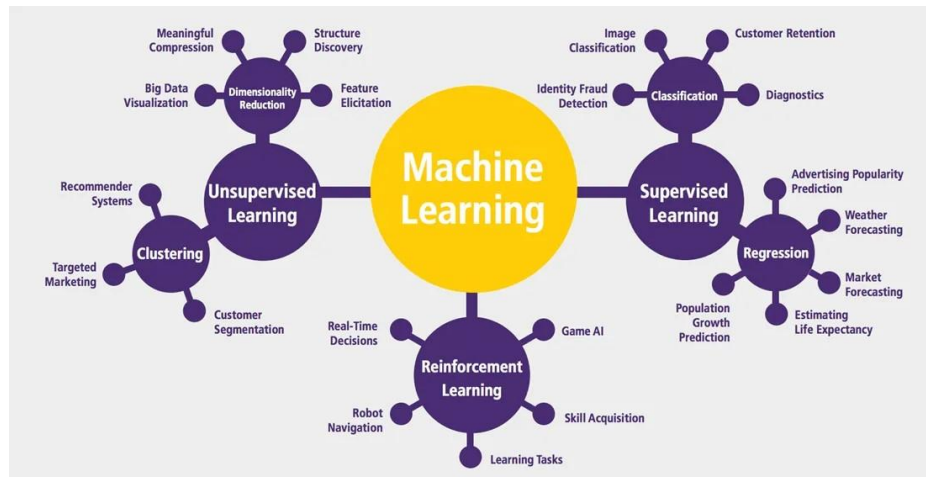


Figura 2-7 Tipos de algoritmos según supervisión y sus aplicaciones. Fuente:[9]

Así, atendiendo a la cantidad y tipo de supervisión que precisan durante su entrenamiento, los podemos dividir en supervisados, sin supervisión, semisupervisados y aprendizaje con refuerzo [7].

- Los **supervisados** precisan que los datos de entrenamiento incluyan la solución deseada, denominadas “etiquetas”, y se basan en buscar la mejor aproximación a estas en los resultados proporcionados por el modelo. Tareas típicas de estos algoritmos son la clasificación y la regresión. La clasificación consiste en identificar a qué clase pertenece una instancia en función de sus características (fake new o no), mientras que la regresión pretende predecir un valor numérico como resultado (precio de un inmueble). Aunque cada algoritmo está orientado a una de estas tareas, la realidad es que se pueden utilizar para ambas.

Si se combinan varios algoritmos en un sistema para la resolución de un problema, el resultado siempre suele ser mejor que el de los modelos individualmente. A esta práctica se la denomina ensamble [7]. Existen dos técnicas principales, *bagging* y *boosting*. En el **bagging** los modelos se entrenan en paralelo y se combinan las predicciones de todos al final. Es el caso del modelo *Random Forest* que combina varios árboles de decisión y veremos en el siguiente apartado[7]. El **boosting**, por otro lado, consiste en encadenar varios métodos secuencialmente, cada uno mejorando las debilidades del anterior.

- Los **no supervisados** no disponen de etiquetas en los datos que digan cómo de correctos son, por lo que lo único que pueden hacer es intentar buscar similitudes entre los elementos o instancias atendiendo a sus características. La validación del modelo se vuelve más complicada al no tener referencias para comparar. Tareas típicas que abordan son:
 - Agrupación o *Clustering*: Consiste en descubrir agrupaciones en los datos, en función de características con valores similares. Elementos de una misma clase tendrán características similares y al contrario (tipos de clientes).
 - Detección de anomalías y novedades: Identificar automáticamente lo que son datos normales y buscar elementos que resulten anómalos o nuevos respecto a los grupos que se van definiendo (pagos fraudulentos).
 - Visualización y reducción de dimensiones: El objetivo es buscar correlaciones entre variables para agruparlas y conseguir modelos más sencillos, más fáciles de entrenar y sin perder mucha información (antigüedad de vehículo y kilómetros recorridos).

- Aprendizaje de reglas de asociación: Profundizar en conjuntos de datos muy grandes, intentando descubrir relaciones de interés entre atributos (preferencias de compras).
- Los **semisupervisados** son aquellos que tienen una pequeña parte de elementos etiquetados y el resto sin etiquetar. Las tareas que suelen realizar son las de ambos y los algoritmos también son una combinación de los dos anteriores.
- Por último el **aprendizaje con refuerzo** [7] es bastante diferente de estos. Un agente realiza una acción y recibe una recompensa o no. El agente va definiendo la mejor estrategia para conseguir el mayor número de recompensas y modificando su comportamiento. Estos algoritmos son típicos de la robótica.

Otra forma de clasificación se basa en si el sistema puede ir aprendiendo con datos nuevos que se van incorporando al sistema de forma continua, a la vez que sigue realizando su tarea, o necesita realizar un entrenamiento con datos y luego incorporar el conocimiento adquirido al algoritmo que se está aplicando con los datos reales.

- En los modelos **Batch Learning**, el sistema necesita entrenar con muchos datos y mucho tiempo, por lo que no puede hacerlo a la vez que van apareciendo los datos. Hay que ir entrenando versiones del modelo, evaluarlas y después lanzarlas a producción cuando estén listas. Este proceso se puede automatizar pero hay que parar el modelo y cargar el nuevo cada vez que se actualice.
- En los sistemas **Online Learning**, los datos se van incorporando al modelo en pequeños lotes de forma incremental y se va entrenando con ellos según llegan. Es útil para sistemas que reciben datos constantemente y tienen que reaccionar con mucha rapidez.

2.3.3 Algoritmos de ML más utilizados

Como se indicó anteriormente, hay gran cantidad de algoritmos de ML con numerosas variaciones sobre ellos, por lo que solamente se van a describir brevemente los más representativos y utilizados.

Algoritmos Supervisados:

- **Regresión lineal.** Es un algoritmo supervisado que proporciona el valor de una o varias variables dependientes, predicción, en función de una o varias variables independientes, características, a las que se les asigna un peso o parámetro que se va ajustando durante el entrenamiento. Para ello, comparamos la predicción realizada mediante la fórmula con el resultado conocido de los datos u objetivos.
Para medir esta comparación se pueden usar distintas funciones de coste. Las más frecuentes son el error cuadrático medio (RMSE, raíz de la suma de los cuadrados) y el error absoluto medio (MAE, media de los errores absolutos). Para ir mejorando el modelo se puede aplicar una fórmula que encuentre los parámetros que minimice el valor de la RMSE o de la MAE, o se puede utilizar una técnica denominada Gradiente Descendente que busca esos valores de los parámetros mediante sucesivas iteraciones hasta encontrar un mínimo para la función de coste.
- **Regresión Logística.** Se suele utilizar para estimar la probabilidad de que una instancia pertenezca a una o varias clases. Es similar a la regresión lineal, pero modificada para que los resultados que proporcione sean valores entre 0 y 1. De esta forma, podemos dividir los resultados en dos clases mediante un plano. Sin embargo, cuando el valor está cerca del 50% el resultado es prácticamente aleatorio. Para entrenar el modelo hay que utilizar la técnica de Gradiente Descendente.

Al igual que en la regresión, en la clasificación también se utilizan unas métricas que nos el nivel de cierto, pero en este caso se hace en función de los valores correctos y los valores totales. De esta forma, tenemos:

- Exactitud o *Accuracy*. Proporción elementos clasificados correctamente, tanto positivos como negativos, con respecto al total de elementos. En datos balanceados es la que más se suele utilizar.
- Recuperación o *Recall*. Proporción de elementos positivos correctamente identificados con respecto al total de elementos positivos.
- Precisión. Porcentaje de elementos positivos correctamente clasificados respecto al número de elementos clasificados como tales.
- F1 Score. Combina las dos anteriores. $F1 = 2 * (\text{precisión} * \text{recuperación}) / (\text{precisión} + \text{recuperación})$ [10].

En la Figura 2-8 podemos observar una representación gráfica de una regresión lineal y una regresión logística.

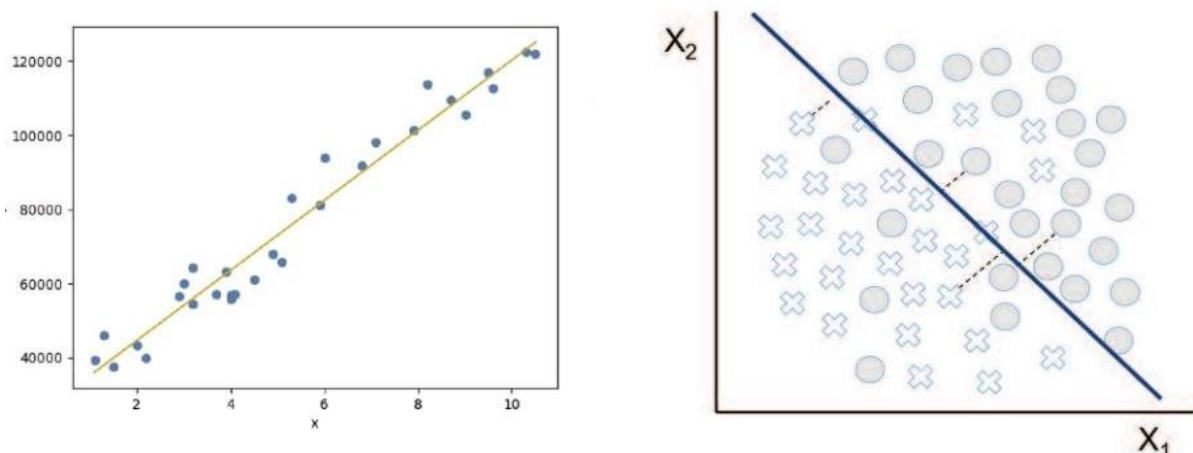


Figura 2-8 Regresión lineal y Regresión logística. Fuente:[8]

- **K-Nearest Neighbors (K-NN).** Es un modelo muy simple, utilizado para realizar predicciones y clasificaciones. Cuando se recibe una nueva instancia, se seleccionan las n instancias más cercanas del conjunto de entrenamiento, en función de las características del modelo, y se le asigna la clase más frecuente en una clasificación, o la media en caso de una regresión.
- **Árboles de decisión (DT).** Es un algoritmo capaz de resolver modelos complejos, muy fácil de visualizar por humanos. Consiste en ir dividiendo los datos en subconjuntos desde un nodo raíz, según cumplan o no una condición sobre los valores de las características, hasta llegar a unos nodos que no tienen división, llamados nodos hoja. Los datos no precisan preparación ni escalado, pues no se aplican fórmulas de coste como en las regresiones. Para su entrenamiento se tienen en cuenta dos valores en cada nodo: el número de elementos en el nodo y el número de errores en la asignación. Si se dividen los errores entre los elementos del nodo, da el valor de pureza, si es cero se dice que es puro. Se van modificando las preguntas de los nodos intermedios hasta ir consiguiendo nodos lo más puros posibles y con el mayor número de elementos, o que no pueden mejorar. Al final, se intenta conseguir los mejores valores globales de pureza para el modelo completo. Como este sistema tiende al sobreajuste, hay una serie de hiperparámetros que intentan mitigar ese comportamiento, como por ejemplo: número máximo de niveles, número mínimo de

elementos en un nodo, etc. Se precisa mucho cálculo para obtener un árbol óptimo, pero, una vez construido, las predicciones son muy rápidas.

- **Support Vector Machine (SVM).** Es uno de los modelos más utilizados en ML y consiste en dividir las instancias en dos clases con una línea recta, tratando de que la separación entre ambas sea lo mayor posible. Las instancias más cercanas al borde de cada una de las clases se llaman vectores de soporte y son los que delimitan la zona de separación. Dado que los datos tienen que ser linealmente divisibles se pueden añadir características que fuercen a los datos a separarse, por ejemplo polinómicas, campanas de Gauss, etc. Tiene muy buen comportamiento para clasificaciones de datos complejos, pero no muy grandes, debido a la complejidad computacional que precisa. En las siguientes gráficas se puede observar un árbol de decisión y un SVM (Figura 2-9).

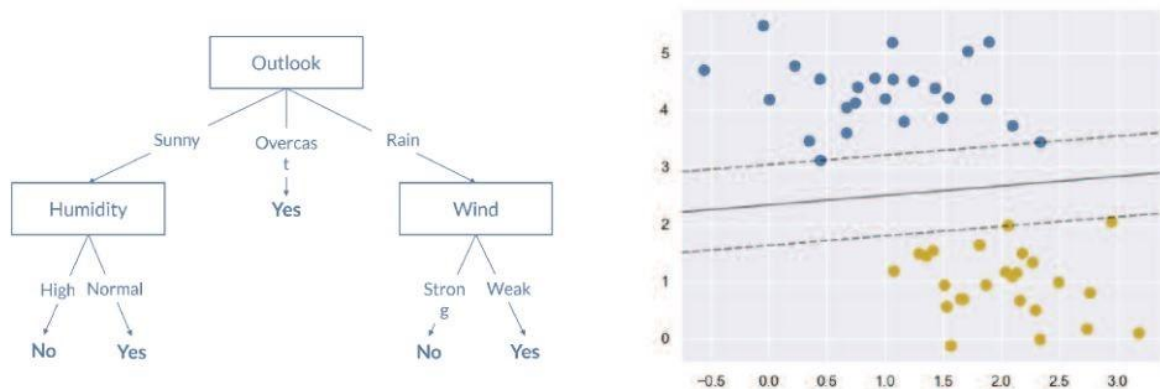


Figura 2-9 Árboles de decisión y SVM. Fuente:[8]

- **Naive Bayes (NB).** Es un algoritmo que se basa en el Teorema de Bayes para calcular la probabilidad de que un elemento pertenezca a una clase. Lo hace en función de la probabilidad de cada una de las características del modelo por igual. Toma las probabilidades de cada atributo de forma independiente, sin que unas influyan sobre las otras, lo que no suele cumplirse en el mundo real. A pesar de su simplicidad, funciona bastante bien en muchas soluciones complejas y sin tener que utilizar conjuntos de datos muy grandes.
- **Random Forest (RF).** Este algoritmo, en realidad, es un ensamble de varios árboles de decisión en modo *bagging*. Es decir, calcula todos los árboles en paralelo sobre conjuntos de datos distintos y, como resultado, proporciona la combinación de estos. Es uno de los algoritmos más utilizados y potentes porque proporciona muy buen rendimiento y reduce los problemas de sobreajuste y sesgo que suelen tener los árboles de decisión.
- **Gradient Boosting (GB).** Este algoritmo, al igual que el anterior, también es un ensamble de árboles de decisión, sólo que en modo *boosting*. Esto es, que combina secuencialmente una serie de árboles de forma que cada uno intenta corregir los errores residuales de predicción del anterior. El algoritmo **XGBoost** es una variante de este algoritmo de muy alto rendimiento, extremadamente rápido, escalable y portable.
- **Adaptive Boosting (AdaBoost).** Este algoritmo es similar al anterior, pero en lugar de centrarse en los errores del árbol anterior, lo que hace es entrenar el modelo, para ajustarse mejor a las instancias que quedaron menos alineadas con el resultado en el paso anterior. Así, al cabo de varias iteraciones, todas las instancias estarán ajustadas.

En la Tabla 2-1 tenemos resumidos los anteriores algoritmos de ML supervisados, con sus siglas.

Algoritmos	Siglas
Regresión lineal	LR
Regresión logística	LLR
K-Nearest Neighbors	K-NN
Árboles de decisión o <i>Decision Trees</i>	DT
Naive Bayes	NB
Support Vector Machine	SVM
Random Forest	RF
Adaptative Boosting	AdaBoost
Gradient Boosting	GB

Tabla 2-1 Algoritmos de ML Supervisados. Fuente:[7]

Algoritmos No Supervisados:

- **K-means.** Uno de los algoritmos no supervisados más conocido y utilizado para tareas de *clustering*. Inicialmente se definen k clústeres aleatoriamente y se calcula el centroide de cada uno, como la media de las instancias pertenecientes a ese grupo. A continuación, se asigna cada elemento al clúster cuyo centroide esté más cercano y se vuelven a calcular centroides. Este proceso es iterativo, repitiéndose hasta que los centroides se mantengan estables. Este algoritmo tiene dos problemas que pueden provocar que el sistema falle: la ubicación inicial de los centroides y la elección de la cantidad de estos. Para evitar el primero, se puede ejecutar el modelo varias veces y comparar los resultados con una métrica denominada inercia, que es la media de las distancias de cada instancia a su centroide más cercano. Para el segundo, se pueden aplicar métricas que ayuden elegir el mejor número de centroides. Este modelo da buenos resultados, es rápido y escalable, pero tiene limitaciones sobre todo cuando los clústeres tienen tamaños variables, densidades dispares, o formas no esféricas.
- **Density-based spatial clustering of applications with noise (DBSCAN).** Es un algoritmo de *clustering* basado en la densidad de los datos, en el que los clústeres se definen como regiones continuas de alta densidad. Por cada instancia se cuenta el número de instancias están a menos de una pequeña distancia de ella, denominando a esta zona como su vecindario. Si el número es mayor de una cantidad dada, a esa instancia se la considera una instancia *core*. Todas las instancias del mismo vecindario pertenecen al mismo clúster. En este pueden coincidir varias instancias *core*, por lo que una secuencia de varias instancias *core* en el mismo vecindario darán lugar a un único clúster mayor. Las instancias que no son *core* y no pertenecen al vecindario de una instancia *core* se consideran anomalías. Este algoritmo funciona bien si los clústeres son lo bastante densos y están bien separados.

En la Figura 2-10 podemos observar el comportamiento de los algoritmos DBSCAN y k-means ante varios conjuntos de datos.

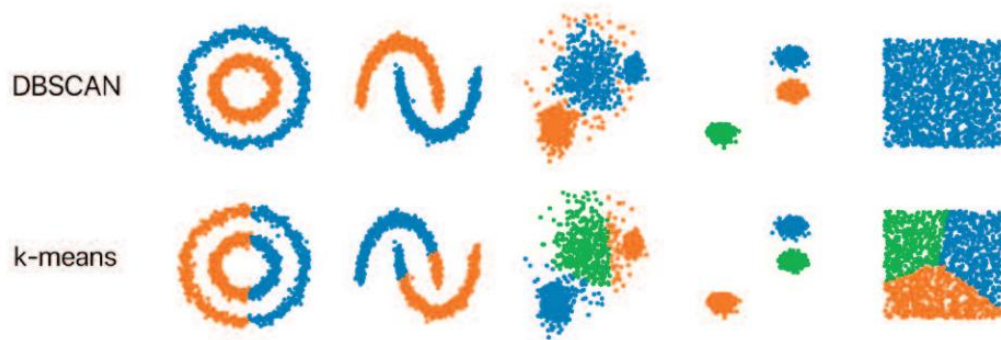


Figura 2-10 Comparación de agrupamiento de k-means con el de DBSCAN. Fuente:[8]

- **Hierarchical Cluster Analysis (HCA).** En este algoritmo se intenta construir una jerarquía de clústeres, bien sea agrupando elementos en base a una condición de similitud y volviéndolos a agrupar sucesivamente, o bien dividiendo un clúster único inicial en otros, de forma recursiva hasta cumplir un criterio determinado.
- **One-class SVM.** Un algoritmo SVM es un clasificador lineal que divide los datos en dos clases, como ya vimos anteriormente. Sin embargo, se puede utilizar para detectar anomalías si identificamos una sola clase y tratamos de encontrar una pequeña región que incluya a todas las instancias. Si aparece una instancia nueva que no cae dentro de esta región, se tratará de una anomalía. Funciona muy bien en conjuntos de datos con muchas dimensiones, pero no escala bien en datos muy grandes.
- **Isolation Forest (IF).** Es utilizado para la detección de anomalías y tiene muy buen comportamiento con grandes conjuntos de datos. Se basa en un algoritmo Random Forest en el que cada árbol crece de forma aleatoria. En cada nivel se elige una categoría y un valor de decisión aleatoriamente, hasta que todas las instancias quedan aisladas en un nodo cada una. Cuando se calcula la media de todos los árboles de decisión, del número de pasos en los que ha quedado aislada cada instancia, se observa que las anomalías quedarán aisladas en menos pasos que las instancias normales.

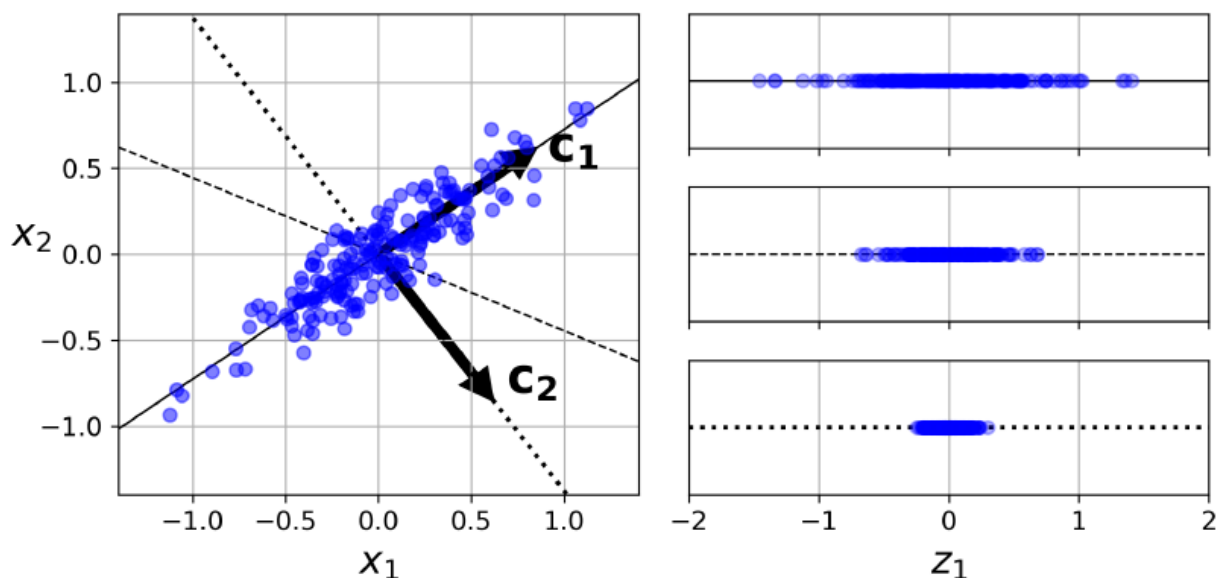


Figura 2-11 PCA, selección de hiperplano sobre el que proyectar. Fuente:[7]

- **Principal Component Analysis (PCA).** Es el algoritmo más popular para la reducción de dimensiones. Su objetivo es conservar las dimensiones que menos información pierden. Para ello, identifica el hiperplano que más cerca está de los datos, que también será el que

tiene mayor varianza y por lo tanto mayor distribución de valores en el eje. Es el componente principal. Si el modelo va a tener más dimensiones, buscará un segundo componente principal, un tercero, etc. A continuación, proyectará los datos sobre esos planos (Figura 2-11).

- **Locally-Linear Embedding (LLE).** Es otro algoritmo para reducir dimensiones. Se basa en capturar la distancia de cada instancia con sus vecinas más cercanas y luego encontrar una representación con menos dimensiones que preserve lo mejor posible estas características entre instancias próximas.
- **T-Distributed Stochastic Neighbor Embedding (T-SNE).** Este algoritmo es muy usado para visualizaciones de clústeres en espacios de muchas dimensiones. Su objetivo es reducir las dimensiones mientras intenta mantener las instancias similares cerca y las distintas alejadas.
- **Algoritmo a priori.** Primer algoritmo que se utilizó para el descubrimiento de reglas de asociación ocultas en los datos. Se basa en generar un conjunto de elementos más frecuentes que un umbral establecido y definir reglas de asociación a partir de ellos, seleccionando aquellas que tengan una alta probabilidad de ocurrir.
- **Equivalence Class Transformation.** Realiza una búsqueda en profundidad en los datos para encontrar los elementos más frecuentes. De esta forma se gana rapidez respecto al anterior algoritmo.

Estos algoritmos supervisados los tenemos relacionados, con expresión de la tarea que suelen realizar y sus siglas, en la Tabla 2-1.

Tarea	Algoritmos	Siglas
<i>Clustering</i>	K-Means	
<i>Clustering</i>	Density-based spatial clustering of applications with noise	DBSCAN
<i>Clustering</i>	Hierarchical Cluster Analysis	HCA
Detección de anomalías	One-class SVM	
Detección de anomalías	Isolation Forest	IF
Reducción dimensiones	Principal Component Analysis	PCA
Reducción. Dimensiones	Locally-Linear Embedding	LLE
Visualización y reducción	T-Distributed Stochastic Neighbor Embedding	T-SNE
Aprendiz. reglas asociación	Algoritmo a priori	A priori
Aprendiz. reglas asociación	Equivalence Class Transformation	ECLAT

Tabla 2-2 Algoritmos de ML No Supervisados. Fuente:[7]

2.3.4 Deep Learning

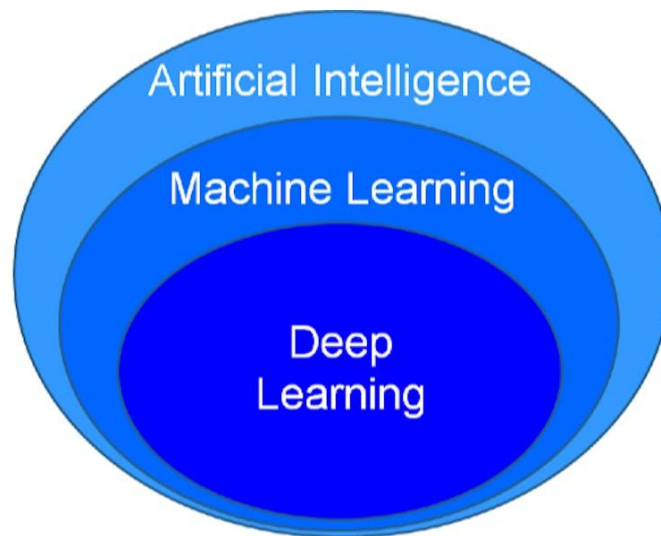


Figura 2-12 DL, subconjunto del ML. Fuente:[11]

El aprendizaje profundo, o Deep Learning (DL), es un subconjunto de algoritmos de ML (Figura 2-12) basados en una estructura jerárquica de redes neuronales artificiales (ANN), interconectadas y organizadas por capas, de tal forma que cada capa va alcanzando unos niveles mayores de abstracción.

Las neuronas artificiales se inspiran en la idea de las neuronas biológicas. Así, al igual que una neurona de un ser vivo tiene unas dendritas por las que recibe estímulos de múltiples células y un axón por donde la célula emite el resultado de su procesamiento, las neuronas artificiales reciben varias entradas y tras combinarlas producen una salida.

Una implementación práctica de estas células son las unidades lógicas de umbral (TLU) que funcionan con valores numéricos, no lógicos. Una TLU recibe varias entradas y a cada una se le asigna un peso. A continuación, suma el producto de cada entrada por su peso. Sobre esta suma aplica una función de paso y este es el resultado de la unidad (Figura 2-13). Viendo su estructura, una única TLU se podría usar para realizar una clasificación lineal como la del SVM que vimos anteriormente.

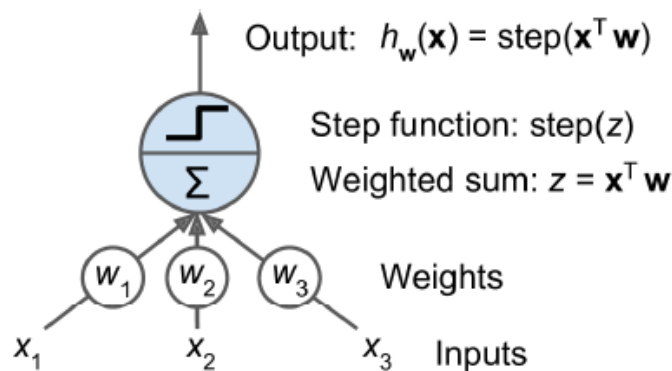


Figura 2-13 Unidad Lógica de Umbral TLU. Fuente:[7]

Una de las primeras arquitecturas y más simples de ANN es el perceptrón, que consiste en una única capa de TLU, cada una conectada a todas las entradas. Estas entradas conectadas a la red, a su vez, se suelen representar mediante unas neuronas, con una entrada y varias salidas, en una primera capa denominada capa de entrada, que no realiza cambios sobre los datos que reciben. Además se suele añadir una neurona adicional en esa capa, siempre con valor 1, denominada *bias* o balanceo. En la Figura 2-14 podemos ver un perceptrón con dos entradas y tres salidas.

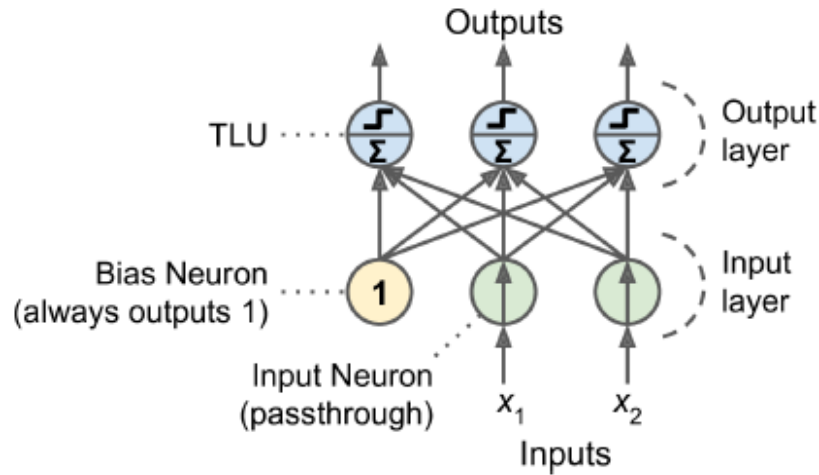


Figura 2-14 Perceptrón con dos entradas. Fuente:[7]

El perceptrón presentaba una serie de limitaciones que fueron solventadas apilando varios perceptrones dando como resultado una ANN denominada Multi-Layer Perceptron (MLP). Esta arquitectura ya posee la estructura típica de las redes neuronales: una capa de entrada, varias capas ocultas y una capa de salida. Además, cada capa, salvo la de salida, tiene una neurona adicional de balanceo (Figura 2-15).

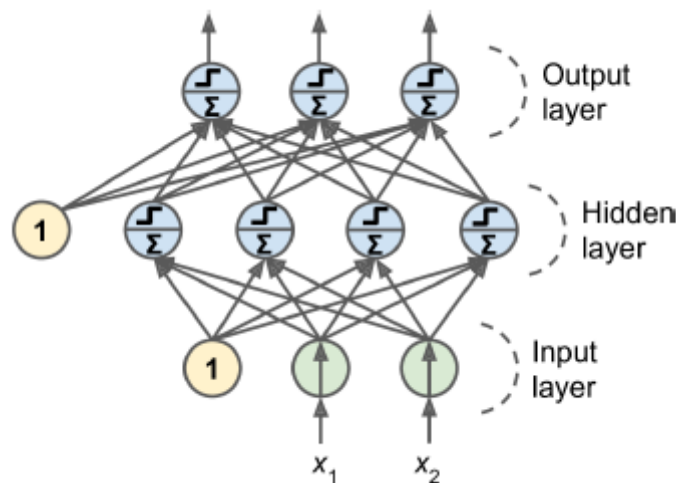


Figura 2-15 Multi-Layer Perceptron (MLP). Fuente:[7]

Sin embargo, este modelo no daba grandes resultados a los entrenamientos, y no fue hasta 1986, cuando se introdujo el algoritmo de entrenamiento **backpropagation**, que empezaron a producirse resultados relevantes. Este algoritmo es un Gradiente Descendente muy eficiente, que en sólo dos pasadas por la red, una hacia delante, ejecutando el algoritmo, y otra hacia atrás, verificando el nivel de acierto de cada célula, va identificando qué valores de balanceo y de pesos de cada conexión van reduciendo el error del modelo. Este proceso se repite hasta que la red converge en una solución. El algoritmo MLP suele ser usado tanto para tareas de regresión como de clasificación.

2.3.5 Arquitecturas de DL

Partiendo de las iniciales redes neuronales se han desarrollado gran cantidad de arquitecturas y algoritmos de DL en uso hoy en día, por lo que solamente se describirán brevemente algunas de las más utilizadas. En la Figura 2-16 se observa una clasificación de arquitecturas de DL, divididas en supervisadas y no supervisadas [12].

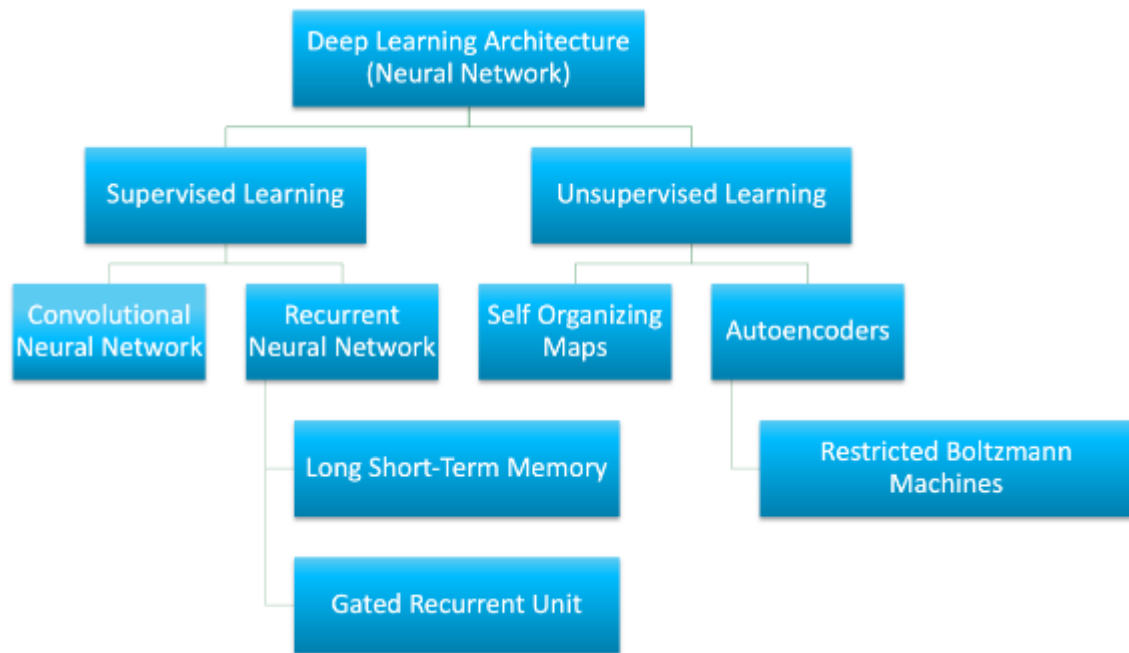


Figura 2-16 Arquitecturas de DL. Fuente:[12]

Aprendizaje Supervisado:

- **Redes Neuronales Convolucionales (CNN).** Tienen su principal aplicación en el reconocimiento de imágenes, de hecho sus primeros modelos nacieron para reconocimiento de caracteres escritos a mano en clasificación de códigos postales. Se inspira en la visualización de los seres vivos, donde las capas primeras identifican líneas, áreas, sombras, bordes, etc., y las internas recombina los anteriores elementos en figuras y objetos de nivel superior, identificando patrones. Se basa en un proceso de extracción de características en sucesivas capas convolucionales que van reteniendo las más importantes en cada convolución. Por último, se realiza una clasificación final del resultado. De gran aplicación en sistemas de reconocimiento facial y sistemas de conducción autónoma.
- **Redes Neuronales Recurrentes (RNN).** Es una de las primeras arquitecturas y punto de partida de muchas otras. Busca mejorar el comportamiento de la red permitiendo conexiones, no sólo en sentido ascendente, sino también con neuronas de la capa inferior o de su misma capa. Esto permite mantener la memoria de entradas anteriores. Usadas en aplicaciones relacionadas con el NLP, ya que se basan en el tratamiento de series temporales y datos secuenciales. Abordan problemas como la traducción de idiomas, el reconocimiento de voz o la captura de imágenes. Podemos encontrar esta tecnología en aplicaciones como Siri o Google Translate.
- **Long Short-Term Memory (LSTM).** Se basan en las RNN, pero introducen el concepto de célula de memoria. Ésta puede retener su valor durante un corto periodo de tiempo, lo que le permite a la célula recordar lo que es importante y no sólo lo último que calculó. Para ello la célula tiene tres puertas que controlan su comportamiento: puerta *input* que permite entrar información nueva, la puerta *forget* que controla cuando una puerta olvida su contenido y la puerta *output* que controla cuando la información de la célula sale (Figura 2-17). Todo esto se gestiona mediante pesos que abren o cierran estas puertas. Este modelo es muy utilizado en sistemas de NLP.
- **Gated Recurrent Unit (GRU).** Es una simplificación del LSTM, sólo tiene dos puertas *reset* y *update*. La primera indica cómo incorporar el contenido de la entrada y la segunda cuánto del contenido anterior mantener. Al ser más simple se puede entrenar más rápidamente y ser más eficiente, pero el LSTM puede llegar a mejores resultados.

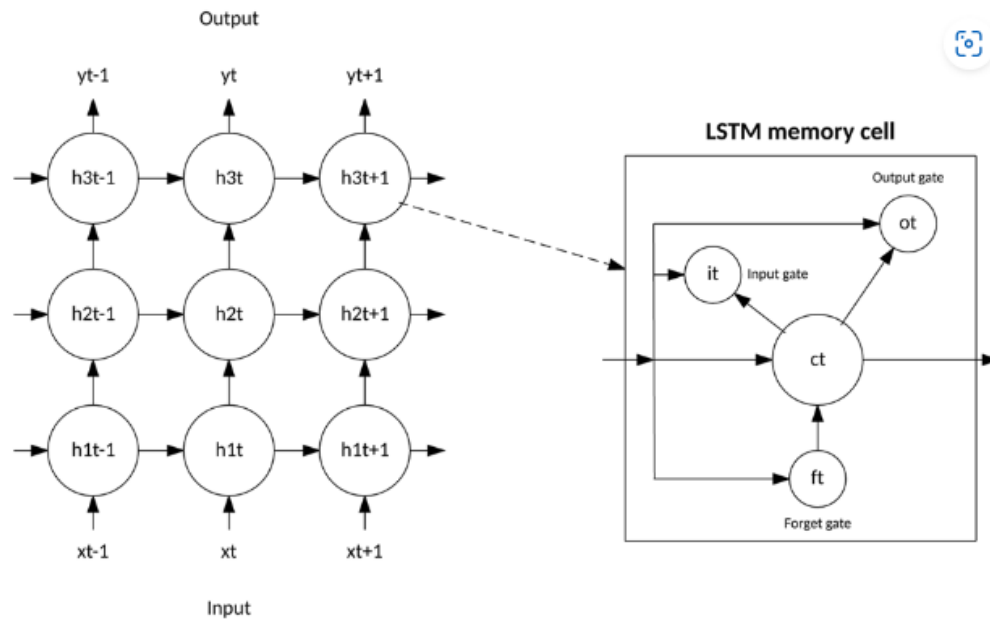


Figura 2-17 Red neuronal LSTM con detalle de una célula. Fuente:[12]

Aprendizaje No Supervisado:

- **Autoencoders.** Es una arquitectura con tres capas. La primera capa, de entrada, se codifica sobre la segunda, llamada oculta, con un número de nodos mucho menor. A continuación se vuelve a decodificar sobre la tercera capa, de salida. Por último se comparan las capas de entrada y salida, y se repite el proceso sucesivamente, modificando los pesos para reducir el error. Se utiliza para reducción de dimensiones, interpolación de datos y algoritmos de compresión.
- **Restricted Boltzmann Machines (RBM).** Es una red de dos capas, entrada y oculta, en la que todos los nodos de una están conectados con los de la otra. En cada entrenamiento se modifica aleatoriamente el sesgo de las células de entrada por lo que, para los mismos datos, cada vez se produce un resultado distinto. Es conocido como un modelo generativo. Se utiliza para reducción de dimensiones y filtrado colaborativo.
- **Redes de conocimiento profundo (DBN).** Es una red multicapa con muchas capas ocultas, en la que cada par de capas conectadas es una RBM. Cada capa RBM oculta es una representación abstracta de la anterior y es entrenada en modo no supervisado para reconstruir su propia entrada. La última capa hace una clasificación final en modo supervisado con datos etiquetados. Todo el sistema es entrenado realizando una propagación hacia el principio de los ajustes. Se suele utilizar en reconocimiento de imágenes, comprensión de lenguaje natural y predicción de fallos.
- **Redes apiladas profundas (DSN).** Se trata en realidad de un conjunto apilado de redes individuales de tres capas. Su objetivo es reducir la complejidad al entrenar una red neuronal con muchas capas ocultas, como las DBN, dividiéndola en varias redes más sencillas, que son entrenadas de forma individual y en paralelo. De esta forma se consiguen resultados similares con mucho mejor rendimiento. Se utilizan en sistemas de recuperación de información y reconocimiento de lenguaje hablado.
- **Redes neuronales generativas adversarias (GAN).** La idea principal consiste en utilizar dos redes neuronales y hacerlas competir entre ellas, de tal forma que una genera un contenido, y a la otra se le muestran datos reales y generados, teniendo que decidir si son reales o no. El objetivo es crear simulaciones realísticas, como imágenes, vídeos, audios o textos. También se han utilizado en la creación de mundos virtuales y *deepfakes*.

2.4 Procesamiento de Lenguaje Natural

El NLP es una rama de la IA que tiene como objetivo conseguir que las máquinas puedan comprender, tratar y utilizar el lenguaje humano. Para ello, se apoya en la lingüística computacional y en modelos estadísticos de ML y DL [13].

Los humanos nos comunicamos y almacenamos gran parte de nuestro conocimiento mediante el lenguaje en los distintos idiomas. Sin embargo, este lenguaje natural tiene una estructura que no es adecuada para las máquinas, aunque en su interior contiene mucha información que se podría tratar por éstas de forma automática. Este es el principal reto del NLP, convertir estos datos no estructurados, y extremadamente complejos, en estructurados, dividiéndolo en frases, palabras, entendiendo las relaciones entre ellos y su significado en el sentido más amplio. Una vez conseguido esto, los ordenadores podrán realizar tareas con estos textos, como clasificarlos, extraer información concreta, interpretarlos o analizar sentimientos, Y todo ello de forma mucho más rápida e imparcial que los humanos.

El procesamiento, no obstante, es muy complejo porque los datos están codificados en múltiples idiomas y dialectos, con reglas de ortografía y gramática muy distintas, con signos diversos de escritura y significados que cambian dependiendo de la situación y la intención. Además, para mayor complicación, al escribir, los humanos cometemos errores ortográficos, usamos abreviaturas, omitimos signos de puntuación y tenemos estilos propios individuales. Y en el lenguaje hablado es aún peor, pues tartamudeamos, mascullamos, usamos muletillas, incluso incorporamos palabras prestadas de otros idiomas [14].

Todo esto obliga a realizar un tratamiento sobre los datos que los normalice a la vez que reduce lo máximo posible las irregularidades propias del lenguaje humano. A la anterior tarea tenemos que añadir su conversión a un formato adecuado para su tratamiento automático por parte de algoritmos de ML, de forma que permita extraer sus características eficientemente para su posterior entrenamiento y obtención de resultados, esto es la vectorización, que consiste en la representación matemática del texto en un espacio numérico multidimensional.

2.4.1 Tareas habituales en el NLP

El NLP engloba multitud de tareas con objetivos diversos. Algunas de las más frecuentes están relacionadas con el preprocesamiento de los textos para su estructuración y formato:

- **Segmentación del texto.** División en frases.
- **Tokenización.** Partición en palabras o *tokens*.
- **Truncamiento o Stemming.** Reducción de la palabra a su raíz, truncando las derivaciones.
- **Lematización.** Reducir las palabras hasta su lema o raíz canónica en diccionario.
- **Etiquetado gramatical.** Identificar la función de una palabra o texto dentro del discurso.
- **Desambiguación del sentido de las palabras.** Algunas palabras cambian de significado según el contexto o su función.
- **Reconocimiento de entidad nominada.** Identificar palabras o frases con su utilidad o categoría. Por ejemplo nombre de persona, lugar, medio de transporte, etc.
- **Resolución de correferencia.** Detectar si dos palabras se refieren a la misma entidad en el texto. Normalmente pronombres, equivalentes o metáforas.

A un nivel más alto de abstracción nos encontramos con otro tipo de tareas más complejas. Entre ellas podemos citar las que están expresamente dirigidas a la comunicación con humanos, como el reconocimiento de voz, consistente en convertir la voz a texto de forma fiable, o la generación de textos en base a una solicitud recibida. En la primera, la dificultad reside principalmente en el habla de cada persona, su rapidez, entonaciones, acentos o gramática incorrecta. En cuanto la segunda, la generación de textos a semejanza de los utilizados en la comunicación humana es probablemente una

de las aplicaciones del NLP que más rápida difusión ha tenido en los últimos años. En este grupo también podríamos incluir los *chatbots* y asistentes virtuales.

Otras tareas están más relacionadas con la extracción de características que permitan una mejor comprensión. Por ejemplo, tenemos enfoques para categorización del contenido, que tratan de hacer un resumen del texto con índice, palabras clave, detección de duplicados, avisos de contenido, etc. Otros, como el análisis de estructura, sirven para hacer modelado de los documentos y extraer información estructurada de ellos. Muy relacionados estarían los resúmenes automáticos y la traducción para aumentar la efectividad y la comprensión la hora de la lectura.

Por último, hay otro grupo que también se basa en la extracción de propiedades del texto y está dirigido a la clasificación automática de los documentos. Así tenemos, filtrado de correos, detección de fake news o gestión automática de archivos, que detectan patrones identificativos y actúan conforme a las instrucciones indicadas.

2.4.2 Representación del texto en vectores numéricos

El lenguaje natural, utilizado por los humanos no puede ser procesado directamente por las máquinas mediante algoritmos de ML, necesita ser convertido a su lenguaje, los números. Para ello, una vez recolectado los datos deseados, es necesario crear un **corpus**, que consiste en un conjunto de textos codificados en vectores numéricos, de tal forma que puedan ser procesados por ML.

Para crearlo, primero realizaremos una segmentación (tokenización) de los textos en oraciones y luego en palabras, eliminando caracteres de puntuación. A continuación, se puede hacer una lematización o reducción de las palabras a sus raíces, eliminando conjugaciones, plurales, etc. Lo siguiente sería eliminar las denominadas *stopwords*, palabras que no aportan gran significado al documento y que además aparecen con mucha frecuencia, es el caso de artículos, preposiciones, etc. Llegados a este punto podemos realizar la codificación numérica del texto en vectores numéricos de múltiples dimensiones. Esta representación se puede realizar a nivel palabra o carácter, ambas tienen sus ventajas e inconvenientes, pero nos centraremos en la codificación de palabras por ser la más frecuente en el ámbito abordado por este trabajo.

"apple"	"computer" ...	"cheese"	"apple"	"computer" ...	"cheese"
0	0	0	0.02	0.84	0.13
1	0	0	0.13	0.22	-0.16
0	1	0	-0.56	-0.75	-0.21
...	...	...	...	...	...
0	0	1	1.24	-0.44	0.34
0	0	0	-0.7	0.56	-0.05

Figura 2-18 Representación vectores *one-hot encoding* y *embedding*. Fuente:[15]

La representación vectorial más sencilla es la conocida como ***one-hot encoding***. Consiste en crear un vocabulario con todas las palabras distintas del corpus y representar cada palabra del texto con un vector con tantas dimensiones como elementos tenga el índice, rellenos todos a cero menos el correspondiente a la posición asignada a la palabra que se quiere representar, que tendrá un uno. Por

ejemplo, si tenemos un vocabulario de diez mil palabras, para representar un texto necesitaríamos un vector con diez mil dimensiones por cada palabra, casi en su totalidad relleno con ceros.

Este método es muy ineficiente a la hora de representar la información pues precisa mucha memoria y no representa las posibles relaciones entre los elementos. Para llegar a un modelo más optimizado se precisa aplicar unas técnicas que permita compactar este modelo, es decir, reducir el número de dimensiones de estos vectores con la menor pérdida de información, es lo que conocemos como **word embedding**. En la Figura 2-18 se observan tres palabras representadas en *one-hot encoding* y esas mismas con sus valores actualizados tras efectuar un *embedding*.

2.4.3 Técnicas de word embedding

Para realizar la codificación en este vector numérico existen varias técnicas de *word embedding* que se pueden clasificar atendiendo a distintos criterios [16], tal y como se puede observar en la Figura 2-19.

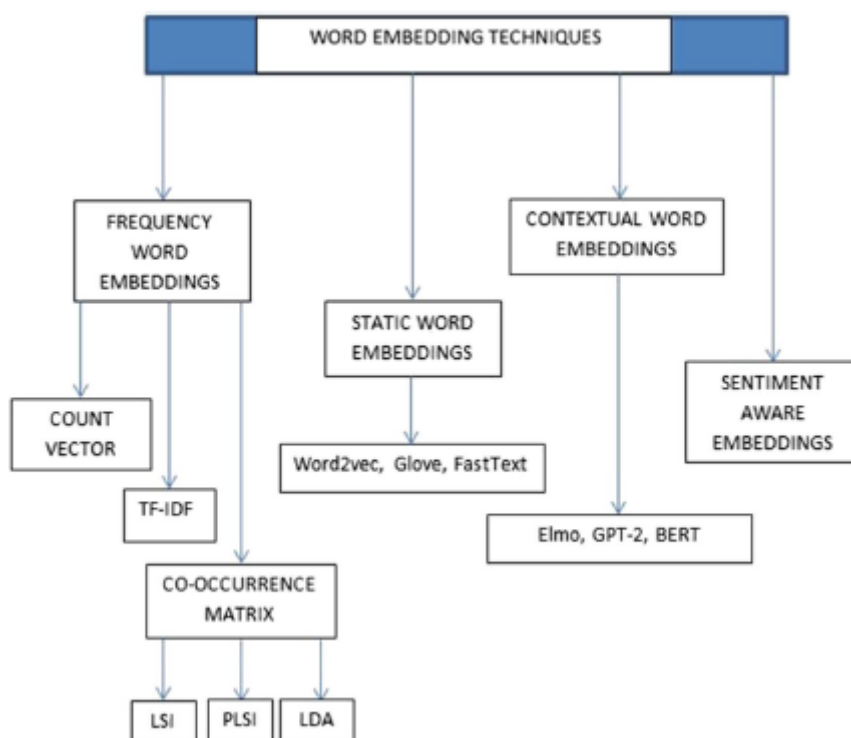


Figura 2-19 Clasificación de técnicas de *word embedding*. Fuente:[16]

Las primeras y más sencillas son las que se basan en la **frecuencia** con que aparecen las palabras en el texto:

- **Bag of words (BoW).** Partiendo de la matriz booleana de *one-hot encoding*, totalizamos el número de ocurrencias de cada palabra del vocabulario y obtenemos un vector con las frecuencias absolutas de cada una. Es realmente muy sencillo, pero solamente sirve para una primera aproximación y hacernos una idea del contenido de los documentos del corpus ya que no contempla la posición de las palabras ni la relación entre ellas. Además habrá palabras que aparezcan muchas veces sin ser especialmente relevantes o al revés.
- **Term Frequency – Inverse Document Frequency (TF-IDF).** En realidad es una mejora de BoW en la que, en lugar de almacenar el número de veces que aparece cada palabra, se utiliza una fórmula con la frecuencia de aparición de la palabra en el documento penalizada si la palabra aparece en muchos documentos del corpus. Esto nos permite reducir la importancia de las palabras comunes y muy frecuentes en el corpus, que no aportan gran significado al documento, pero seguimos sin tener correlación entre las palabras.

- Hay una variante de estos métodos, *n-grams*, en la que en vez de guardar la estadística de cada palabra se guardan las de n elementos consecutivos. En este caso sí se consigue almacenar algo más de significado, pero muy limitado, pues sólo es operativo para palabras contiguas. Además está la dificultad de elegir que bigramas o trigramas de palabras de los posibles son más representativos.

Los algoritmos anteriores tenían el problema de perder contenido semántico de las palabras por no tener información sobre su contexto. Para mantenerlo en cierta medida, a la vez que se reducen las dimensiones de los vectores, hay un conjunto de técnicas que se basan en la factorización de las palabras mediante **matrices de coocurrencia** o correlación, que registran cuántas veces dos términos aparecen juntos en un contexto. El objetivo es comprimir la información más relevante de una matriz con una dimensión muy alta y con información muy dispersa, en matrices de dimensión mucho menor donde se guarda la probabilidad de coincidencia de unos términos con otros.

- **Latent Semantic Analysis (LSA).** Intenta solventar dos problemas de las técnicas del grupo anterior: sinonimia y polisemia. Se basa en el concepto de que dos palabras con el mismo significado aparecerán con valores similares de correlación con los mismos términos, serán sinónimas, o el mismo valor singular (SVD). Por otro lado, si una palabra parece pertenecer a dos clústeres diferentes de correlaciones es muy probable que sea una palabra con dos significados. En este caso se hace una descomposición en dos SVD, creando un nuevo espacio que será una variación del espacio anterior, resolviendo el problema del sinónimo. Su principal dificultad reside en añadir nuevos documentos al modelo pues las palabras nuevas tendrán que actualizar las estadísticas de todas las demás.
- **Probabilistic LSI (PLSI).** Es una variante del anterior con un enfoque probabilístico. En vez de dividir el documento en SVD, asume que un documento es un proceso probabilístico asociado a unos temas que va generando palabras relacionadas con cada uno de esos temas. A pesar de su flexibilidad tiene ciertos problemas, como la dificultad para representar tópicos relacionados o que el número de parámetros crece linealmente con el número de documentos.
- **Latent Dirichlet Allocation (LDA)** Al igual que el anterior, es un modelo estadístico que supone que los documentos tienen temas y las palabras pertenecen a estos. La diferencia radica en que se supone que hay un número concreto de temas en todo el corpus y que cada documento es una combinación de esos temas. Surge para solventar el problema de PLSI en el que los parámetros crecían linealmente con la cantidad de documentos. Se utiliza en numerosas aplicaciones como agrupación de documentos, clasificación de textos o recomendación de contenidos.

Otro grupo de algoritmos es el de los **word embeddings estáticos**. Estos, mediante un algoritmo entrenado para una tarea concreta, convierten cada palabra en un vector con valores continuos de menor número de dimensiones, pero más denso. Se les denomina estáticos porque utilizan un tamaño de vocabulario fijo. Hay varios modelos, pero comentaremos los más conocidos.

- **Word2vec.** Apareció en 2013 de la mano de ingenieros de Google y es probablemente el modelo más popular, y uno de los más eficientes y predictivos para realizar *word embedding*. Se basa en la idea de que las palabras se pueden representar mediante un vector definido con la información proporcionada por las palabras de su contexto, si se entrena un corpus lo suficientemente grande. Para ello suele utilizar redes neuronales recurrentes porque permiten que la información vaya quedando residualmente en las células, mejorando el resultado.
La idea principal es que dos palabras con un contexto muy parecido tienen que tener vectores muy próximos en el nuevo espacio escalar. Además, encontraron que este sistema

de representación, si se utilizaba con corpus de gran tamaño, era capaz de capturar reglas semánticas y sintácticas interpretables de una forma bastante sencilla, basándose en la comparación de los vectores. Por ejemplo, observaron que habían capturado la regla de los plurales porque si comparaban la relación entre los vectores de la palabra *car* con *cars*, era la misma que la de *apple* con *apples* y la de *family* con *families* [17]. En la Figura 2-20 se observa cómo la relación entre los vectores *man* y *woman* es similar a la de los vectores *uncle* y *aunt*, o *king* y *queen*. Así mismo, para desplazar el vector *king* hasta *kings* aplicaríamos el mismo vector que de *queen* a *queens*.

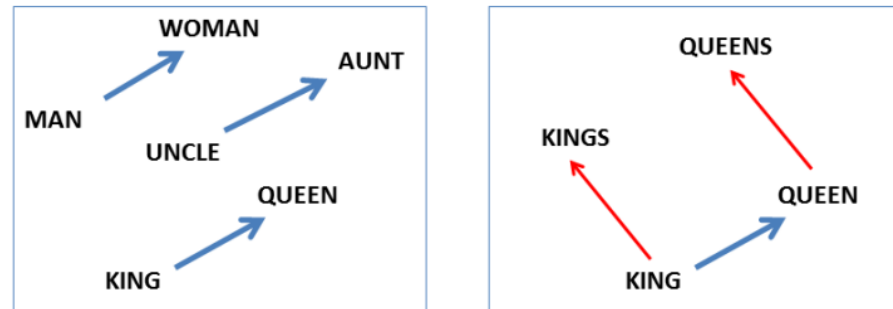


Figura 2-20 Vectores con relaciones sintácticas y semánticas. Fuente:[17]

Tratando de minimizar la complejidad computacional y utilizando para el aprendizaje RNN, que recorrieran el texto secuencialmente, se adoptaron dos arquitecturas que intentaban ser lo más simples posible.

- **Continuous Bag of Words (CBOW).** Para cada conjunto de palabras de un radio determinado que define el contexto, se intenta predecir cuál sería la palabra central de ese conjunto, como se representa en el diagrama que aparece a la izquierda en la Figura 2-21.
- **Skip-Gram.** Al contrario del anterior, para una palabra determinada se intentaba predecir las palabras más probables que la podrían acompañar, o sea su contexto. A la derecha en la Figura 2-21.
- **Global Vectors for Word Representation (GloVe).** Es un modelo desarrollado en Stanford en 2014 con el fin de aprender representaciones vectoriales de palabras a partir de grandes corpus de texto. Mientras que su predecesor en esta tarea, Word2vec, utiliza una ventana, es decir, escanea el contexto de esa ventana recorriendo todo el corpus, utilizando redes neuronales recurrentes para entrenarlo, Glove es un modelo estadístico que se basa en las matrices de coocurrencias de todo el corpus. En esta matriz registra la frecuencia con que dos palabras aparecen juntas dentro de un contexto. El entrenamiento consiste en encontrar, mediante cálculo de probabilidades, la combinación de vectores de dimensiones reducidas que mejor se ajusten a las relaciones entre palabras definidas en la tabla de coocurrencias. Si bien es más eficiente computacionalmente que Word2vec, éste último suele dar resultados más precisos.
- **FastText.** Este modelo de *word embedding* fue desarrollado por Facebook y presentado por primera vez en 2017. Al igual que word2vec, busca generar representaciones vectoriales entrenando los vectores de un vocabulario, pero permite trabajar a nivel de subpalabras.

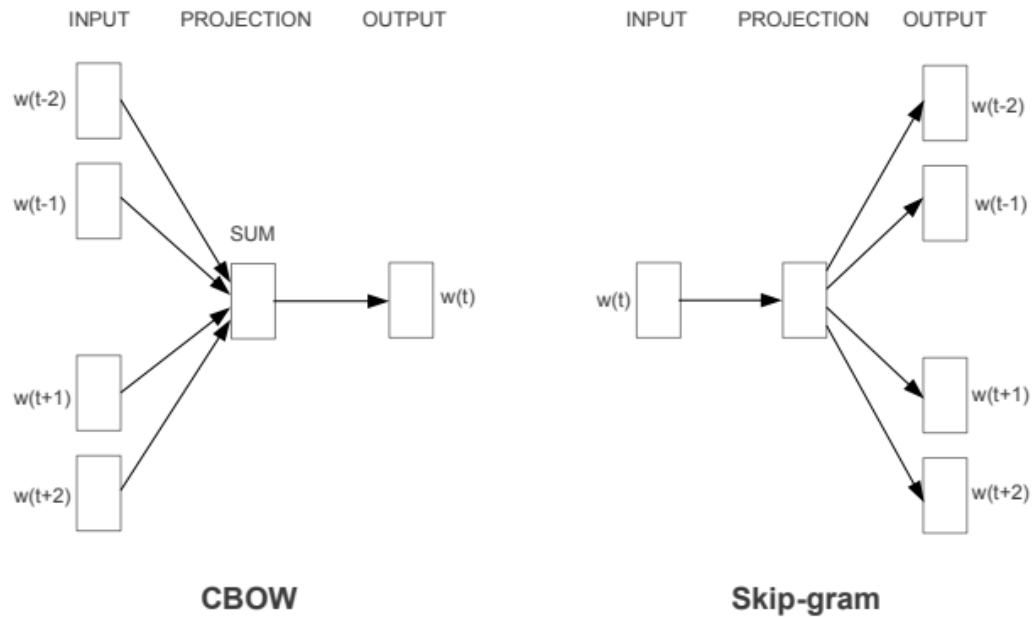


Figura 2-21 Arquitecturas utilizadas en el modelo Word2vec. Fuente:[18]

Un grupo de técnicas muy extendido en la actualidad es el de los **métodos contextuales**, que parten de la premisa de que lo que define en realidad una palabra es su contexto, no partiendo de ningún algoritmo de *embedding* inicial que limite el contenido original del texto a tratar.

Como hemos comentado anteriormente, las RNN son una de las arquitecturas subyacentes más utilizada para el entrenamiento de los métodos de *word embedding*. Su funcionamiento se asemeja a la forma secuencial en que se leería un texto por un humano. En estas redes, la salida del procesamiento de una palabra se añade a la entrada del paso siguiente (Figura 2-22). De esta forma se consigue que, de alguna forma, se vaya acumulando la información del contexto en el cálculo de cada célula. Sin embargo, este peso va disminuyendo conforme se van distanciando las palabras. Esta pérdida de memoria es lo que se conoce como desaparición de gradiente.

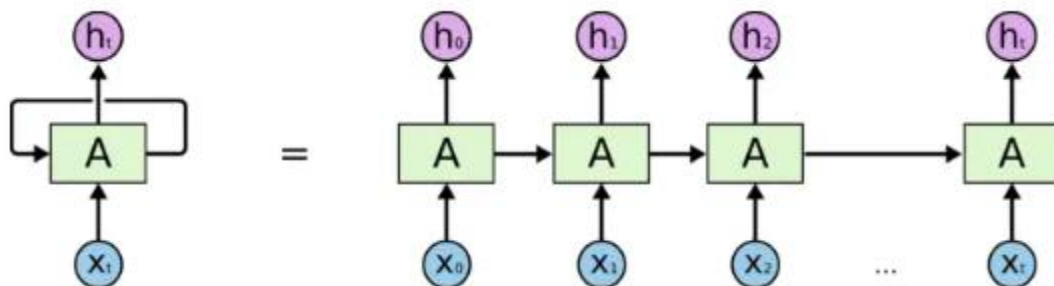


Figura 2-22 Red Neuronal Recurrente desplegada en el tiempo. Fuente:[19]

Para minimizar este problema de pérdida de memoria de las RNN se definió un algoritmo denominado **Mecanismo de Atención** que permitía que determinadas palabras mantuvieran un peso alto en relación con la que se estaba tratando en la RNN a pesar de su distanciamiento en el texto. Para ello se entrenan dos redes neuronales que tienen por objetivo generar dos vectores por cada palabra del texto o vector *value*. Una generará un vector con los valores que definen mejor la palabra, es el vector *key* y la otra generará un vector que mejor define la palabra que encaja con el contexto de la palabra clave, es el vector *query*. A continuación se realiza el producto escalar entre el vector *query* de la palabra tratada en la RNN y los vectores *key* de todas las demás del texto. De esta forma obtenemos un

nuevo vector por palabra al que denominaremos vector de atención. Los valores más altos de atención serán los más adecuados para reforzar su peso en el algoritmo aunque estén alejados en el texto.

En el año 2017 apareció un estudio “Attention is all you need” que revolucionó lo que se estaba haciendo en ese momento y decía que, sin utilizar redes neuronales recurrentes ni convolucionales, sólo con mecanismos de atención, habían conseguido mejores resultados y más rápido, y proponía una nueva arquitectura de red neuronal más sencilla, los **transformers** [20].

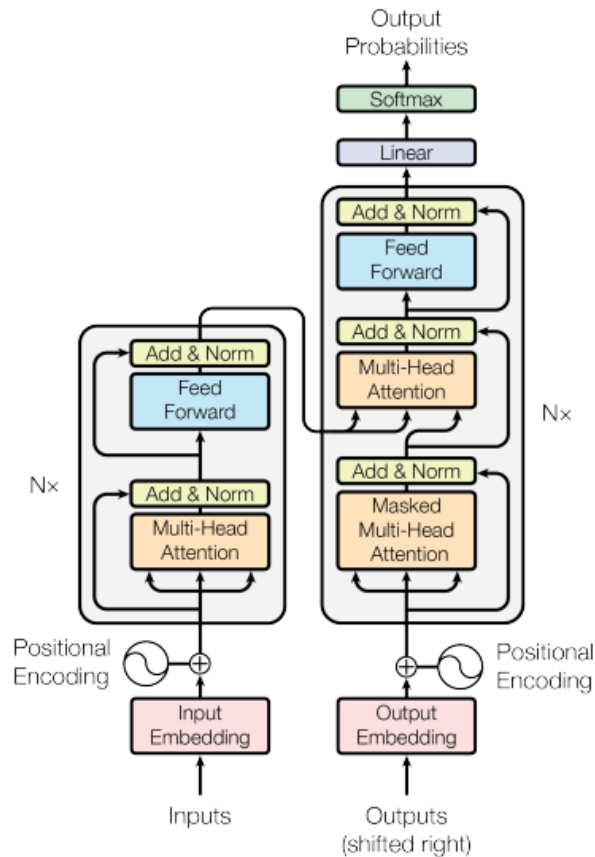


Figura 2-23 Arquitectura transformer. Fuente:[20]

Como podemos apreciar en la Figura 2-23, esta arquitectura tiene dos bloques principales el **encoder** y el **decoder**. El **encoder**, a la izquierda, recibe la entrada a procesar y genera el vector de valor. A continuación, construye los vectores de atención basándose en la relevancia de cada palabra respecto a las demás. Seguidamente se procesan estos datos en más capas neuronales que permiten encontrar patrones más complejos y se realizan conexiones entre capas para evitar desvanecimiento de gradiente. Por último, se normaliza la representación, una vez realizado el aprendizaje, y genera la salida. El **decoder**, igual que el **encoder**, recibe una entrada que sería el objetivo que se quiere conseguir, para utilizarlo en el entrenamiento. Por ejemplo, en una traducción sería la palabra en el idioma de destino. Sin embargo, este componente utiliza dos mecanismos de atención, uno con la entrada de objetivos y otro con las representaciones aprendidas de la salida del **encoder**. De esta forma, encaja lo aprendido en el **encoder** (texto traducido por el modelo) con lo aprendido en el **decoder** (texto traducido externo), aplicando las reglas de un modelo sobre otro. Después aplica capas de procesamiento neuronal para capturar patrones, igual que el **encoder** y por último normaliza y genera la salida final.

Esta arquitectura, además de ser computacionalmente más eficiente, tiene la ventaja de permitir la entrada en bloque de varias palabras a las que se les añade información sobre su posición dentro de la secuencia. Esto aumenta aún más su potencia al permitir procesamiento en paralelo.

La ventaja que proporcionó la arquitectura *transformer* al permitir procesar en paralelo numerosas entradas sin necesidad de recorrer secuencialmente los documentos y su eficiencia de cálculo con el modelo de atención, fomentó que aparecieran nuevos modelos basados en esta tecnología.

- **Embeddings from Language Models (ELMo).** Fue desarrollado por la Universidad de Allen en 2018 y consiste en un modelo contextualizado en ambas direcciones. Para su funcionamiento utiliza una red neuronal LSTM bidireccional. Se ha entrenado con grandes conjuntos de datos dando muy buenos resultados. Es especialmente preciso a la hora de identificar polisemia adaptando el significado real de la palabra al contexto de la frase.
- **Bidirectional Encoder Representations from Transformers (BERT)** Es un modelo generado por Google, basado en la arquitectura de *transformers* descrita anteriormente que tiene como objetivo interpretar el lenguaje de búsqueda de una forma más natural. Para ello, además de entender mejor la estructura del lenguaje, utiliza la bidireccionalidad a la hora de analizar una frase. Es decir, lo que viene delante de la palabra y lo que viene detrás de ella. Para su entrenamiento utiliza técnicas como la ocultación de palabras y la predicción de la siguiente oración.
- **Generative Pretrained Transformer (GPT).** Desarrollado por OpenAI en 2018. Al igual que BERT se basa en la arquitectura de *transformers* y también se ha entrenado con gran cantidad de documentos sin supervisión. Es autoregresivo, es decir, realiza predicciones de una palabra en función de las anteriores, por lo que no es bidireccional como BERT o XLNet.
- **XLNet.** Fue creado por Google y Carnegie Mellon University en 2019. Utiliza los *transformers* en su modelo, como los anteriores, y también es bidireccional. Como peculiaridad utiliza la permutación de palabras en el entrenamiento, esto es, que predice las palabras permutando el orden, o sea todas las palabras del documento con todas las demás.

Por último hay un grupo de métodos que podríamos denominar **conscientes del sentimiento** que intentan solventar el problema que tienen los algoritmos basados en contexto para distinguir las palabras que, aun teniendo el mismo contexto, tienen polaridades opuestas, y por lo tanto deben estar alejadas en el espacio vectorial. Para ello se utiliza un entorno híbrido de aprendizaje supervisado y no supervisado en el que se incluyen etiquetas con la polaridad para entrenar el sistema.

2.4.4 Modelos enormes de lenguaje

Los algoritmos anteriores, basados en la arquitectura de *transformers*, con su mejora de rendimiento computacional y procesamiento en paralelo, y los avances tecnológicos del momento, han permitido que se pudieran generar modelos enormes de lenguaje (LLM) preentrenados en la realización de tareas muy generales, de tal forma que puedan ser utilizados para desarrollar otras tareas más específicas, aprovechando el conocimiento sintáctico y semántico de los lenguajes que ya tienen adquirido estos modelos.

En la Figura 2-24 se pueden visualizar los principales LLM existentes en noviembre de 2023, con distintos colores según su constructor y tamaños proporcionales al número de parámetros. Se hace notar que el número de parámetros, reflejado en el eje de ordenadas de la gráfica, está dimensionado en *billions*, haciendo referencia, en este caso, a la notación anglosajona equivalente a miles de millones en notación española. En el resto del documento se ha utilizado la notación española.

- Cargar e iniciar el modelo LLM preentrenado que tomamos como base, por ejemplo BERT con sus parámetros predefinidos.
- Realizar el afinado o *fine-tuning* del modelo. Entrenar el modelo sobre el conjunto de datos propios y con una función de coste específica.
- Evaluar el rendimiento del modelo y ajustar hiperparámetros del modelo hasta que alcance un nivel adecuado.
- Evaluar el modelo sobre los datos de prueba y medir rendimiento.
- Implementar el modelo ya ajustado para la realización de la nueva tarea.

2.5 Detección de fake news

La detección de fake news de forma rápida y eficaz es un asunto de vital importancia para nuestra sociedad, pues vivimos en una época en la que la información se distribuye de forma exponencial, y si esta es falsa sus consecuencias pueden ser desastrosas, desde hundir la candidatura de un aspirante a gobernar el país más poderoso del mundo, a un ataque de pánico en el mercado de valores, pasando por el rechazo de la población a recibir un tratamiento médico.

2.5.1 Combatir las fake news individualmente

La mejor forma de combatir las fake news es evitar su propagación y para ello necesitamos tener la suficiente información, prudencia y criterio para no reenviarlas sin comprobarlas antes. El problema es que no es tan sencillo saber si una noticia que llega a nuestro alcance es falsa o no, pues no viene con una etiqueta que garantice su certeza. Entonces, ¿cómo podemos saber si es falsa? El Parlamento Europeo nos propone un conjunto de medidas que podemos realizar personalmente para comprobarlo antes de difundirlas [22]:

- **Comprueba el contenido.** Las noticias deben llevar hechos y datos precisos que se puedan comprobar. Además, las noticias de los medios de comunicación fiables suelen ser imparciales, publicando los artículos tendenciosos en la sección de opinión.
- **Comprueba el medio.** Verifica desde dónde lo estás viendo, si es un sitio conocido o no.
- **Comprueba el autor.** Si es un periodista o escritor conocido, o ni siquiera está firmado.
- **Comprueba las fuentes.** ¿Las fuentes citadas son fiables? Medios de comunicación, expertos en la materia fiables o directamente son anónimos.
- **Comprueba las imágenes.** Se pueden manipular muy fácilmente, pero si es así existirán los originales en otro contexto. Hay herramientas automáticas que ayudan a detectarlas.
- **Piensa antes de compartir.** Puede que lo estemos malinterpretando o que esté diseñado para provocar emociones intensas. Hay que ser racionales.
- **Cuestiona tus propias ideas preconcebidas.** Si algo es demasiado afín a nuestras ideas o creencias, para un segundo a verificarlo antes de reenviarlo.
- **Mantente al tanto de los últimos engaños.** Cuando sepas que una información es falsa, dilo para avisar a los demás.

2.5.2 Verificación de hechos o Fact-checking

Otra herramienta importante para hacer frente a esta amenaza es disponer de plataformas donde poder verificar si una noticia que se está difundiendo rápidamente es una fake news. Esto es lo que se conoce como verificación de hechos o *fact-checking*.

Existen numerosas agencias de verificación, que tanto a nivel nacional como internacional, realizan esta tarea. La tipología es variada, las hay independientes o asociadas a otras entidades, como medios de comunicación, privadas o públicas, generalistas o temáticas, etc.

Como organización de referencia internacional está la **Internacional Fact-Checking Network** (IFCN) del Instituto Poynter, creada en 2015 que, con más de 100 organizaciones periodísticas

asociadas de todo el mundo, tiene como misión promover la verificación de las noticias para la lucha global contra la desinformación (Figura 2-25).



Figura 2-25 Sitio web del International Fact-Checking Network. Fuente:[23]

Según un artículo del Instituto de Español de Estudios Estratégicos (IEEE), en España, hay tres agencias miembro del IFCN actualmente: la Agencia EFE, Newtral y Maldita.es [24]. Además de éstas, existen otras, de menores dimensiones, que no han firmado el compromiso ético del IFCN, y por lo tanto no se pueden considerar miembros, como por ejemplo, La chistera o Bendita.eu.

A la información proporcionada por todas estas agencias cabe hacerles una reseña importante. Como se vio en anteriores capítulos, todos tenemos sesgos cognitivos de distinta índole y, lógicamente, estas agencias y los periodistas que trabajan en ellas también los tienen. Por lo tanto, la información de verificación proporcionada también estará sesgada. La única forma de poder comprobar los hechos que queramos verificar es contrastar la información en varias plataformas, a ser posible de tendencias ideológicas distintas.

Algunas de las principales agencias de verificación de noticias serían:

- **FactCheck.org.** EEUU, 2003. Es la primera que surgió como tal para reducir el nivel de engaño y confusión en la política de los Estados Unidos. Es un proyecto de la Universidad de Pensilvania y trata temas variados de actualidad como política, salud, conflictos internacionales o cambio climático.
- **PolitiFact.** EEUU, 2007. Pertenece al Poynter Institute, organización sin ánimo de lucro de San Petersburgo (Florida). Su foco se centra en las declaraciones relacionadas con la política en los Estados Unidos. Tiene un Truthometer donde cada noticia recogida se califica en distintos grados de certeza.
- **The Fact Checker.** EEUU, 2007. Perteneciente a The Washington Post, es una columna de este medio con el propósito de remarcar la verdad en declaraciones políticas sobre cualquier tema conflictivo de actualidad. Especialmente, intenta aclarar temas complejos, dados al engaño y a la ambigüedad en manos de los políticos.
- **Channel 4 Fact Check.** Reino Unido. Es una rama del canal de televisión Channel 4 on line, que intenta llegar a la verdad de las declaraciones hechas por los cargos públicos. Tiene bastante prestigio por su integridad.
- **Chequeado.** Argentina, 2010. Inspirado en los anteriores, es un proyecto de La Voz Pública, fundación sin ánimo de lucro de Buenos Aires. Fue el primer sitio de América

Latina dedicado a este fin. Se interesa por políticos, economistas, empresarios, periodistas, medios de comunicación u otros formadores de opinión en Argentina. Publica y clasifica esta información en nueve categorías, desde verdadero hasta falso, pasando por engañoso entre otros.

- **Newtral.** España, 2018. Fundada por Ana Pastor, no tiene accionistas ni forma parte de ninguna organización empresarial. Compuesta por profesionales liberales como periodistas, investigadores, ingenieros o programadores, es una productora de medios audiovisuales y una agencia de verificación de hechos. Forma parte de numerosos proyectos divulgativos y de formación, nacionales e internacionales. Recientemente ha abierto una nueva vía relacionada con la IA y el *fact-checking*.
- **Maldita.es.** España, 2018. Fundada por dos periodistas procedentes de La Sexta TV se financia mediante colaboraciones con distintos medios de comunicación y alianzas tecnológicas. Tiene varias áreas temáticas que abordan la desinformación desde distintos puntos: Maldita Hemeroteca, Maldito Bulo, Maldita Ciencia, Maldito Dato y Maldita Tecnología. Cada una de estas ramas muestra una relación de temas de actualidad y noticias con sus claves y publicaciones falsas relacionadas.
- **EFE Verifica.** España. Es un servicio de la Agencia EFE, agencia estatal de noticias perteneciente a la SEPI. Analiza declaraciones políticas y contenidos relevantes de actualidad para comprobar si se ajustan a la realidad y proporciona una conclusión explicativa del asunto (Figura 2-26).

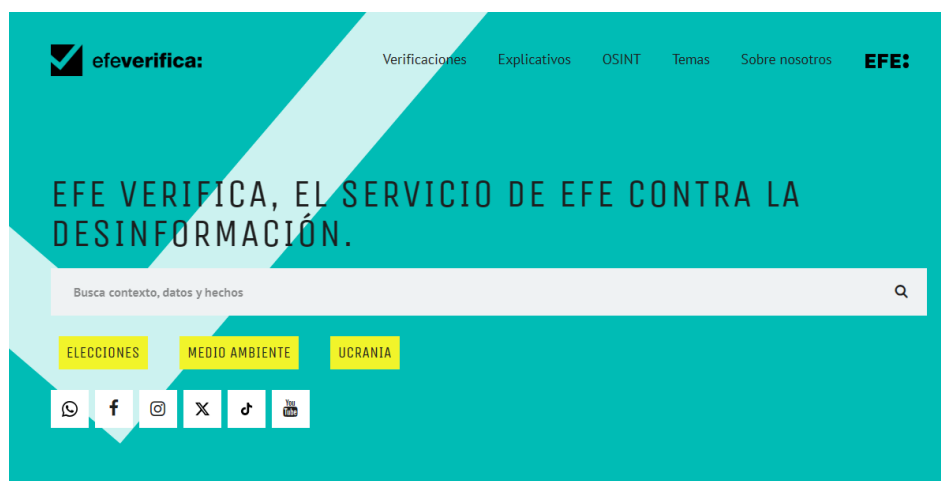


Figura 2-26 EFE Verifica, sitio web de verificación de la Agencia EFE. Fuente:[25]

2.5.3 Enfoques de detección automática

Los métodos anteriores podríamos considerarlos como manuales pues precisan de la intervención de las personas, bien para no distribuir algo de dudosa certeza, o bien para mantener un repositorio de noticias falsas actualizado por profesionales fiables en sitios de confianza. Pero esto no es suficiente dada la velocidad y cantidad con la que se generan las noticias falsas. Es por ello que se precisan medios automáticos para afrontar este problema.

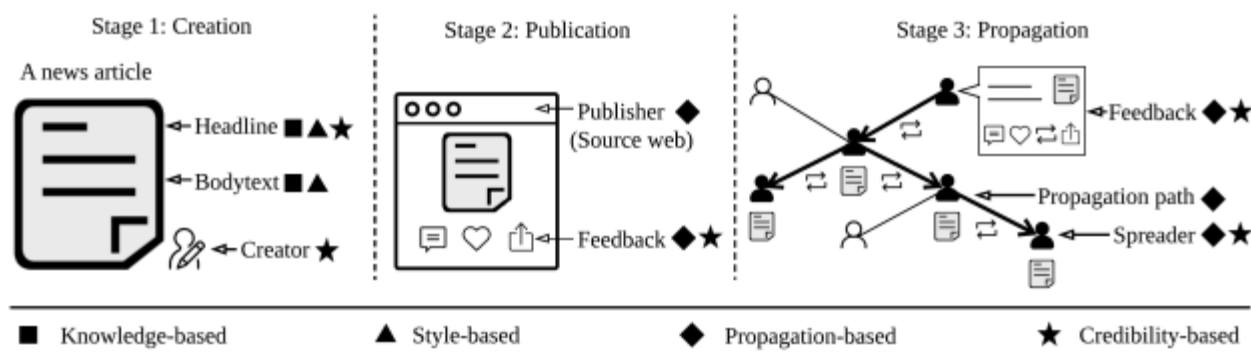


Figura 2-27 Perspectivas de la detección de fake news en su ciclo de vida. Fuente:[26]:

El análisis y detección de noticias falsas de forma automática puede precisar de distintas disciplinas del conocimiento humano, como la lingüística, la psicología, la filosofía, las ciencias sociales o la economía que, al poder interactuar con grandes cantidades de datos y potentes modelos de lenguaje, puede proporcionar nuevas oportunidades para luchar contra la epidemia de desinformación. Hay numerosos enfoques, pero los podemos agrupar en cuatro perspectivas (Figura 2-27) [26]:

- **Basados en el conocimiento.** Estos métodos validan la información contenida en la noticia comparándola con datos ya contrastados, mediante grafos de relaciones entre hechos, fechas, personas y lugares. Es información fiable está disponible para su comparación en conjuntos de datos creados para ese fin, denominados Knowledge Bases (KB).
- **Basados en el estilo.** Similar al anterior, extrae el contenido de la noticia, pero no para verificar los datos, si no para ver la intención con que está escrita: si busca engañar, confundir o provocar sentimientos fuertes que nublen la razón del lector. Se buscan frases poco formales, falta de referencias a fuentes o palabras ambiguas, por ejemplo. Una característica suele ser la polarización, muy negativa o muy positiva. También comprueban la coherencia del discurso de la noticia, extrayendo la información de distintas partes y las contrasta entre sí. Busca inconsistencias en el mensaje transmitido. Un caso muy frecuente es la comparación entre título y cuerpo de la noticia, las fake news suelen utilizar titulares que no están respaldados en el contenido del artículo, y así consiguen que los usuarios que sólo se leen el titular, la reenvíen sin saber en realidad lo que dice la noticia.
- **Basados en la propagación.** En este caso se pone la atención en cómo se difunde la noticia, si hay una cascada de noticias cayendo constantemente o se sigue algún tipo de estrategia de propagación. Normalmente estas noticias están diseñadas para expandirse muy rápidamente en las redes sociales por lo que suelen tener unos patrones característicos de difusión.
- **Basados en la credibilidad de las fuentes (creadores y usuarios).** Los actores en el ciclo de vida de las noticias falsas también pueden proporcionar una buena fuente de información para su detección. La credibilidad de los creadores, publicadores, replicadores y propagadores de las noticias son fundamentales para la credibilidad de la noticia. Habitualmente los autores de noticias y los publicadores suelen tener una postura ante la realidad que se extiende en el tiempo. Cambios de postura muy radicales también son un ejemplo de fuente poco creíble.

A su vez, estas cuatro perspectivas las podemos agrupar en dos grupos de enfoques de la detección, dependiendo de dónde se obtiene la información utilizada en el proceso:

- **Contenido:** Cuando los datos estudiados, se encuentran intrínsecos en la propia noticia, es decir, no precisan más entradas que grandes cantidades de noticias, etiquetadas o no, dependiendo de si utilizamos modelos supervisados o no supervisados. Podríamos agrupar en este bloque las perspectivas que estudian el conocimiento y el estilo. En este conjunto

también podríamos incluir aquellos enfoques que obtienen las características del texto de forma automática directamente mediante *embeddings* como Word2vec, Glove o BERT.

- **Contexto:** En este bloque están las que necesitan información externa a la noticia en sí misma, podríamos decir que son las que se centran en la propagación en las redes y la credibilidad de las fuentes. Este conjunto necesita de conjuntos de datos de entrenamiento más complejos que aporten esa información contextual, muchas veces procedentes de las plataformas de transmisión como las redes sociales.

2.6 Revisión sistemática de literatura (SLR)

Según García-Peñalvo, “La revisión sistemática de literatura es un método sistemático para identificar, evaluar e interpretar el trabajo de académicos y profesionales en un campo elegido. Su propósito es identificar lagunas en el conocimiento y necesidades de investigación en un campo concreto. Las revisiones sistemáticas conforman una familia amplia de métodos y aproximaciones, y resultan totalmente necesarias por el volumen tan enorme de producción científica en formato digital al que se tiene potencialmente acceso.”[27].

Una buena investigación científica se sustenta en la información aportada por los resultados de estudios previos relacionados con la cuestión que se aborda. De ahí que una revisión del estado del arte habitualmente suela formar parte de todo trabajo académico y que si ésta es deficiente el resultado del trabajo se pueda ver comprometido.

Estas revisiones conllevan un esfuerzo considerable, por lo que los artículos que realizan una revisión de calidad de la literatura publicada se han hecho muy populares. Un artículo de revisión tiene como objetivo presentar el conocimiento sobre la materia, pero, sobre todo, lo que se desconoce, tomando como punto de partida unas preguntas de investigación iniciales que se deben resolver.

Una revisión literaria, para que pueda ser considerada sistemática, debe cumplir una serie de requisitos:

- Claridad, que implica una estructura que sea fácil de navegar y entender, y una metodología fácil de evaluar.
- Validez interna que proteja al estudio de sesgos a la hora de seleccionar los artículos primarios, basándose en la relevancia y el rigor de éstos.
- Auditable, es decir transparente en la revisión, para garantizar que las conclusiones se basan en los datos primarios y que los argumentos no se han construido intencionadamente para sustentar una hipótesis preconcebida.
- Reproducible, debe poder ser reproducida por otros investigadores, de tal forma que, siguiendo los mismos pasos, lleguen a las mismas conclusiones.

2.6.1 Metodología PRISMA.

El marco PRISMA (Preferred Reporting Items for Systematics Reviews and Meta-Analysis), publicado en 2009 y actualizado en 2020, es uno de los más utilizados hoy en día para realización de artículos de revisión sistemática de literatura científica, especialmente en estudios médicos, y es el que se utilizará en este documento.

Esta metodología fue diseñada para ayudar en la realización de revisiones sistemáticas, transparentes y de valor para los usuarios. Para ello, partiendo de una pregunta claramente formulada y utilizando un método explícito para identificar, seleccionar y evaluar informes relevantes, se deberá recolectar y analizar datos relevantes de los estudios incluidos en la revisión, para llegar al final, a presentar unas conclusiones sobre los resultados que respondan a las preguntas iniciales.

La declaración PRISMA no pretende ser una guía sistemática que conduce la revisión, sino más bien un elemento de control, que a la hora de realizar el planeamiento y la conducción de la

investigación nos permita asegurarnos de que toda la información necesaria ha sido capturada. La declaración se estructura en 7 secciones, con 27 elementos de verificación, con recomendaciones para cada uno y una plantilla de diagramas de flujo para explicar la selección de artículos primarios realizada. Algunos de los elementos, a su vez, tienen subapartados que permiten una mejor aproximación a las tareas. Además, incluye una lista abreviada para realizar los resúmenes de los artículos y conferencias [28]. En la Tabla 2-3 se recogen los elementos de la metodología PRISMA, agrupados en sus siete secciones.

Para una explicación más detallada de los ítems y su mejor aplicación en la revisión se consultó el artículo “PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews” de Page [29].

Sección	Tema	Núm.
Título	Título	1
Resumen	Resumen	2
Introducción	Razón	3
	Objetivos	4
Métodos	Criterios de elegibilidad	5
	Fuentes de información	6
	Estrategia de búsqueda	7
	Proceso de selección	8
	Proceso de recopilación de datos	9
	Ítems de datos	10 a-b
	Evaluación sesgo del estudio	11
	Medidas de efecto	12
	Métodos de síntesis	13 a-f
	Evaluación sesgo de publicación	14
	Evaluación de la certeza	15
Resultados	Selección de estudios	16 a-b
	Características del estudio	17
	Riesgo de sesgo en los estudios	18
	Resultados de estudios individuales	19
	Resultados de síntesis	20 a-d
	Sesgos del reporte	21
	Certeza de la evidencia	22
Discusión	Discusión	23 a-d
Otra Información	Registro y protocolo	24 a-c
	Soporte	25
	Conflictos de interés	26
	Disponibilidad de datos, código y otros materiales	27

Tabla 2-3 Secciones y Temas de la metodología PRISMA. Fuente:[28]

2.6.2 Herramienta Parsifal

Hacer una revisión sistemática es una tarea que requiere bastante trabajo por parte del investigador, diseñando el protocolo a seguir, ajustando la cadena de búsqueda, clasificando los documentos relevantes, eliminando los que atiendan a los criterios de exclusión y todo ello sin perder la referencia inicial de la investigación. Parsifal es una herramienta web colaborativa, basada en la metodología PRISMA y diseñada para ayudar a los investigadores a realizar esta tarea a la vez que permite compartir su trabajo con otros colaboradores [30].

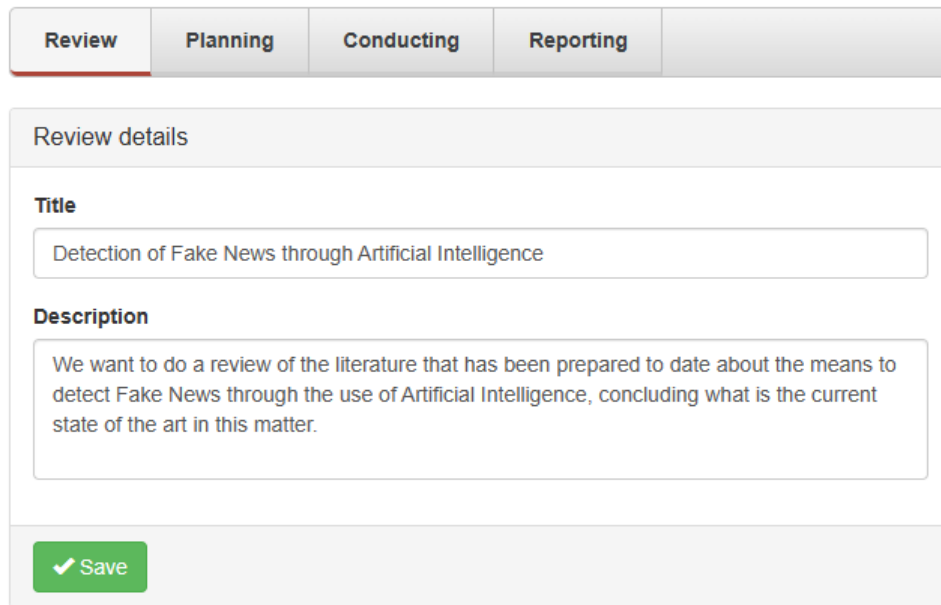


Figura 2-28 Pantalla descriptiva de la revisión en Parsifal. Fuente: Autor

En la Figura 2-28 podemos ver una pantalla de ejemplo de la aplicación Parsifal, concretamente en la que se nombra y describe el trabajo a realizar.

La aplicación divide el proceso en tres fases: planificación, realización e información, accesibles en todo momento mediante unos selectores, a modo de pestaña, en la parte superior de la ventana:

- **Planificación.** Este primer bloque de preparación de la tarea sirve para fijar de antemano los criterios que aplicaremos después, en la conducción de la revisión de forma sistemática. Se divide a su vez en otros.
 - **Protocolo:** En esta zona nos ayudará a decidir lo que buscamos. Permitirá indicar los objetivos, los términos PICOC, las preguntas de investigación, las palabras clave, la cadena de búsqueda general, las fuentes de búsqueda seleccionadas y los criterios de inclusión y exclusión.
 - **Lista de Evaluación de la Calidad.** En esta parte prepararemos las preguntas y respuestas, así como su valoración, que nos permitirán verificar los documentos seleccionados y ordenarlos por relevancia.
 - **Formulario de extracción de datos.** Desde aquí se podrán definir los campos que tendrá el formulario que utilizaremos para extraer los datos de los artículos primarios que formen parte de la revisión.
- **Conducción.** En esta etapa podremos importar los resultados de las consultas realizadas en las fuentes de información y extraer las propiedades de los artículos que sean definitorias para nuestro enfoque. Como la anterior, también se subdivide.
 - **Búsqueda.** Desde aquí se puede realizar una búsqueda directamente en algunas bases de datos documentales, como Scopus.

- **Importación de Estudios.** En caso de no poder consultar directamente, se podrán importar los artículos de las fuentes elegidas si estas utilizan un formato estándar para estos metadatos, como BibTeX.
- **Selección de estudios.** Se unifican los estudios de todas las fuentes y permite detectar duplicados, aceptarlos o rechazarlos, realizando una primera lectura rápida del asunto directamente desde la aplicación, sin tener que recuperar el documento.
- **Evaluación de la calidad.** Una vez seleccionados y utilizando las preguntas y respuestas de evaluación de la calidad de la fase de planificación, podremos hacer una evaluación y ordenación por relevancia de los artículos elegidos.
- **Extracción de datos.** Para los artículos que hayan superado la nota de corte del paso anterior, se rellenará el formulario de extracción de información que definimos en planificación. Con todos los datos rellenos se podrán exportar a una hoja de cálculo para su estudio.
- **Análisis de datos.** En este bloque aparecen unos gráficos sobre los artículos recopilados en la revisión: según fuente, aceptados o rechazados y artículos finales por año de publicación, como este diagrama de sectores de la Figura 2-29.
- **Información.** En esta última parte podremos obtener un documento con toda la información del proceso realizado que servirá de base para realizar el artículo de revisión sistemática literaria.

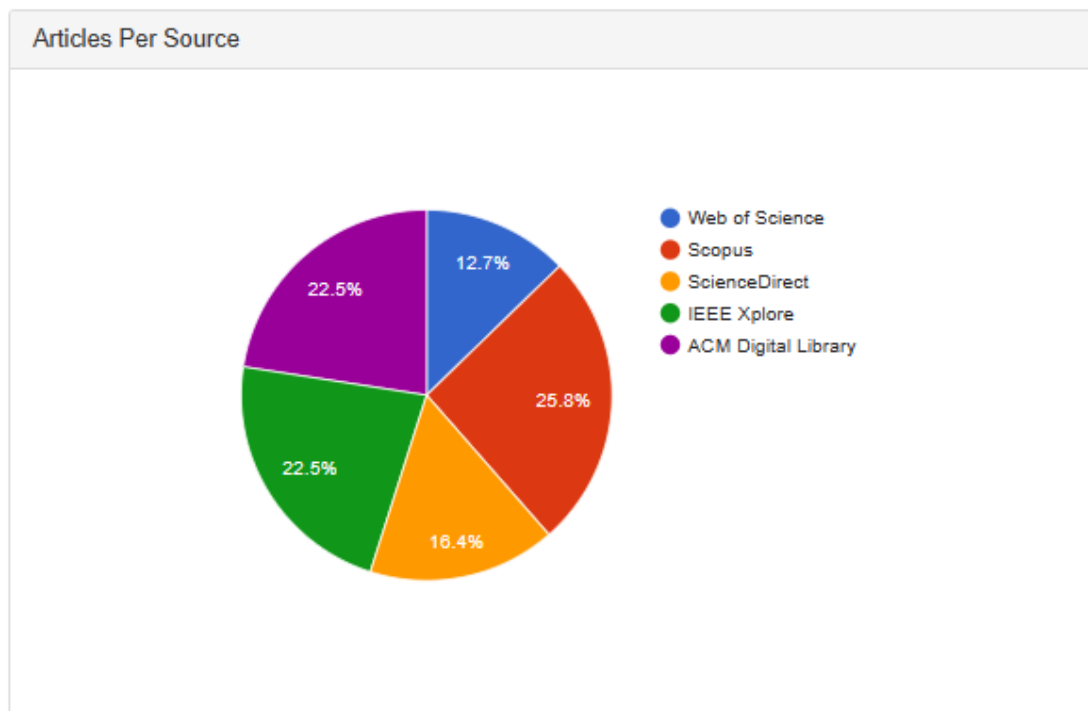


Figura 2-29 Gráfico de Parsifal con número de artículos encontrados por fuente. Fuente: Autor

3 MÉTODO DE REVISIÓN

Para resolver las cuestiones planteadas en los objetivos de este trabajo, en el punto 1.2 de este documento, se ha realizado una revisión sistemática de la literatura científica publicada. En su desarrollo se han seguido las directrices de la metodología PRISMA y se ha utilizado como herramienta de apoyo la aplicación Parsifal, descritas éstas en los apartados 2.6.1 y 2.6.2 de este documento, respectivamente.

3.1 Criterios de elegibilidad

En la planificación del modelo se fijaron los criterios de inclusión y exclusión que permitirían detectar aquellos artículos que se consideraran válidos para la revisión.

3.1.1 Criterios de Inclusión:

- CI1: Trata sobre detección de fake news en español o inglés
- CI2: Utiliza técnicas de ML y NLP
- CI3: Es un estudio primario realizado desde una perspectiva técnica
- CI4: Se ha realizado en los últimos cinco años: 2019-2023 (última búsqueda 30/11/2023).

3.1.2 Criterios de Exclusión:

- CE1: Documento no accesible
- CE2: Es una revisión o encuesta de estudios, no es un estudio primario
- CE3: Se basa en detección manual por agencias de verificación o usuarios
- CE4: Trata noticias en idiomas distintos al inglés o español, como árabe, turco, etc.
- CE5: Está dirigido a un tema o evento específico, como finanzas, política, salud, Covid o procesos electorales
- CE6: Aborda noticias multimodales basadas en videos, imágenes o *deepfakes*
- CE7: No aporta claramente los datos necesarios para la revisión, como técnicas utilizadas, características estudiadas o *datasets* utilizados
- CE8: No proporciona resultados de ejecución obtenidos o métricas de rendimiento
- CE9: Es un estudio anterior al 2019

3.2 Fuentes de información

Para la búsqueda de documentos se seleccionaron cinco de las bases de datos y librerías digitales más utilizadas en la comunidad científica: ACM Digital Library, IEEE Xplore, Science Direct, Scopus

y Web of Science. En la Figura 3-1 tenemos la pantalla de resultados de una de ellas, concretamente ACM Digital Library. La última búsqueda de información sobre estos medios se realizó el 30/11/2023.

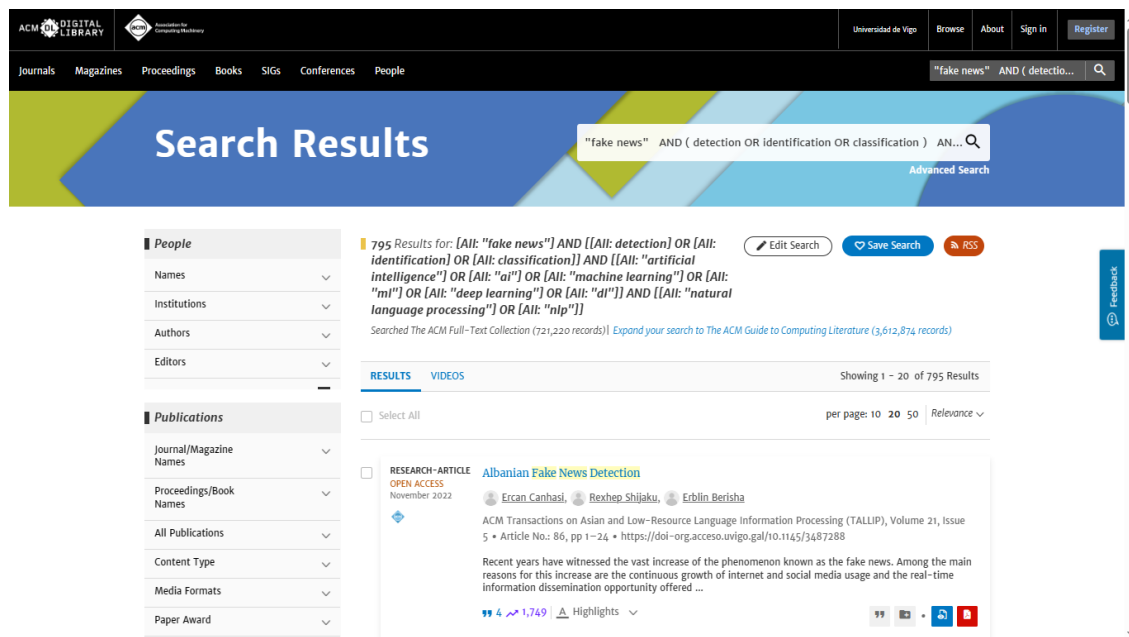


Figura 3-1 Pantalla de resultados de ACM. Fuente: Autor

3.3 Estrategia de búsqueda

Para cumplir el objetivo de la revisión, se llegó a la conclusión que se precisaba recopilar estudios primarios que tratasen sobre:

- Detección, mediante Inteligencia Artificial y Procesamiento del Lenguaje Natural, de fake news en español o inglés.

Tras unos primeros sondeos sobre las bases de datos mencionadas, se confeccionó la siguiente cadena de búsqueda, que cubría los requisitos esenciales que debían cumplir los estudios conforme al enunciado, incorporaba sinónimos y abreviaturas. Esta consulta sería utilizada para buscar en todos los metadatos de las bases de datos, con pequeños ajustes requeridos por cada plataforma.

"fake news"

AND (detection OR identification OR classification)

AND ("artificial intelligence" OR "AI" OR "machine learning" OR "ML" OR "deep learning" OR "DL")

AND ("natural language processing" OR "NLP")

Como se pretendía seleccionar artículos técnicos, que aportaran resultados de rendimiento de los experimentos realizados, se amplió la cadena de búsqueda anterior añadiendo los siguientes criterios de selección con términos relacionados con la ejecución práctica:

AND (technique* OR model* OR method* OR algorithm*)

AND (approach* OR study)

AND (accuracy OR performance)

AND (dataset* OR feature*)

3.4 Proceso de selección

Estos artículos precisaban de una selección, que, basándose en los criterios de inclusión y exclusión definidos anteriormente, permitiera escoger a los que fueran válidos.

La primera parte de la selección se realizaría de manera automática en base al título del documento. Para ello se aumentó la cadena de búsqueda utilizada anteriormente y se volvió a ejecutar

(fake* OR rumo* OR hoax*)
AND (detect* OR classifi* OR indentifi*)
AND NOT (review* OR survey* OR SLR)
AND NOT (covid* OR health* OR virus* OR medic*)
AND NOT (politic* OR president* OR elections OR financ*)
AND NOT (image* OR deepfake* OR facial OR cheapfakes OR video* OR audio* OR multimodal)
AND NOT (arabic OR albanian OR persian OR hindi OR chinese OR Indonesia* OR urdu OR asia* OR philippin* OR Slovak OR Bangla OR German OR Bengal* OR Brazil* OR Sinhala OR Pakistan* OR portug* OR khasi OR china OR dravidi* OR tamil OR malay* OR romani* OR turki* OR French OR Vietnam* OR hausa OR Korean OR thai* OR Africa* OR kurdish OR tunisian OR India*)

Ejecutadas todas estas consultas en las bases de datos seleccionadas, de cada una de ellas, se extrajo un fichero de exportación con los metadatos de los artículos en formato BibTeX, que fue importado en la herramienta Parsifal, tal y como se muestra en la Figura 3-2.

Parsifal

Blog About Help

José Manuel Martínez Sánchez

⚙️ 🔄

Milagros / Detección de Fake News mediante Inteligencia Artificial en español o inglés.

Review

Planning

Conducting

Reporting

1. Search

2. Import Studies

3. Study Selection

4. Quality Assessment

5. Data Extraction

6. Data Analysis

Study Selection

All Sources

ACM Digital Library

IEEE Xplore

ScienceDirect

Scopus

Web of Science

🔍 Find Duplicates

📄 Export Articles

Action:

Select...

Go

 0 of 275 selected

Show: @ All

☐ Accepted

☐ Rejected

☐ Unclassified

☐ Duplicated

☐	Bibtex Key ↕	Title	↕ Author	↕ Journal	↕ Year	↕ Added by	↕ Added at	↕ Status
☐		Towards Fake News Identification using Machine Learning	Abdallah, Yara and Salhab, Nazih and Falou, Ammar El	1st International Conference in Advanced Innovation on Smart City, ICAISC 2023 - Proceedings	2023	José Manuel Martínez Sánchez	02 Dec 2023 20:11:01	Rejected
☐		Towards Fake News Identification using Machine Learning	Abdallah, Yara and Salhab, Nazih and Falou, Ammar El	None	2023	José Manuel Martínez Sánchez	02 Dec 2023 20:11:43	Duplicated
☐		Optimizing fake news detection for Arabic context: A multitask learning approach with transformers and an enhanced Nutcracker Optimization Algorithm	Abdelghani Dahou and Ahmed A. Ewees and Fatma A. Hashim and Mohammed A.A. Al-qaness and Dina Ahmed Orabi and Eman M. Soliman and Elsayed M. Tag-eldin and Ahmad O. Aseini and Mohamed (Abd Elaziz)	Knowledge-Based Systems	2023	José Manuel Martínez Sánchez	02 Dec 2023 20:12:07	Rejected
☐		A Closer Look at Fake News Detection: A Deep Learning Perspective	Abdalla, Ayat and Al-Sadi, Aisha and Abdullah, Malik	None	2020	José Manuel Martínez Sánchez	02 Dec 2023 20:11:21	Accepted
☐		Deep fake news detection system based on concatenated and	Ahmed Sedik and Amr A. Aboahany and Karam M. Sallam and Kumudu	Expert Systems with Applications	2022	José Manuel Martínez	02 Dec 2023 20:12:07	Rejected

Figura 3-2 Herramienta Parsifal, selección de artículos. Fuente: Autor

Desde esta herramienta, se procedió a la eliminación de los duplicados automáticamente, en base a título y autor, y a la lectura de los resúmenes de los no duplicados, para seguir descartando a los no procedentes.

Por último, mediante una lectura ligera de la introducción, se seleccionaron los documentos que pasaban a la siguiente fase y serían objeto de una evaluación más profunda.

3.5 Proceso de extracción de datos

En esta fase de la investigación se procedió a la lectura detallada de los textos y extracción de los datos relevantes. Para ello se utilizó como apoyo una herramienta denominada ChatPDF, basada en GPT 3.5 y GPT 4, que permitía estudiar y organizar los documentos estudiados y realizar consultas sobre los mismos basándose en estos modelos de lenguaje preentrenados. Esta herramienta divide el espacio de trabajo en tres zonas: relación de chats creados, documento en formato pdf para consulta directa y chat propiamente dicho, como podemos observar en la Figura 3-3.

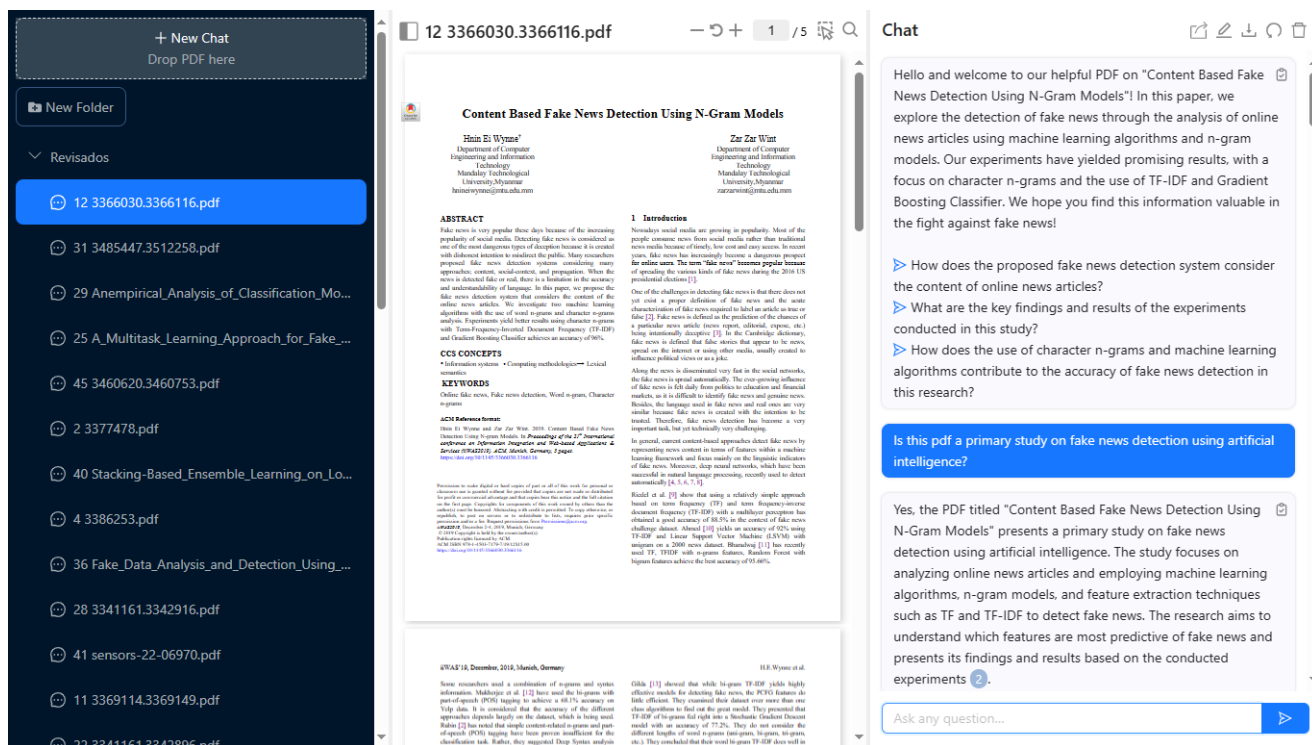


Figura 3-3 Pantalla de herramienta ChatPDF. Fuente: Autor

El procedimiento seguido fue el siguiente. Desde la herramienta Parsifal se exportaron los metadatos a una tabla Excel, en la que cada fila correspondía a un artículo y cada columna a un metadato, y se añadieron las columnas necesarias para recoger los datos adicionales que se habían definido en la planificación para su extracción.

En el siguiente paso se introdujo cada texto en formato pdf en la herramienta ChatPDF y se le fueron realizando, una a una, las preguntas de la siguiente batería de catorce, previamente preparadas (Tabla 3-1) y alguna adicional que se consideró necesaria en casos concretos.

Este proceso sirvió como toma de contacto con el documento y orientación para una mejor comprensión. El resultado de todas las conversaciones se exportó a un fichero de texto por cada documento por si fuera necesaria su consulta posterior. En la Tabla 3-2 se presentan algunos resultados de estas conversaciones a modo de ejemplo.

Número	Pregunta
1	¿Es un estudio primario sobre detección de noticias falsas utilizando inteligencia artificial?
2	¿En qué idioma están las fake news estudiadas?
3	¿Cuál es el objetivo de este artículo?
4	¿Cuántos estudios se han realizado en el documento?
5	¿Qué características son estudiadas?
6	¿Existe algún modelo de aprendizaje automático o aprendizaje profundo aplicado?
7	¿Se aplica alguna técnica de procesamiento de lenguaje natural?
8	¿El autor utiliza algún conjunto de datos para entrenar su modelo?
9	¿Se especifica el tamaño del conjunto de datos utilizado?
10	¿Existe algún tipo de métrica de precisión?
11	¿Cuál es la precisión y el indicador F1 del modelo?
12	¿Qué fortalezas y limitaciones tiene el modelo?
13	¿Cuáles son las conclusiones?
14	¿Puede resumir este documento en 50 palabras?

Tabla 3-1 Batería de preguntas realizadas a ChatPdf. Fuente: Autor

Pregunta	Respuesta
1. Is this pdf a primary study on fake news detection using artificial intelligence?	Yes, the PDF titled "A Multitask Learning Approach for Fake News Detection: Novelty, Emotion, and Sentiment Lend a Helping Hand" presents a primary study on fake news detection using artificial intelligence. The authors introduce a novel multi-task learning framework for misinformation detection, exploring the mutual learning of fake news detection with auxiliary tasks such as novelty detection, emotion recognition, and sentiment prediction ² . They also conduct extensive experiments on benchmark datasets to validate the performance of the proposed model, achieving state-of-the-art performance ² . The methodology section provides a detailed description of the proposed work, including the phases of novelty detection, emotion recognition, sentiment prediction, and fake news detection ² . This primary study contributes to the advancement of fake news detection using AI and provides valuable insights into the role of novelty, emotion, and sentiment in this domain.
2. In what language are the fake news studied?	The PDF does not explicitly mention the language of the fake news studied. However, the authors are affiliated with the Indian Institute of Technology Patna and the Institute of Formal and Applied Linguistics at Charles University, Czech Republic, suggesting a potential focus on English-language fake news. Additionally, the use of benchmark datasets and references to prior work on fake news detection using English-language datasets ² imply a focus on English-language content.

3. What is the objective of this article?

The objective of the article titled "A Multitask Learning Approach for Fake News Detection: Novelty, Emotion, and Sentiment Lend a Helping Hand" is to present a novel multi-task learning framework for automatic fake news detection that jointly learns novelty, emotion, and sentiment patterns in the data to boost fake news detection performance (2). The authors aim to explore the role of textual novelty, emotion, and sentiment for automatic fake news detection and to validate the performance of the proposed model through extensive experiments on benchmark datasets (2). The article also aims to contribute to the advancement of fake news detection using AI and to provide valuable insights into the role of novelty, emotion, and sentiment in this domain (2).

4. How many studies have been performed in the document?

The PDF titled "A Multitask Learning Approach for Fake News Detection: Novelty, Emotion, and Sentiment Lend a Helping Hand" presents a single primary study on fake news detection using artificial intelligence. The study proposes a novel multi-task learning framework for misinformation detection and conducts extensive experiments on benchmark datasets to validate the performance of the proposed model (2). While the article references prior work on fake news detection and related tasks, these are not considered separate studies within the document. Therefore, the document presents one primary study on fake news detection using AI.

5. What features are studied?

The primary study presented in the PDF titled "A Multitask Learning Approach for Fake News Detection: Novelty, Emotion, and Sentiment Lend a Helping Hand" explores the role of textual novelty, emotion, and sentiment for automatic fake news detection (2). The proposed multi-task learning framework jointly learns novelty, emotion, and sentiment patterns in the data to boost fake news detection performance (2). The study uses a two-phase approach to extract novelty, emotion, and sentiment labels from the data and develop a multitask model for fake news detection (7). Therefore, the features studied in this document are novelty, emotion, and sentiment in the context of fake news detection.

6. Are there any machine learning or deep learning models applied?

Yes, natural language processing (NLP) techniques are applied in the document for the purpose of fake news detection. The study discusses the use of a pre-trained BERT model for emotion recognition (2), which is a prominent example of the application of NLP techniques in the context of understanding and analyzing textual data. Additionally, the proposed multi-task learning framework for fake news detection involves the extraction of novelty, emotion, and sentiment labels from the data, which likely involves the application of various NLP techniques for text processing, feature extraction, and sentiment analysis. Therefore, the document demonstrates the application of natural language processing techniques in the context of fake news detection.

Can you summarize this document in 50 words?

The study proposes a novel multi-task learning framework for detecting fake news by jointly learning novelty, emotion, and sentiment patterns in the data. The proposed approach achieves state-of-the-art performance on three benchmark datasets, demonstrating the effectiveness of the proposed model in improving fake news detection performance.

Tabla 3-2 Preguntas y respuestas de ejemplo en ChatPDF. Fuente: Autor

A continuación, se realizó la lectura detallada del documento con una triple función: verificar el cumplimiento de los criterios de selección, evaluar su calidad y extraer la información necesaria para incluirla en el documento Excel de recolección de datos (Figura 3-4).

Fuente	Autor	Título	Año	País	Tipo	Publicador	Características	Metodos	NLP	Dataset	Precisión	Fortalezas	Limitaciones	Comentarios
ACM Digital Library	Shu, Kai	Beyond News Contents: The Role of Social Context for Fake News Detection	2019	EEUU	Conference	ACM	Contenido y Contexto Triple relación publicador, noticias y usuarios	Modelos: ML y DL Factorización no negativa de matrices (NMF)	Teoría de la estructura retórica (RST): árbol representa estructuras retóricas Investigación lingüística y el recuento de palabras (LJWC): Extrae lexicos con categorías psicolingüísticas	FakeNewsNet obtenido desde BuzzfeedNews (81 true, 51 false) PolitiFact (120 true, 120 false)	Accuracy = 0.878, Precision = 0.867, Recall = 0.893, F1 = 0.880	Alta precisión y F1: gran eficacia. Detección temprana gracias al contexto social	Depende de los datos de participación del usuario, no siempre accesibles. El sesgo de los editores no siempre está claro. Necesita de un contexto social claro para dar buenos resultados	Conten participa redes y noticia
ACM Digital Library	Wynne,	Content Based Fake News Detection Using N-Gram Models	2020	Myanmar	Conference	ACM	Contenido, división en n-gramas de palabras y caracteres	RF, GB	Preprocesamiento y tokenización, TF, TF-IDF, n-grams de palabras y de caracteres	Kaggle1 (3128 true, 3128 false) Kaggle2 (1872 true, 2137 false)	Accuracy = 0.96	Alcanza una alta precisión basándose en los tamaños de los n-gramas y frecuencias	Se podrían usar otros factores basados en contexto para mejorar el rendimiento	Los cor no son
Web of Science	Ali, Abdul	Deep Ensemble Fake News Detection Model Using Sequential Deep Learning Technique	2022	Arabia	Journal	MDPI	Contenido Características extraídas del texto mediante TF-IDF	Clasificadores Redes secuenciales profundas SDL para extraer características ocultas MLP clasificador última capa	Preprocesam. Tokeniz. Lematiz. Stopwords BoW, TF-IDF, n-gram Selección de características más representativas mediante SDL	LIAR (12836) ISOT (T=21417, F=23481)	A=0.75, F1=0.85 A=0.999, F1=0.999	Alta precisión Utiliza datasets muy conocidos LIAR y ISOT		Se com extracc conjunt una cla
Web of Science	Prachi,	Detection of Fake News Using Machine Learning and Natural Language	2022	Bangladesh	Journal	JAIT	Extraídas automáticamente mediante TF-IDF	ML: LR, NB, DT, SVM DL: LSTM y GRCU	Preprocesamiento regreg stopwords,	Kaggle (26000)	LR: A=0.74, F1=0.73 NB: A=0.74, F1=0.73	Alta precisión utilizando NLP y DL		Compar DL solo

Figura 3-4 Hoja de cálculo para extracción de datos. Fuente: Autor

3.6 Lista de datos a extraer

Basándose en las preguntas de investigación a resolver, se identificó el siguiente conjunto de datos considerados interesantes para el estudio:

- Autores
- Título
- Año de Publicación
- Fuente de búsqueda y obtención
- País
- Tipo: journal o conferencia
- Publicador
- Características utilizadas:
 - Contenido: Intrínsecas a la noticia (conocimiento, sintácticas y semánticas)
 - Contexto: Externas a la noticia (propagación, usuarios y fuentes)
- Modelos de ML o DL aplicados
- Métodos de NLP utilizados
- Conjuntos de datos de entrenamiento: nombre, propósito, tamaño, fake/real news
- Resultados obtenidos: (Accuracy, F1)
- Fortalezas y Limitaciones
- Comentarios
- Resumen automático
- Resumen

3.7 Evaluación de calidad

Para evaluar la calidad de los estudios aceptados, se efectuaron las siguientes preguntas, con tres posibles respuestas y puntuaciones: Sí = 1, Parcialmente = 0,5 y No = 0. Así mismo, se indicó una puntuación mínima por artículo de 2 puntos, por debajo de la cual se consideraría que no tenía la calidad mínima para entrar en la investigación.

- QA1: ¿Está claro el objetivo del estudio y es acorde con lo que busca esta revisión?
- QA2: ¿Están claramente definidos los métodos y técnicas utilizados?
- QA3: ¿Está adecuadamente descrito el conjunto de datos utilizado?
- QA4: ¿Proporciona discusión sobre los resultados con métricas de rendimiento?

Las preguntas de evaluación de calidad y sus pesos se definieron en la herramienta Parsifal en la zona de planificación para utilizarlo después en la conducción de la revisión (Figura 3-5).

Parsifal Blog About Help José Manuel Martínez Sánchez

Milagros / Detección de Fake News mediante Inteligencia Artificial en español o inglés.

Review **Planning** Conducting Reporting

Protocol Quality Assessment Checklist Data Extraction Form

Quality Assessment Checklist

Questions

Answers

Score

Questions

- ¿Está claro el objetivo del estudio y es acorde con lo que busca esta revisión? [edit] [remove]
- ¿Proporciona discusión sobre los resultados aportando métricas de rendimiento? [edit] [remove]
- ¿Están claramente definidos los métodos y técnicas utilizados? [edit] [remove]
- ¿Está adecuadamente descrito el conjunto de datos utilizado? [edit] [remove]

+ Add Question

Answers

Description	Weight	
Sí	1.0	[edit] [remove]
Parcialmente	0.5	[edit] [remove]
No	0.0	[edit] [remove]

+ Add Answer

Figura 3-5 Preguntas de evaluación de calidad en Parsifal. Fuente: Autor

4 RESULTADOS

En esta sección del documento se realizará, en primer lugar, una exposición del proceso de selección seguido para identificar los estudios más representativos para este trabajo. A continuación, se efectuará una presentación de estos artículos, con expresión del tipo de característica en la que se han basado. Por último, se sintetizarán éstos mediante gráficos y tablas.

4.1 Selección de estudios

Usando las herramientas y métodos presentados en la sección anterior, se procedió a abordar la mencionada metodología PRISMA. En virtud de ésta, se fueron cubriendo las fases establecidas. Un resumen de ello puede verse en la Figura 4-1.

Efectuada la consulta sobre las fuentes seleccionadas, con los criterios de búsqueda descritos en la planificación, fueron identificados 692 estudios.

A continuación, se inició la fase de cribado. Se añadió la consulta por título, pasando a tener 286, y por último, tras filtrar los últimos cinco años se alcanzó la cantidad de 275 estudios. Estos artículos se cargaron en Parsifal y se detectaron los duplicados, quedando 209 textos pendientes de clasificar. Tras la lectura de los resúmenes se seleccionaron 162 publicaciones, y finalmente, después de una lectura ligera de la introducción se eligieron 68 estudios que pasaron a la última etapa.

En la fase de idoneidad se eliminaron 35 y al final quedaron 33 seleccionados. Las principales causas de descarte para ello fueron:

- No eran estudios primarios: En un principio, algunos artículos parecían ser estudios con propuestas novedosas, pero al final resultaban ser una revisión o una comparativa de estudios realizados por otros, no realizados directamente por el autor.
- Orientados a un idioma distinto del inglés o español: Aunque inicialmente no se mencionaba, a medida que se leía el documento se observaba que el objetivo estaba restringido a una lengua distinta de español o inglés, o que el conjunto de datos de entrenamiento estaba en un idioma distinto de español o inglés.
- Multimodal: Contemplaban la influencia de elementos multimedia en los algoritmos de detección.

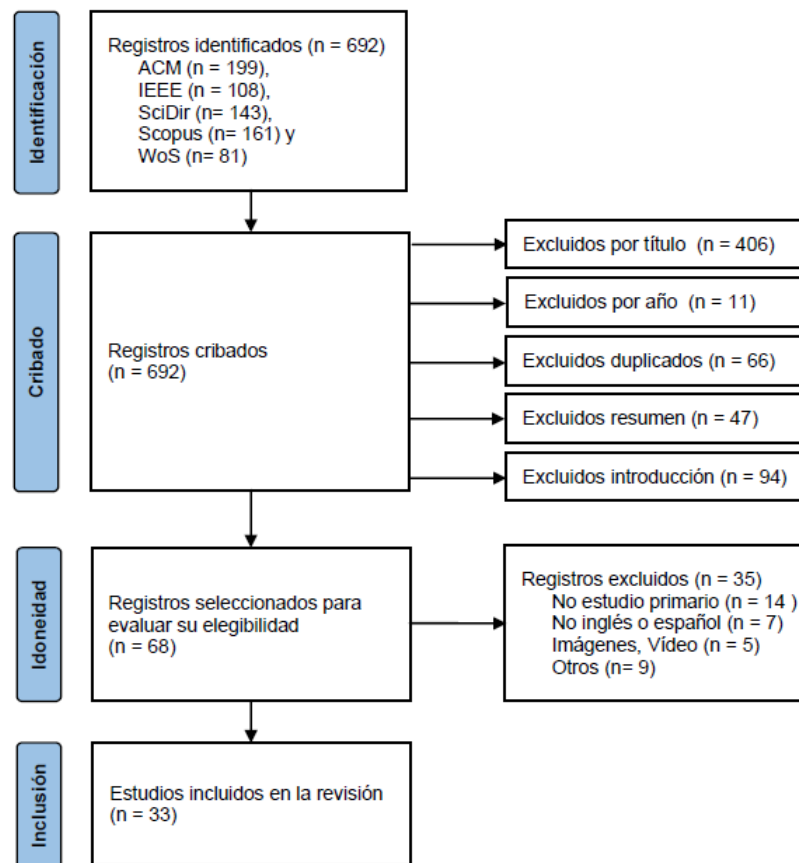


Figura 4-1 Diagrama de selección de estudios según PRISMA (elaboración propia).

En cuanto a la evaluación de la calidad sólo hubo dos artículos que no definían claramente los métodos y técnicas utilizados, y siete que usaban un *dataset* creado por el autor o poco conocido y no hacían una definición extensa del mismo que permitiera reproducir el experimento. No obstante, todos superaron la puntuación mínima requerida para formar parte del estudio.

4.2 Descripción de los estudios

A continuación se presentan los 33 estudios seleccionados en la Tabla 4-1 con su referencia, país de desarrollo, año de publicación, autor, tipo de características estudiadas y un breve resumen de su contenido. Estos estudios son procedentes en un 58% de actas de conferencias de congresos científicos y en el 42% restante de artículos de revistas académicas.

Ref.	País / Año	Autor	Caracter.	Resumen
[31]	Jordania 2020	Abedalla et al.	Contenido	Enfoque de concordancia entre título y cuerpo, tanto en estilo como en semántica.
[32]	Arabia 2022	Ali et al.	Contenido	Extracción de TF-IDF, selección de características con clasificadores de redes secuenciales profundas y clasificación final con MLP
[33]	Arabia 2022	Aljabri et al.	Contenido	Comparativa de dos clasificadores RF y SVM con varias técnicas de extracción, con el mismo <i>dataset</i> .
[34]	Arabia 2022	Althubiti et al.	Contenido	Método: STODL-FNDC Sobre un <i>embedding</i> de Glove, modelo basado en Deep Belief Network con un algoritmo de mejora de los hiperparámetros del modelo "Sea Turtle Foraging Optimization".

[35]	Líbano 2020	Antoun et al.	Contenido	Su intención es utilizarlo en tiempo real. Comprensión del lenguaje (estilo y semántica), detectando subjetividad, presuposición o sesgo gracias a modelos preentrenados como BERT, XLNET o RoBERTa
[36]	España 2021	Bonet-Jover et al.	Contenido	Orientado a noticias periodísticas, busca que éstas tengan estructura adecuada y su contenido responda a las 5W1H preguntas básicas de una noticia
[37]	Austria 2021	Brasoveanu et al.	Contenido	Integración de características semánticas y sintácticas, comparando varios clasificadores. También presenta un método para construir grafos de conocimiento.
[38]	EEUU 2021	Dou et al.	Contenido Contexto	Framework: UPFD Se basa en el contenido de las publicaciones históricas de los usuarios para detectar sus preferencias, su perfil en las redes y las interacciones con las noticias.
[39]	EEUU 2020	Islam et al.	Contenido Contexto	Modelo RumorSleuth En el mismo proceso detecta la veracidad del rumor y la postura del usuario. Se basa en el contenido de la noticia, los comentarios y el perfil del usuario en Twitter.
[40]	Pakistán 2021	Islam et al.	Contenido Contexto	Modelo Ternion. Combina la postura del usuario según el contenido de la noticia y la credibilidad del autor mediante una API de Twitter
[41]	India 2019	Jadhav et al.	Contenido	Utiliza un modelo semántico estructurado basado en redes neuronales recurrentes. Estudia palabras claves y conceptos.
[42]	Grecia 2021	Kasseropoulos et al.	Contenido	Comprueba 84 características léxicas y 300 procedentes de <i>embedding</i> . Se basa en que las fake news son más cortas, menos palabras técnicas y más cortas, más redundancia, etc.
[43]	Bangladesh 2022	Khan et al.	Contenido	Comparativa de varios métodos de clasificación de ML y TF-IDF con BERT sobre un texto al que se ha hecho stemming.
[44]	India 2021	Kumari et al.	Contenido	Detecta novedades, reconoce emociones y predice sentimientos polarizados, que son claros indicadores de una fake news. Utiliza <i>embeddings</i> sobre modelos preentrenados.
[45]	India 2022	Kumari et al.	Contenido	Detecta, mediante modelos preentrenados, novedad (GloVe) y emoción (BERT) en el contenido, como indicadores de noticias falsas, porque aumenta la difusión en las redes.
[46]	China 2019	Li et al.	Contenido	Extracción de características mediante TF-IDF y LSA, aprendizaje mediante <i>stacking</i> de cinco modelos y fusión del resultado.
[47]	EEUU 2020	Liu et al.	Contenido Contexto	FNED: Modelo para rápida detección en redes sociales utilizando las respuestas de los usuarios a una noticia y sus perfiles.

[48]	EEUU 2022	Mosallanezhad et al.	Contenido Contexto	Modelo REAL-FND Captura en BERT y divide las características en dos: invariables y dependientes del dominio. Mediante aprendizaje por refuerzo puede adaptar a otro dominio.
[49]	China 2023	Nadeem et al.	Contenido Contexto	Modelo: HyproBert CNN, para extracción de datos espaciales BiGRU para extracción de datos contextuales CapsNet con autoatención para comprensión jerárquica. de relaciones Capa densa para clasificación
[50]	India 2021	Nagaraja et al.	Contenido	Comparativa de dos clasificadores, NB y SVM, sobre un mismo <i>dataset</i> vectorizado.
[51]	EEUU 2022	Nguyen et al.	Contexto	Framework: FANG Utiliza la representación gráfica del contexto social de la difusión de noticias (interacción de usuarios, noticias y medios de publicación) mediante GraphSage.
[52]	Bangladesh 2022	Prachi et al.	Contenido	Comparación de seis modelos de ML y DL sobre un mismo <i>dataset</i> vectorizado con TF-IDF
[53]	India 2021	Ramya et al.	Contenido	Comparativa de ejecución de cuatro modelos de redes neuronales basadas en mecanismos de atención sobre características léxicas y de sentimiento, y lingüísticas.
[54]	India 2019	Reddy et al.	Contenido	Sistema compuesto por cinco clasificadores de ML sobre el mismo conjunto de datos. El resultado final es por mayoría.
[55]	EEUU 2019	Sadiq et al.	Contenido	Combina dos <i>embeddings</i> : uno con <i>autoencoders</i> para características complejas (sarcasmo, sátira, etc.), otro de Google News y otro con características sintéticas.
[56]	China 2021	Sheng et al.	Contenido	Framework: Pref-FEND Se basa en la integración de dos modelos. Uno basado en patrones de estilo y otro en entidades de conocimiento para localizar afirmaciones verificables.
[57]	EEUU 2019	Shu et al.	Contenido Contexto	Framework: TriFN. Basado triple relación entre publicadores, noticias y usuarios. Contempla sesgo de los publicadores y participación de los usuarios
[58]	India 2019	Telang et al.	Contenido	Comparación de distintos clasificadores sobre un mismo corpus procesadas sus frecuencias TF-IDF con GloVe
[59]	EEUU 2020	Veyseh et al.	Contenido Contexto	Detección de relaciones semánticas entre una noticia y sus réplicas. Para ello utiliza modelos preentrenados como GloVe, ELMo, BERT o GPT
[60]	Filipinas 2021	Villagracia et al.	Contenido	Modelo que utiliza algoritmo XGBoost sobre datos de estilo y sentimientos. Añade características, como bigrama de palabra del título o publicaciones de autor.
[61]	Myanmar 2020	Wynne et al.	Contenido	Detección basada en frecuencias de palabras y n-gramas

[62]	China 2022	Xu et al.	Contenido	Captura estructuras semánticas complejas y dependencias a larga distancia entre los datos, detectando relaciones entre afirmaciones y evidencias.
[63]	EEUU 2020	Zhou et al.	Contenido	Modelo de detección temprana, antes de propagación. Se basa en características interpretables de estructura y estilo del texto: léxico, sintáctico semántico y discurso.

Tabla 4-1 Estudios seleccionados con autor, características y resumen. Fuente: Autor

4.3 Síntesis de resultados

En base a los datos extraídos de los artículos seleccionados para el estudio, se presentan los resultados obtenidos, resumidos en forma de gráficos y tablas.

4.3.1 Características estudiadas

Mostramos las características según los enfoques para la detección automática presentados en el punto 2.5.3 de este trabajo y en la agrupación que se hizo de ellos en dos categorías:

- Contenido, cuando se refiere a información intrínseca al texto de la noticia. Lo componen los siguientes enfoques:
 - Conocimiento. Comprobación de la veracidad de los hechos, personas, lugares y fechas que intervienen en la noticia comparándolos con hechos ya contrastados en bases de datos de conocimiento, conocidas como *Knowledge Bases*.
 - Estilo. Estudio del léxico, la sintaxis o el discurso utilizados, buscando rasgos que identifiquen una forma de escribir propia de las fake news.
- Contexto, cuando la información se obtiene del entorno o plataforma que sustenta a la misma. Agrupan los enfoques de:
 - Propagación. Se basa en la forma en que se difunde, tiempos, patrones, etc.
 - Fuentes. Se centra en los publicadores, creadores y difusores de las noticias, su credibilidad y actitud ante ellas.

Nos encontramos con que 24 de los 33 estudios se basan exclusivamente en el contenido de la noticia, 8 utilizan ambos grupos y tan solo uno se basa únicamente en el contexto, como podemos observar en la Figura 4-2. Parece que lo más habitual es utilizar el contenido de las noticias, aunque lo lógico sería pensar que una aproximación híbrida sería más completa.

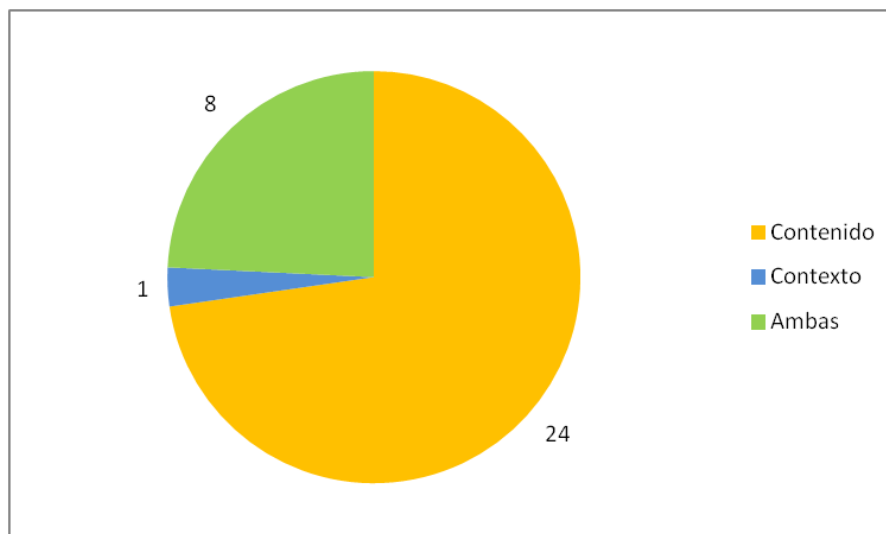


Figura 4-2 Tipos de características estudiadas (elaboración propia).

Si profundizamos en los enfoques utilizados, observamos que, dentro de la categoría del contenido, las características más frecuentes son las basadas en el estilo, concretamente en el léxico, seguidas por el contenido sintáctico, y ya más alejadas las de discurso. Si bien es cierto que muchos estudios tratan las dos primeras juntas, pues no distinguen entre ambos grupos de categorías. En cuanto al conocimiento y la validación de los hechos, se han identificado bastantes menos iniciativas.

También aparece un grupo importante relacionado con el contenido, identificado como automático. En estos estudios, no importa exactamente el significado de las características extraídas, pues se hace automáticamente, basándose para ello, en las frecuencias de los términos y modelos de aprendizaje automático.

Por otro último, los enfoques basados en el contexto tienden a utilizar los tres componentes en igual proporción, de hecho en algunas aproximaciones combinan los tres. (Figura 4-3).

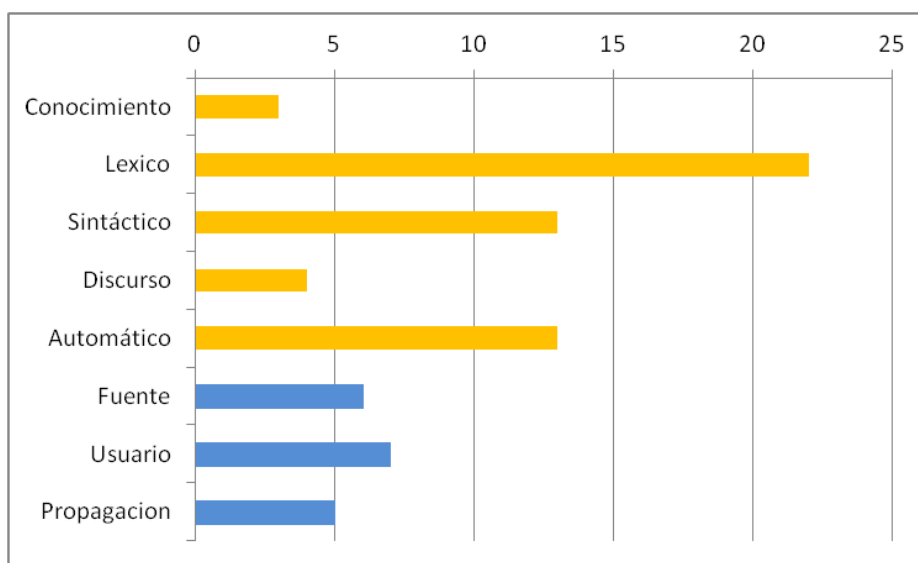


Figura 4-3 Enfoques utilizados en la detección (elaboración propia).

4.3.2 Métodos de extracción de características

La revisión partía de la premisa de que los estudios tenían que servir para detectar fake news de forma automática mediante IA en español o inglés, por lo tanto, una consecuencia implícita era que se tenía que utilizar NLP para entender las claves del idioma en que estaban escritas, y por ello fueron descartados aquellos estudios que no lo utilizaban.

Estudiando esta parte del tratamiento, encontramos que todos los trabajos realizaban un preprocesamiento de los textos de entrenamiento, buscando mejorar su calidad, eliminar errores o elementos discordantes en el corpus. Además, la mayoría de ellos realizaban tareas como tokenización, lematización, stemming o eliminación de *stopwords*, por lo tanto no se han incluido en la gráfica por claridad.

En el Figura 4-4 se puede observar que la técnica de NLP más utilizada es la de TF-IDF, si bien es cierto que en muchos estudios sirve como base para un tratamiento posterior. Esto mismo sucede con otras técnicas sencillas como BoW, TF o N-GRAMS. Otro grupo es el de los *embeddings* Word2Vec, GloVe y BERT que son utilizados como entrada del texto en bastantes estudios.

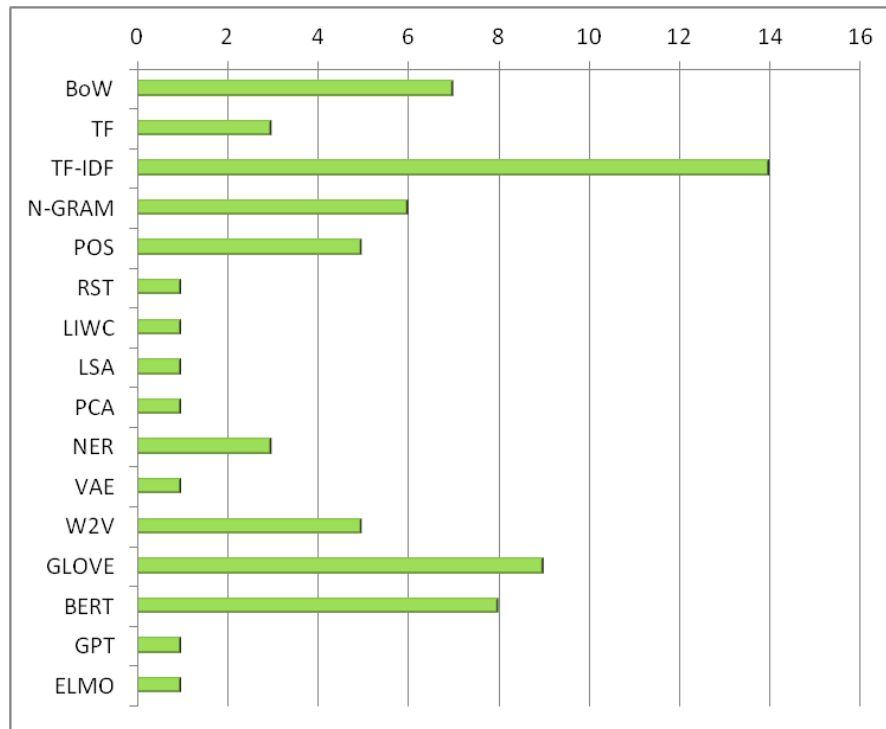


Figura 4-4 Técnicas de NLP utilizadas (elaboración propia).

4.3.3 Modelos de ML y DL utilizados

Todos los trabajos utilizan ML o DL, y algunos los dos, ya que uno de los condicionantes de los criterios de inclusión era que utilizara algún algoritmo de IA para su detección. A la vista de los resultados, parece que hay una mayor tendencia a utilizar DL en la detección de fake news, aunque hay una cantidad relevante de modelos mixtos, como podemos visualizar en la Figura 4-5.

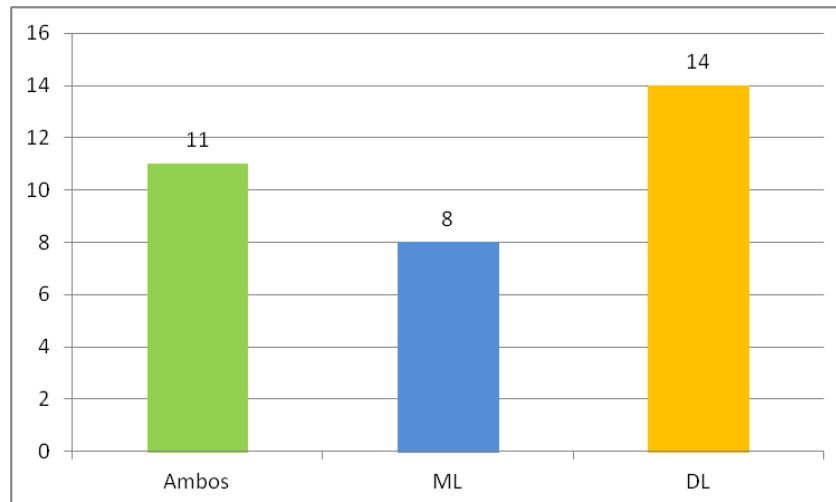


Figura 4-5 Tipos de Aprendizaje automático utilizados (elaboración propia).

Si nos adentramos en las técnicas de ML utilizadas en los trabajos presentados, en la Figura 4-6 vemos que la más utilizada es SVM ya que es un clasificador sencillo de implementar que da muy buenos resultados. La regresión logística es también muy utilizada, al igual que el *random forest*, si bien es cierto que estas técnicas se utilizan muchas veces como proceso final de clasificación de un ensamble de otros modelos, en ocasiones incluso de DL.

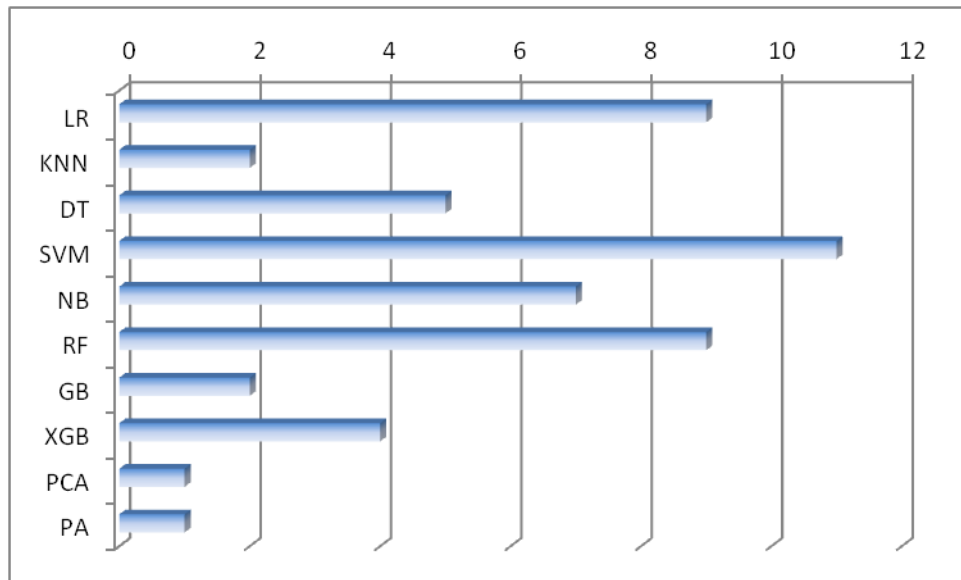


Figura 4-6 Técnicas de ML utilizadas (elaboración propia).

En cuanto a las técnicas de DL, podemos apreciar en la Figura 4-7 que la más utilizada con diferencia es la red neuronal LSTM, en la mayoría de los casos acompañada de mecanismos de atención. Pero si unimos todas las RNN y su variaciones, es decir RNN, LSTM, GRU y HAN la hegemonía de estos tipos es abrumadora. Esto no es muy sorprendente, pues son las más utilizadas en el NLP desde hace algunos años.

En cuanto a los transformers empiezan a hacer su aparición con modelos basados en BERT, pero parece que en estos estudios aún no se está produciendo una avalancha a la misma velocidad que en otros campos, como chats, navegadores o traductores.

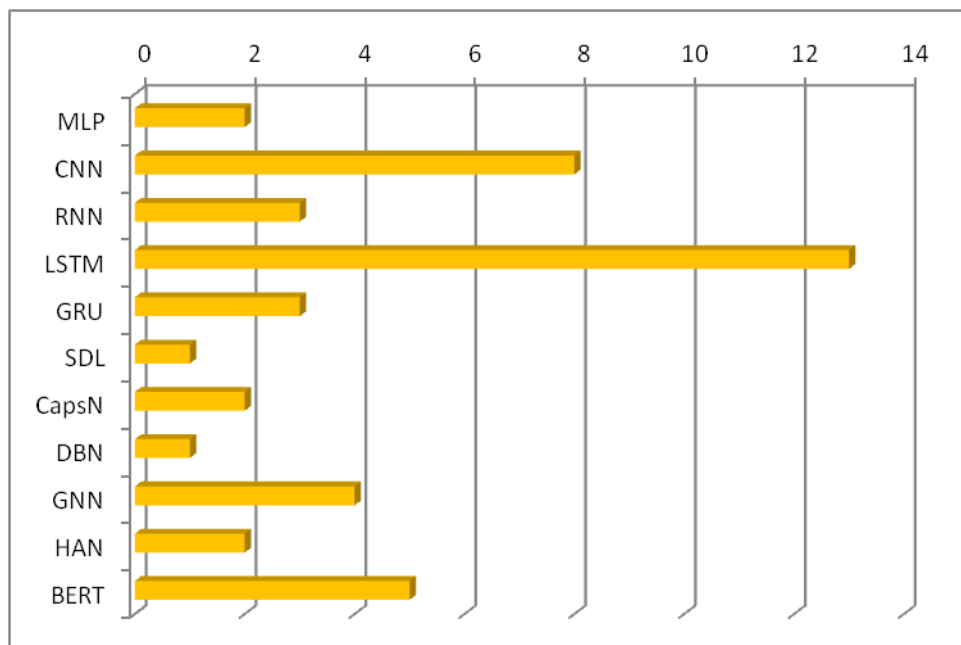


Figura 4-7 Técnicas de DL utilizadas (elaboración propia).

4.3.4 Técnicas de IA utilizadas

En la Tabla 4-2 podemos observar cada estudio con las técnicas de IA utilizadas, de ML y de NLP.

Ref.	ML / DL	NLP
[31]	DL: CNN, LSTM, Bi-LSTM	Preprocesado y tokenización con NLTK. Glove y BERT
[32]	DL: Redes secuenciales profundas SDL para extraer característ. ocultas MLP clasificador última capa	Preprocesamiento, tokenización, lematización, <i>stopwords</i> , BoW, TF-IDF, n-gram SDL: Selección de características representativas
[33]	ML: RF, SVM	Preprocesamiento, <i>stopwords</i> , tokenización, stemming, Hash, TF-IDF, n-grams
[34]	DL: Deep Belief Network DBN STO Optimización hiperparámetros	Preprocesamiento, TF-IDF, <i>embedding</i> de GloVe
[35]	ML: XGBoost DL: XLNet, RoBERTa y BERT	Tokenización palabras, n-grams y char-grams, comparación <i>embedding</i> título y búsquedas en Google (la más importante). Word2vec de Google-news.
[36]	DL: LSTM ML: LR, DT, SVM, NB	Tokenización (Spacy) y división en partes mediante CRF
[37]	ML:SVM DL: LSTM, CNN, BiLSTM + atención, GRU + atención y Capsule Network	Named Entity Recognition Linking and Classification (NERLC), n-grams, detección de temas, Etiquetado gramatical <i>Part of Speech</i> (POS), extracción de relaciones de <i>Knowledge Graphs</i> como DBpedia
[38]	DL: Graph Neural Networks (GNN) y Graph Convolutional Networks (GCN)	Word2vec y BERT para endógenas
[39]	DL: STM texto y postura de usuario Variational autoencoders VAE para perfiles de usuario	RNN para extraer características del texto doc2vec para hacer <i>embeddings</i> de cada tweet
[40]	ML: DT, RF, LR y SVM	preprocesamiento, tokenización, part of speech, Named Entity Recognition NER y TF-IDF
[41]	DL: Deep Structured Semantic Model (DSSM), LSTM	Tokenización, BoW, TF-IDF
[42]	ML: SVM, DT, RF, GB DL: CNN, LSTM	Preprocesamiento de texto, Named Entity Recognition (NER), extracción de características léxicas, word <i>embeddings</i> , PCA.
[43]	ML: LR, NB, SVM y Pasivo-Agresiv. DL: BERT	Preprocesamiento, stemming, TF-IDF y BERT
[44]	DL: Bi-LSTM, MLP	<i>Embedding</i> BERT y Quora para novedad <i>Embedding</i> BERT y Unified para emoción <i>Embedding</i> Glove y GoEmotion para emoción.
[45]	ML: LR DL: CNN, PCA	Natural Language Inference NLI, <i>Embeddig</i> con Glove para novedad, <i>Embedding</i> con BERT para emoción
[46]	ML: RF, XGBoost, <i>Bagging</i> , Lightgbm. <i>Stacking</i> fusionar modelos con validación cruzada y LR	Preprocesamiento: tokenización, lematización, <i>stopwords</i> , BoW, TF-IDF y Latent Semantic Array LSA para reducir y extraer principales características

[47]	DL: Extractor características respuestas usuarios, CNN y clasificador Positive-Unlabeled PU	Tokenización con NLTK
[48]	DL: BERT para tratar la noticia, Hierarchical Attention Network HAN para tratar comentarios de usuarios	Preprocesamiento, eliminación de palabras fuera de vocabulario, truncamiento hasta 512 palabras. BERT para el <i>embedding</i> del texto
[49]	CNN, extracción de datos espaciales BiGRU extracción contextuales CapsNet y autoatención para Jerarquía. de relaciones y Capa densa para clasificación	Preprocesamiento, Tokenización y DistilBERT para <i>word embedding</i>
[50]	ML: NB y SVM	Preprocesamiento, tokenización, lematización, <i>stopwords</i> , BoW. Análisis semántico
[51]	DL: GNN estructura social y patrones de participación. BiLSTM	Noticias: vector TF-IDF + vector semántico a cada palabra con <i>embeddings</i> preentrenados de GloVe Fuentes: Igual, con información del sitio web Usuarios: Igual con información de comentarios
[52]	ML: LR, NB, DT, SVM	Preprocesamiento: regexp, <i>stopwords</i> ,
	DL: LSTM y BERT	tokenización y lematización. BoW y TF-IDF
[53]	DL: CNN, LSTM, BiLSTM y CLSTM HAN (Hierarchical Attention Network)	Preprocesamiento y selección de funciones + <i>word embedding</i> GloVe (100d) preentrenado = <i>embedding</i> para DL
[54]	ML: NB, RF, DT,SVM, K-NN	cBOW, TF-IDF
[55]	ML: RF DL: Bi-LSTM	Variational Autoencoders VAE para <i>embeddings</i> de 300d. Se añade <i>embedding</i> word2vec de 300 d de Google news. Se añaden 300 d sintéticas de ambos. Se crea espacio de 900 dimensiones
[56]	DL: Bi-LSTM, EANN-Text, BERT-Emo para patrones DeClarE, EVIN, MAC para hechos GNN para mapas de preferencias,	Tokenización, <i>encoding</i> , y extracción de elementos de estilo y afirmaciones del texto para los mapas de preferencias
[57]	ML y DL Factorización no negativa de matrices (NMF)	Teoría de la estructura retórica (RST): árbol representa estructuras retóricas Investigación lingüística y recuento de palabras (LIWC): Extrae léxicos categorías psicolingüísticas
[58]	ML: LR, RF DL: RNN, LSTM, GRU	Preprocesamiento, tokenización, TF-IDF, GloVe,
[59]	DL: RvNN, Autoatención	Glove, ELMo, BERT y GPT
[60]	ML: XGBoost	n-grams, TF-IDF, extracción de características
[61]	ML: RF, GB	Preprocesamiento y tokenización, TF, TF-IDF, n-grams de palabras y de caracteres
[62]	DL: Semánticas con GNN (GET). Redes neuronales gráficas (GNN)	Preprocesamiento, tokenización
[63]	ML: XGBoost.	BoW, W2V, S2V

Tabla 4-2 Estudios seleccionados con métodos utilizados. Fuente: Autor

4.3.5 Modelos de Aprendizaje y grado de precisión

En la Tabla 4-3, que se presenta a continuación, tenemos, por cada uno de los estudios seleccionados, los métodos y conjuntos de datos utilizados, el tamaño de éstos y los resultados obtenidos en términos de *accuracy* y F1-score.

En general, se observan unos valores bastante altos en los resultados. En muchos casos, esto se puede deber a que se hayan utilizado como datos de prueba para validar los modelos, particiones aleatorias del mismo conjunto de datos de entrenamiento. El problema es que esos datos tienen un mismo origen, y por lo tanto se parecen bastante, incluso se pueden llegar a repetir, lo que hace que se pueda producir cierto sobreajuste en el modelo, y por lo tanto, que la generalización no sea todo lo buena que se espera cuando se le presenten datos que no tengan ninguna intersección con los datos de entrenamiento, como explica en su estudio, Abedalla et al [31], que dio una precisión o *accuracy* de 0,712 con datos totalmente distintos a los de entrenamiento y lo considera un buen resultado en comparación con el resto de trabajos.

Otras circunstancias que pueden desvirtuar los resultados están relacionadas con la calidad de los conjuntos de datos o *datasets* utilizados, en cuanto a que:

- No son lo suficientemente extensos, porque su creación es laboriosa, sobre todo el etiquetado y clasificación en noticias ciertas o falsas.
- No son lo suficientemente originales, porque se crean desde unos pocos repositorios, como Politifact, Gossip, etc. y tienden a repetir contenidos y técnicas.
- No son totalmente reales ya que muchas veces se crean en fábricas de noticias falsas para este fin y les falta la frescura que tendrían las noticias falsas reales.

Así, podemos ver valores por encima de 0,90 que, lógicamente, si fueran ciertos en las noticias reales, podríamos implementarlo de forma global a nivel de usuarios en los navegadores y redes sociales y las fake news ya no serían un problema.

Destacan especialmente algunos resultados por encima de 0,99 como los de:

- Ali et al [32], que utiliza redes secuenciales profundas, SDL para extraer características ocultas sobre textos vectorizados mediante TF-IDF y un clasificador MLP final.
- Althubiti et al [34], que trata contenido extraído mediante *word embedding* de GloVe y procesado con Deep Belief Network (DBN) con un mecanismo automático de optimización de hiperparámetros STODL.
- Jadhav et al [41], también basado en contenido y DL como los anteriores, utiliza un modelo semántico estructurado profundo (DSSM) con palabras claves y conceptos, en combinación con redes neuronales recurrentes LSTM.
- Khan et al [43], basado también en DL, hace una extracción de características generalizada mediante BERT sobre texto al que se le ha hecho stemming y TF-IDF.
- Kumari et al [44], basado en contenido, utiliza para el aprendizaje una doble LSTM con un Multi-Layer Perceptron (MLP) como clasificador, y para capturar las características de novedad y emoción, *embeddings* de BERT y Glove.
- Nadeem et al [49], este último utiliza un modelo híbrido para extraer la características del contenido y del contexto. Hace un *word embedding* mediante DistlBERT del texto de las noticias y con redes CNN extrae datos espaciales, con redes Bi-GRU extrae datos contextuales. Por último con CapsNet hace una comprensión jerárquica de relaciones y con una capa densa hace la clasificación final.

Un dato a tener en cuenta en los resultados es el número de posibilidades que tenga la clasificación del conjunto de datos utilizados. Una etiqueta binaria, del tipo cierto o falso, es más sencilla de pronosticar y dará mejores resultados que una con seis dimensiones, como en el caso del *dataset* LIAR.

Ref.	Método	Dataset	Accuracy	F1-Score
[31]	CNN, LSTM, Bi-LSTM	Fake News Challenge (75000)	0,712	
[32]	SDL	LIAR (12836)	0,750	0,850
		ISOT (T=21417, F=23481)	0,999	0,999
[33]	SVM	WELFake	0,980	0,980
	RF	(T=35028, F=37106)	0,940	0,940
[34]	DBN + STO	ISOT-1 (T=21417, F=23481)	0,996	0,996
	Optimización hiperparámetros	ISOT-2 (T=1000, F=1000)	0,955	0,955
[35]	XLNet	QICC (T=192, F=192)	0,980	0,980
	BERT		0,920	0,960
[36]	5W1H Pipeline	FNDDeepML.(200 noticias)	0,750	0,740
[37]	CapsNet	LIAR (12836)	0,649	
		Politifact (3651)	0,524	
[38]	GNN + GCN + BERT	FakeNewsNet (T=157, F=157)	0,846	0,845
		Gossipcop (T=2732, F= 2732)	0,972	0,972
[39]	STM+VAE	PHEME (297 hilos de twiter)		0,730
		Twitter 15+16 (1729 hilos)		0,890
[40]	SVM	Kaggle (T=10369, F=10349)	0,931	0,941
[41]	DSSM + LSTM	LIAR	0,990	0,990
[42]	SVM	Sigkdd2200 (T=2972, F=2014)	0,772	
	DT		0,658	
	RF		0,758	
	GB		0,762	
	CNN		0,792	
	LSTM		0,752	
[43]	LR	Kaggle (T=10387, F=10413)	0,979	0,980
	NB		0,950	0,947
	PA		0,990	0,990
	SVM		0,991	0,991
	BERT		0,997	0,997
[44]	Bi-LSTM, MLP	ByteDance (400893)	0,997	0,990
		FNC (49972))	0,969	0,980
		Covid-Stance (14000)	0,986	0,980
[45]	CNN + PCA + LR	ByteDance (320767)	0,961	0,981
		FNC-1 (49972)	0,947	0,970
		Covid-Stance (14000)	0,941	0,950
		LIAR-PLUS (12836)	0,777	0,785
[46]	Stacking ML+LR	FakeNewsChallenge (66034)	0,999	0,878
		FakeNewsNet (997)	0,882	0,752
[47]	CNN + PU	Weibo16 (T=2351, F=2313)	0,938	0,942
		Twiter15 (T=353, T=327)	0,985	0,980
[48]	REAL-FND	FakeNewsNet (T=2645,F=2770)		0,862
	Noticia: BERT	GossipCop(T=3586,F=2230)		0,901
	Comentarios: HAN			
[49]	CNN+BiGRU+CapsNet+	ISOT (44898)	0,993	0,992

	Autoatencion	FA-KES (804)	0,611	0,611
		ISOT-R10	0,957	0,941
[50]	NB	Autor	0,630	0,670
	SVM		0,750	0,750
[51]	GNN + BiLSTM	Dataset (T=606, F=448)	0,751	
[52]	LR	Kaggle (26000)	0,740	0,730
	NB		0,740	0,730
	DT		0,900	0,890
	SVM		0,770	0,760
	LSTM		0,950	0,940
	BERT		0,980	
[53]	CNN	LIAR (12000)	0,730	0,770
	LSTM		0,560	0,720
	CLSTM		0,560	0,720
	BiLSTM		0,670	0,710
[54]	Ensamble ML	Kaggle	0,926	0,919
[55]	VAE+Bi-LSTM+RF	Google News (2600 artículos)	0,899	0,839
		Dataset final (75000)		
[56]	Bi-LSTM + BERT+GNN	Weibo (T=3201,F=3161)	0,756	0,754
		Twitter (T=9010,F=5699)	0,814	0,801
[57]	TriFN	BuzzfeedNews (91 T, 91 F)	0,864	0,878
	(contenido+contexto)	PolitiFact (120 T, 120 F)	0,878	0,880
[58]	LR	Signal Media News(10000 true)	0,670	0,680
	RF	Kaggle (10000 false)	0,700	0,700
	RNN		0,710	0,799
	LSTM		0,664	0,754
	GRU		0,695	0,788
[59]	RvNN, Autoatención	Twitter15 (1118)	0,770	0,814
		Twitter16 (1381)	0,768	0,825
[60]	XGBoost	Kaggle (T=7924, F=7924)	0,966	0,966
[61]	RF	Kaggle1 (3128 T, 3128 F)	0,950	
	GB	Kaggle2 (1872 T, 2137 F)	0,960	
[62]	GNN	Snopes (T=1164, F=3177)		0,800
		Politifact (T=1867, F=1701)		0,691
[63]	ML: XGBoost.	BuzzfeedNews (T=90, F=90)	0,879	0,879
		PolitiFact (T=120, F=120)	0,892	0,892

Tabla 4-3 Estudios con técnicas, *datasets* y resultados obtenidos. Fuente: Autor

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Discusión

En este apartado procederemos a responder a las preguntas de investigación propuestas en los objetivos de este trabajo.

5.1.1 PI 1. *¿Cuáles son las áreas de la IA con aplicación en la detección de fake news?*

En esta investigación se han contrastado numerosos artículos sobre la detección automática de fake news y en todos ellos se han identificado dos componentes principales de IA que han participado en el proceso: NLP y ML.

En cuanto al NLP, se ha detectado que en la mayoría de los trabajos se efectúa un procesamiento del conjunto de datos con el objetivo de obtener una representación vectorial de su contenido, y de su contexto en su caso, con un número de dimensiones reducido, pero muy representativo de la información que necesitamos para identificar las fake news.

Así, se han encontrado trabajos orientados al contenido, que se basan en la frecuencia de aparición de las palabras en un corpus con distintas perspectivas, desde léxicas, como por ejemplo, palabras con sentimientos polarizados, que indican novedad, emoción o empatía, que utilizan términos ambiguos o palabras con doble significado; hasta sintácticas, donde priman los tipos de palabras que se usan nombres, verbos, pronombres, incluso signos de puntuación. También relacionadas con el contenido tenemos algunas que tratan de ver discordancias entre distintas partes de la noticia, principalmente entre titulares y texto principal.

Por último tenemos un número considerable, que basados en el contenido, no especifican de qué tipo son las características utilizadas, porque son extraídas de los datos automáticamente mediante DL.

En cuanto al contexto parece que hay tres focos de interés. Por un lado, tenemos todo lo relacionado con las fuentes, como su credibilidad o su postura. Por otro lado, tenemos los usuarios, su perfil, su reacción ante los contenidos, su relación con las fuentes o sus preferencias históricas. Por último, está la propagación de la noticia, donde lo que se contempla es su rapidez, alcance, profundidad, etc. Algunos modelos realizan combinaciones de los tres, incluso combinándolos con el contenido mismo de la noticia.

Finalmente, tenemos algunos artículos que tratan de crear modelos con grafos de conocimiento desde el contenido de las noticias, que sirvan para realizar verificación de hechos, contrastándolos con otras fuentes, por ejemplo en DBpedia.

5.1.2 PI 2. *¿Qué técnicas se están usando en la detección de fake news y qué resultados?*

En relación a las técnicas utilizadas y su resultado, tenemos básicamente dos bloques, como se ha comentado anteriormente: ML y DL. Pues bien, la tendencia es clara y los resultados acompañan: los modelos más utilizados son de DL y los mejores resultados también son los de esta rama del aprendizaje automático.

En cuanto a las distintas redes neuronales utilizadas, si bien las más utilizadas son las RNN, concretamente en su modalidad de LSTM, los resultados son, en general, igual de buenos en las redes neuronales profundas que en éstas. De hecho, si en la comparación tenemos en cuenta todos los estudios de ambas clases, las LSTM darían, proporcionalmente, peores resultados.

En cuanto a los resultados, parecen bastante prometedores, pero da la sensación de que los niveles de acierto están ligeramente altos para lo que parece razonable, en algunos casos por encima de 0,99. Esto parece indicar que los datos quizás estén sobreexplotados y los modelos estén sobreajustados.

5.1.3 PI 3. *¿Qué retos y limitaciones se están encontrando en la detección de fake news?*

En general, la principal limitación que se está encontrando para la detección automática de fake news consiste en la falta de datos actualizados y debidamente clasificados. A diferencia de los modelos de lenguaje como BERT o GPT, que pueden aprovechar una amplia variedad de datos para su entrenamiento de manera relativamente sencilla, simplemente ocultando palabras de la inmensidad de textos disponibles en internet, la tarea de etiquetar las noticias como verdaderas o falsas no es tan inmediata. Esto se debe a que, inicialmente, no siempre se puede determinar de manera clara y rápida si una noticia es falsa o no. Por lo que no somos capaces de producir noticias clasificadas al ritmo necesario para poder entrenar modelos de ML adecuadamente.

Por otro lado, las noticias generadas sintéticamente carecen de la autenticidad presente en las noticias reales. Al proceder de otras noticias ya existentes y ser creadas mediante algoritmos, suelen tener correlaciones internas entre ellas y sesgos que hacen que al entrenar los modelos tiendan al sobreajuste de los algoritmos.

Para complicar aún más el problema, los creadores de noticias falsas están constantemente generando nuevos contenidos, utilizando técnicas avanzadas y aprovechando las últimas tecnologías en generación de texto e imágenes. Esto provoca que los modelos tengan que estar adaptándose constantemente a las nuevas fake news, que son cada vez más parecidas a las noticias reales. Un hecho a resaltar es que los redactores, periodistas y creadores de contenido en general, también usan cada día más las herramientas generadoras de texto, por lo que el estilo de las noticias reales y falsas también se va pareciendo cada vez más.

Un reto detectado, y quizás uno de los más importantes es la mejora de los modelos híbridos, para que sean capaces de confrontar adecuadamente el contenido de las noticias con el contexto que la rodea de forma rápida y eficaz. Determinadas alteraciones en la información, con ciertos matices, omisiones o cambios en lugares, personas o fechas, puede hacer que algo pase de cierto a falso. Sin embargo, los valores contextuales son más difíciles de manipular y con interfaces a las plataformas se pueden contrastar estos datos sobre fuentes, usuarios y la propagación seguida, para llegar a conclusiones sobre su veracidad.

En resumen, podríamos decir que la principal limitación es la falta de datos adecuados y actualizados, provocada por la rapidez con la que se generan y modifican las noticias falsas, y el principal reto a afrontar es la creación de modelos híbridos que tengan más en cuenta el contexto como elemento verificador.

5.2 Conclusiones

La primera conclusión a la que se ha llegado tras la realización de este estudio es, que si tenemos suficientes noticias clasificadas, seremos capaces de construir un modelo, entrenarlo y predecir, con una probabilidad altísima, si una nueva noticia pertenece a una clase u otra. Pero debido a la constante evolución de los contenidos falsos, probablemente, esto no permitirá, por sí solo, determinar con mucha certeza si es falsa o no.

Esto nos lleva a la segunda conclusión. La detección de fake news es un problema multifacético que precisa de varios enfoques. Es necesario, por supuesto, el estudio del contenido y el entrenamiento con modelos basados en las características lingüísticas del texto, pero igual de necesario es un enfoque basado en el contexto, que permita utilizar todos los metadatos que acompañan a la noticia sobre los creadores, distribuidores y formas de propagación. Por último, no debemos olvidar que estamos verificando noticias y las que son verdaderas se construyen con datos, por lo que una aproximación que sea capaz de extraer y cotejar automáticamente esta información con otras fuentes siempre será de gran ayuda.

Ahondando más en este último punto, la detección de fake news es un problema complejo, porque decir si una noticia es cierta o falsa, en sí mismo, también es un asunto complejo, y tiene bastantes implicaciones, incluso legales. Evidentemente, hay noticias que son fácilmente identificables por un humano como fake news, porque hay datos que permiten hacerlo o porque están redactadas de tal forma que las hace parecerlo. Probablemente las máquinas acaben controlando éstas, y ya con esto habremos reducido bastante el problema. Pero hay mucha información en una noticia que es ambigua y que dependerá, incluso, del sesgo del propio lector el que parezca más o menos cierta. Algunos conjuntos de datos muy populares tienen varias clasificaciones, como LIAR que tiene seis clasificaciones, que van desde verdadero a totalmente falso y engañoso, pasando por mayormente verdadero, medio verdadero, apenas verdadero y falso. Incluso para una persona sería difícil decidir en cuál de las categorías encajaría un contenido.

Como última conclusión, una reflexión. Si al final se consiguiese y se implantase un sistema de catalogación automática de noticias falsas para reducir el nivel de desinformación de la sociedad. ¿Quién decidiría lo que es verdad? ¿Una máquina basándose en los datos? ¿Qué datos y quién se los daría?

5.3 Líneas futuras

A continuación, se comentan algunas posibles líneas futuras de progreso en este campo identificadas durante el desarrollo de este trabajo.

Un elemento que podría tener bastante repercusión en este tema sería la creación de un conjunto de datos de gran tamaño, disponible para los investigadores, multilingüe, actualizado, fiable y lo más libre de sesgo posible. Esto permitiría realizar comparaciones fidedignas entre distintos estudios y avanzar en aquellas direcciones que dieran mejores resultados.

La tecnología *transformer* está demostrando ser sumamente eficaz en muchos ámbitos, llegando sus modelos a generalizar y aprender reglas para las que ni siquiera se les estaba entrenando. Una vía de investigación podría ser utilizar esta tecnología para alcanzar una comprensión más profunda de la semántica de las noticias y su relación con el contexto, aprovechando además, la mayor capacidad y velocidad de procesamiento que ofrecen.

Otra posible vía de estudio sería investigar la posibilidad de utilizar modelos preentrenados de lenguaje en varios idiomas en el modelo, de forma que permitieran utilizar algoritmos de detección sobre plataformas multilingües.

Las noticias cada vez tienen más contenidos audiovisuales, por lo tanto los futuros sistemas de detección deberían ser multimodales para aprovechar la información contenida en estos componentes multimedia.

Por último, dada la velocidad de propagación de las fake news en las plataformas en línea y redes sociales, podría ser interesante que éstas contaran con mejores elementos de trazabilidad que permitieran detectar patrones de noticias falsas más fácilmente. Adicionalmente, y dado que las noticias pueden saltar de una aplicación a otra, sería interesante que hubiera una mayor colaboración entre las distintas plataformas para implementar estrategias comunes de control y mitigación.

6 BIBLIOGRAFÍA

El acceso a los recursos “en línea” recogidos en esta bibliografía se ha realizado en el periodo comprendido entre los meses de octubre y noviembre de 2023, estando todos ellos disponibles en ese momento.

- [1] Fundacion Telefonica, Fake News. La fábrica de mentiras, 2023.
- [2] International Federation of Journalists, «¿Qué son las fake news?,» [En línea]. Available: https://www.ifj.org/fileadmin/user_upload/Fake_News_-_FIP_AmLat.pdf.
- [3] «Las fake news siempre han existido, peros hoy en día se han visto catapultadas por las redes sociales,» Oficina de transferencia de resultados de investigación, UCM, 2020. [En línea]. Available: <https://www.ucm.es/otri/noticias-las-fake-news-siempre-han-existido-pero-hoy-en-dia-se-han-visto-catapultadas-por-las-redes-sociales>.
- [4] D. Murias, «16 fotos manipuladas que pasaron a la historia,» 2016. [En línea]. Available: <https://www.xataka.com/magnet/16-fotos-manipuladas-que-pasaron-a-la-historia>.
- [5] A. Gómez-Pérez, «Inteligencia artificial y Lengua española,» Real Academia Española, 2023. [En línea]. Available: https://www.rae.es/sites/default/files/2023-05/Discurso%20Ingreso%20Asuncion%20Gomez-Perez_0.pdf.
- [6] S. Greengard, «What is Artificial Intelligence & How does it work?,» Datamation, 2019. [En línea]. Available: <https://www.datamation.com/artificial-intelligence/what-is-artificial-intelligence/>.
- [7] A. Géron, Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow, O'Reilly, 2019.
- [8] E. Menasalvas, «Algoritmos de Machine Learning,» Universidad Politécnica de Madrid, 2021. [En línea]. Available: <https://blogs.upm.es/catedra-idanae/wp-content/uploads/sites/698/2021/04/Idanae-1T21.pdf>.
- [9] S. Bag, «Which Machine Learning Algorithm Should You Use By Problem Type?,» Analytics Vidhya, 2020. [En línea]. Available: <https://medium.com/analytics-vidhya/which-machine-learning-algorithm-should-you-use-by-problem-type-a53967326566>.
- [10] R. D. Badra, «The Machine Learners,» 2023. [En línea]. Available: <https://www.themachinelearners.com/metricas-de-clasificacion/>.

- [11] P. West, «What is deep learning and how do I deploy it in imaging?,» Vision Systems Design, 2020. [En línea]. Available: <https://www.vision-systems.com/boards-software/article/14173008/what-is-deep-learning-and-how-do-i-deploy-it-in-imaging>.
- [12] S. Madhavan, «Deep learning architectures,» IBM Developer, 2021. [En línea]. Available: <https://developer.ibm.com/articles/cc-machine-learning-deep-learning-architectures/>.
- [13] IBM, «¿Qué es el procesamiento del lenguaje natural (NLP)?,» [En línea]. Available: <https://www.ibm.com/es-es/topics/natural-language-processing>.
- [14] SAS Institute Inc., «Procesamiento de lenguaje natural (NLP) ¿Qué es y por qué es importante?,» 2023. [En línea]. Available: https://www.sas.com/es_es/insights/analytics/what-is-natural-language-processing-nlp.html.
- [15] Q. Jiao, «A brief survey of word embedding and its recent development,» 2021. [En línea]. Available: <https://ieeexplore.ieee.org/document/9390956>.
- [16] J. Johnson, «A detailed review on word embedding techniques with emphasis on word2vec,» Springer, 2023. [En línea]. Available: <https://doi.org/10.1007/s11042-023-17007-z>.
- [17] T. Mikolov, «Linguistic Regularities in Continuous Space Word Representations,» 2013. [En línea]. Available: <https://aclanthology.org/N13-1090.pdf>.
- [18] T. Mikolov, «Efficient Estimation of Word Representations in Vector Space,» 2013. [En línea]. Available: <https://arxiv.org/pdf/1301.3781.pdf>.
- [19] A. Mittal, «Understanding RNN and LSTM,» 2019. [En línea]. Available: <https://aditimittal.medium.com/understanding-rnn-and-lstm-f7cdf6dfc14e>.
- [20] A. Vaswani, «Attention Is All You Need,» 2017. [En línea]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- [21] Information is beautiful, «The rise and rise of AI. Large Language Models (LLMs),» 2023. [En línea]. Available: <https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-language-models-llms-like-chatgpt/>.
- [22] N. Bentzen, «Como detectar que una noticia es falsa,» EPRS Servicio de Estudios del Parlamento Europeo, 2019. [En línea]. Available: [https://www.europarl.europa.eu/RegData/etudes/ATAG/2017/599386/EPRS_ATA\(2017\)599386_ES.pdf](https://www.europarl.europa.eu/RegData/etudes/ATAG/2017/599386/EPRS_ATA(2017)599386_ES.pdf).
- [23] Instituto Poynter, «International Fact-Checking Nertwork,» 2023. [En línea]. Available: <https://www.poynter.org/ifcn/>.
- [24] F. R. Caja, «El fact checking. Las agencias de verificación de noticias en España,» 2020. [En línea]. Available: https://www.ieee.es/Galerias/fichero/docs_opinion/2020/DIEEE089_2020FERROJ_agencias.pdf.
- [25] Agencia EFE, «EFE Verifica, El servicio de EFE contra la desinformación,» 2023. [En línea]. Available: <https://verifica.efe.com/>.
- [26] X. Zhou, «Fake News: A Survey of Research, Detection Methods, and Opportunities,» 2018. [En línea]. Available: <https://arxiv.org/pdf/1812.00315v1.pdf>.
- [27] F. J. García-Peñalvo, «Desarrollo de estados de la cuestión robustos: Revisiones Sistemáticas de

- Literatura,» 2022. [En línea]. Available: <https://revistas.usal.es/tres/index.php/eks/article/view/28600/27680>.
- [28] M. J. Page, «The PRISMA 2020 statement: an updated guideline for reporting systematic reviews,» 2023. [En línea]. Available: <https://www.bmj.com/content/bmj/372/bmj.n71.full.pdf>.
- [29] M. J. Page, «PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews,» 2021. [En línea]. Available: <https://www.bmj.com/content/bmj/372/bmj.n160.full.pdf>.
- [30] F. J. Rodríguez-Sedano, «Uso de herramienta on-line Parsifal para la elaboración de una revisión sistemática de la literatura (SLR),» 2019. [En línea]. Available: <https://zenodo.org/records/2603914>.
- [31] A. Abedalla, «A Closer Look at Fake News Detection: A Deep Learning Perspective,» de *Proceedings of the 3rd International Conference on Advances in Artificial Intelligence*, Istanbul, Turkey, 2020.
- [32] A. M. Ali, «Deep Ensemble Fake News Detection Model Using Sequential Deep Learning,» *SENSORS*, vol. 22, n° 18, 2022.
- [33] M. Aljabri, «Fake News Detection Using Machine Learning Models,» de *2022 14th International Conference on Computational Intelligence and Communication Networks (CICN)*, 2022.
- [34] S. A. Althubiti, «Natural Language Processing with Optimal Deep Learning Based Fake News Classification,» *CMC-COMPUTERS MATERIALS & CONTINUA*, vol. 73, n° 2, pp. 3529-3544, 2022.
- [35] W. Antoun, «State of the Art Models for Fake News Detection Tasks,» de *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies, ICIoT 2020*, 2020.
- [36] A. Bonet-Jover, «Exploiting discourse structure of traditional digital media to enhance,» *EXPERT SYSTEMS WITH APPLICATIONS*, vol. 169, 2021.
- [37] A. Brasoveanu, «Integrating Machine Learning Techniques in Semantic Fake News Detection,» *NEURAL PROCESSING LETTERS*, vol. 53, n° 5, pp. 3055-3072, 2021.
- [38] Y. Dou, «User Preference-Aware Fake News Detection,» de *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Virtual Event, Canada, 2021.
- [39] M. Islam, «RumorSleuth: Joint Detection of Rumor Veracity and User Stance,» de *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Vancouver, British Columbia, Canada, 2020.
- [40] N. Islam, «Ternion: An Autonomous Model for Fake News Detection,» *APPLIED SCIENCES-BASEL*, vol. 11, n° 19, 2021.
- [41] S. Jadhav, «Fake News Identification and Classification Using DSSM and Improved Recurrent Neural Network Classifier,» *APPLIED ARTIFICIAL INTELLIGENCE*, vol. 33, n° 12, pp. 1058-1068, 2019.
- [42] D. Kasseropoulus, «An Approach Utilizing Linguistic Features for Fake News Detection,» *IFIP Advances in Information and Communication Technology*, vol. 627, p. 646 – 658, 2021.

- [43] R. Khan, «Improved Fake News Detection Method based on Deep Learning and Comparative Analysis with other Machine Learning approaches,» de *2022 International Conference on Engineering and Emerging Technologies (ICEET)*, 2022.
- [44] R. Kumari, «A Multitask Learning Approach for Fake News Detection: Novelty, Emotion, and Sentiment Lend a Helping Hand,» de *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021.
- [45] R. Kumari, «What the fake? Probing misinformation detection standing on the shoulder of novelty and emotion,» *INFORMATION PROCESSING & MANAGEMENT*, vol. 59, n° 1, 2022.
- [46] S. Li, «Stacking-Based Ensemble Learning on Low Dimensional Features for Fake News Detection,» de *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 2019.
- [47] Y. Liu, «FNED: A Deep Network for Fake News Early Detection on Social Media,» *ACM Trans. Inf. Syst.*, vol. 38, n° 3, 2020.
- [48] A. Mosallanezhad, «Domain Adaptive Fake News Detection via Reinforcement Learning,» de *Proceedings of the ACM Web Conference 2022*, Virtual Event, Lyon, France, 2022.
- [49] M. Nadeem, «HyproBert: A Fake News Detection Model Based on Deep Hypercontext,» *SYMMETRY-BASEL*, vol. 2, p. 15, 2023.
- [50] A. Nagaraja, «Fake News Detection Using Machine Learning Methods,» de *International Conference on Data Science, E-Learning and Information Systems 2021*, Ma'an, Jordan, 2021.
- [51] V.-H. Nguyen, «FANG: Leveraging Social Context for Fake News Detection Using Graph Representation,» *Commun. ACM*, vol. 65, n° 4, p. 124–132, 2022.
- [52] N. N. Prachi, «Detection of Fake News Using Machine Learning and Natural Language,» *JOURNAL OF ADVANCES IN INFORMATION TECHNOLOGY*, vol. 13, n° 6, pp. 652-661, 2022.
- [53] S. Ramya, «Attention-Based Deep Learning Models for Detection of Fake News in Social Networks,» *International Journal of Cognitive Informatics and Natural Intelligence*, vol. 15, n° 4, 2021.
- [54] P. B. P. Reddy, «Fake Data Analysis and Detection Using Ensembled Hybrid Algorithm,» de *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 2019.
- [55] S. Sadiq, «High Dimensional Latent Space Variational AutoEncoders for Fake News Detection,» de *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2019.
- [56] Q. Sheng, «Integrating Pattern- and Fact-Based Fake News Detection via Model Preference Learning,» de *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, Virtual Event, Queensland, Australia, 2021.
- [57] K. Shu, «Beyond News Contents: The Role of Social Context for Fake News Detection,» de *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, Melbourne VIC, Australia, 2019.
- [58] H. Telang, «Anempirical Analysis of Classification Models for Detection of Fake News Articles,» de *Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer*

and Communication Technologies, ICECCT 2019, 2019.

- [59] A. P. B. Veyseh, «Rumor Detection in Social Networks via Deep Contextual Modeling,» de *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, Vancouver, British Columbia, Canada, 2020.
- [60] M. Villagrancia Octaviano, «Fake News Detection Using Machine Learning,» de *2021 5th International Conference on E-Society, E-Education and E-Technology*, Taipei, Taiwan, 2021.
- [61] H. E. Wynne, «Content Based Fake News Detection Using N-Gram Models,» de *Proceedings of the 21st International Conference on Information Integration and Web-Based Applications & Services*, Munich, Germany, 2020.
- [62] W. Xu, «Evidence-Aware Fake News Detection with Graph Neural Networks,» de *Proceedings of the ACM Web Conference 2022*, Virtual Event, Lyon, France, 2022.
- [63] X. Zhou, «Fake News Early Detection: A Theory-Driven Model,» *Digital Threats*, vol. 1, nº 2, 2020.