



Artificial Intelligence techniques based on the integration of symbolic logic and deep neural networks: A systematic literature review

Pablo Negro¹ Claudia Pons^{1,2,3}

1. CAETI – Centro de altos estudios en tecnología informática, facultad de tecnología. UAI.
2. LIFIA – Facultad de Informática. Universidad Nacional de La Plata.
3. CIC - Comisión de Investigaciones Científicas de Buenos Aires.

Abstract. Artificial Intelligence is tackled from two predominant but very different approaches: symbolic Artificial Intelligence, which is inspired by mathematical logic and is based on the manipulation of abstract linguistic representations, and non-symbolic Artificial Intelligence, which focuses on the construction of predictive mathematical models from large sample data sets. Significantly, the shortcomings of each of these approaches align with the strengths of the other, suggesting that an integration between them would be beneficial. A successful synthesis of symbolic and non-symbolic artificial intelligence would give us the advantages of both worlds.

This work aims to identify and classify solutions and architectures that use applied Artificial Intelligence techniques, based on the integration of *symbolic and non-symbolic logic (particularly machine learning with artificial neural networks)*, to provide a comprehensive, exhaustive and organized vision of the solutions available in the literature, making them the subject of a carefully designed and implemented SLR (Systematic Literature Review). The resulting technologies are discussed and evaluated from both perspectives: symbolic and non-symbolic Artificial Intelligence.

The PICOC method (*Population, Intervention, Comparison, Outputs, Context*) plus Limits, which determine the scope of the search, has been used to define the research questions and analyze the results. From a total of 65 candidate studies found, 24 articles (37%) relevant to this study were selected. Each study also focuses on different application domains such as intelligent agents, image classification, theorem provers, cyber-security, image interpretation, mathematics, medicine, robotics and general application.

Through the analysis of the selected works, it was possible to classify, organize and explain the different ways in which the deficiencies of non-symbolic Artificial Intelligence are addressed by proposals based on symbolic logic. The study also determined in which stages of the development process said proposals are applied. In addition, the study made it possible to determine which are the logic tools that are preferably applied, for each area and each domain. Although no clear *architectural pattern* has been found, efforts to find a general-purpose model that combines both worlds are driving trends and research efforts.

Resumen. La Inteligencia Artificial se aborda desde dos enfoques predominantes pero muy diferentes entre sí: la Inteligencia Artificial simbólica, que está inspirada en la lógica matemática y se basa en la manipulación de representaciones lingüísticas abstractas, y la Inteligencia Artificial no simbólica, que se centra en la construcción de modelos matemáticos predictivos a partir de grandes conjuntos de datos de muestra. Significativamente, las deficiencias de cada uno de estos enfoques se alinean con las fortalezas del otro, lo que sugiere que una integración entre ellos sería beneficiosa. Una síntesis satisfactoria de inteligencia artificial simbólica y no-simbólica nos daría las ventajas de ambos mundos.

Este trabajo tiene como objetivo identificar y clasificar soluciones y arquitecturas que utilizan técnicas de Inteligencia Artificial aplicada, basadas en la integración *de lógica simbólica y no-simbólica (en particular aprendizaje automático con redes neuronales artificiales)*, para proporcionar una visión integral, exhaustiva y

organizada de las soluciones disponibles en la literatura, haciéndolas objeto de una SLR: Systematic Literature Review (Revisión Sistemática de la Literatura) cuidadosamente diseñada e implementada sobre el tema. Las tecnologías resultantes se discuten y evalúan desde ambas perspectivas: la Inteligencia Artificial simbólica y la no simbólica.

Se ha utilizado el método PICOC (*Población, Intervención, Comparación, Salidas, Contexto*) más Límites, que determinan el alcance de la búsqueda, para definir las preguntas de investigación y analizar los resultados. De un total de 65 estudios candidatos encontrados, se seleccionaron 24 artículos (37%) relevantes para este estudio. Cada estudio, además, se centra en diferentes dominios de aplicación tales como agentes inteligentes, clasificación de imágenes, comprobadores de teoremas, *cyber*-seguridad, interpretación de imágenes, matemática, medicina, robótica y de aplicación general.

A través del análisis de los trabajos seleccionados, se logró clasificar, organizar y explicar las diferentes formas en que las deficiencias de la Inteligencia Artificial no simbólica son abordadas por propuesta basadas en lógica simbólica. El estudio también determinó en qué etapas del proceso de desarrollo dichas propuestas son aplicadas. Complementariamente, el estudio permitió determinar cuáles son las herramientas de la lógica que se aplican preferentemente, para cada área y cada dominio. Si bien no se ha encontrado un *patrón arquitectónico* claro, los esfuerzos por encontrar un *modelo de propósito general* que combine ambos mundos dirigen las tendencias y esfuerzos de investigación.

Keywords: Artificial Intelligence, Machine Learning, Deep Learning, Logic, Hybrid Model.

Palabras clave: Inteligencia Artificial, Aprendizaje Automático, Deep Learning, Lógica, Modelo híbrido.

1 Introducción

Hoy en día, la inteligencia artificial (IA) es una disciplina próspera con muchas aplicaciones prácticas y temas de investigación activos. En los primeros días de la inteligencia artificial, la disciplina abordó y resolvió rápidamente problemas que son intelectualmente difíciles para los seres humanos, pero relativamente sencillos para las computadoras, problemas que pueden describirse mediante una lista de reglas matemáticas formales. El verdadero desafío para la inteligencia artificial resultó ser resolver las tareas que son fáciles de realizar para las personas, pero difíciles de describir formalmente: problemas que resolvemos intuitivamente, que se sienten automáticos, tales como reconocer el habla o rostros en imágenes, Goodfellow et al., (2016).

Esta dicotomía da lugar a dos enfoques predominantes pero muy diferentes entre sí: la Inteligencia Artificial simbólica, que está inspirada en la lógica matemática y se basa en la manipulación de representaciones lingüísticas abstractas, y la Inteligencia Artificial no simbólica, que se centra en la construcción de modelos matemáticos predictivos a partir de grandes conjuntos de datos de muestra.

El aprendizaje profundo o Deep Learning (DL) es un tipo particular de Inteligencia Artificial no simbólica, el cual propone un enfoque biológicamente inspirado que aporta el poder de las redes neuronales profundas al influir en la tarea genérica del aprendizaje por prueba y error, donde su eficacia se ha demostrado de manera convincente en una amplia variedad de ámbitos, Garnelo et al., (2016). A pesar de su éxito innegable, varias investigaciones han hecho evidencia de importantes deficiencias en el aprendizaje profundo contemporáneo (Garnelo et al., 2016; Lake et al., 2017; Marcus, 2018; P. W. Battaglia et al., 2018; Yann LeCun, Yoshua Bengio, 2015), en particular:

Ineficiencia de datos (alta complejidad de la muestra), dado que requieren conjuntos de datos muy grandes para que funcionen de manera eficaz, lo que implica además que son lentos para aprender (Schmidhuber, 2015; Tsividis et al., 2017).

Poca generalización, dado que las redes neuronales de hoy son propensas a fallar desastrosamente cuando se exponen a datos fuera de la distribución en la que fueron entrenadas. Por ejemplo, cambiar solo el color o el tamaño de un objeto en un videojuego podría obligar a un agente entrenado con aprendizaje por refuerzo, a volver a aprender el juego desde cero, Tsividis et al., (2017).

Carecen de la capacidad de razonar en un nivel abstracto, lo que dificulta la implementación de funciones cognitivas de alto nivel como el aprendizaje de transferencia, el razonamiento analógico y

el razonamiento basado en hipótesis. Se han realizado varios intentos recientes para imitar procesos lógicos en redes neuronales (Rocktäschel & Riedel, 2017; Evans et al., 2018).

Finalmente, por su *falta de interpretabilidad* su funcionamiento es en gran medida *opaco* para los humanos, lo que los hace inadecuados para dominios en el que la verificabilidad es importante (Santoro et al., 2017; Garnelo et al., 2016; Lake et al., 2017; Marcus, 2018; P. W. Battaglia et al., 2018).

Por otro lado, las soluciones basadas en lógica podrían abrir nuevas formas de diseñar sistemas inteligentes explicables, con capacidades de razonamiento lógico. La creación de un modelo que *combine ambos enfoques* podría darnos lo mejor de ambos mundos. Sus representaciones estarían fundamentadas y aprendidas de datos con mínimos antecedentes. *Una arquitectura que combine todas estas características podría dar soporte a las propiedades deseadas de eficiencia de datos, poder de generalización e interpretabilidad humana*, Domingos, (2018).

Dado el estado actual de las tecnologías de IA, en su mayoría exitosas en escenarios de aplicaciones bien delimitados, un tema clave para la ingeniería de sistemas inteligentes es la integración de las diversas técnicas de IA: en términos de ingeniería de *software*, no sólo como integrar diversas tecnologías, sino también como preservar la integridad conceptual cuando se combinan enfoques altamente heterogéneos, que dan lugar a múltiples abstracciones de diversa naturaleza, P. Battaglia et al., (2016).

Hoy en día está surgiendo una clara dicotomía dentro del área de la IA. Mientras que el éxito abrumador de las técnicas no-simbólicas como DL básicamente ha borrado la mayoría de las objeciones contra la IA, la preocupación pública sobre el papel que los sistemas inteligentes tendrán en la sociedad, ha planteado nuevos interrogantes, como la necesidad de *explicar* el comportamiento de tales sistemas y hacerlos comprensibles para los humanos, particularmente cuando la IA juega un papel fundamental dentro de las organizaciones, Calegari et al., (2020). En tal sentido, el enfoque general sobre la capacidad de la IA de explicarse requerirá la adopción extensiva de técnicas simbólicas (posiblemente integradas con no-simbólicas) para alcanzar sus objetivos principales, como la observabilidad, interpretabilidad, explicabilidad y responsabilidad, Barredo Arrieta et al., (2020). Como resultado, exactamente cuando el enfoque general de la IA está en las técnicas no-simbólicas, como las que han hecho que la IA funcione en aplicaciones del mundo real, los enfoques simbólicos están comenzando a ser examinados cuidadosamente, dado que esto podría hacer que la IA también se adaptara al entendimiento humano, Domingos, (2018).

Las técnicas basadas en lógica no son sólo las más antiguas, sino que posiblemente representan el camino más directo hacia la comprensión humana. Si no es por otra razón, el planteo de la lógica como el estudio del razonamiento adecuado, se asimila a los procesos cognitivos humanos, McCarthy, (1987).

Además, los enfoques basados en lógica son desde hace mucho tiempo el núcleo de muchos modelos y tecnologías exitosas basadas en agentes. En términos generales, los agentes cognitivos inteligente explotan directamente modelos y tecnologías basados en lógica para el proceso racional, la representación del conocimiento, la comunicación expresiva y la coordinación efectiva, Russell & Norvig, (2010).

En la presente revisión, se pretende entender y representar el estado actual de las tecnologías de *DL potenciadas con sistemas basados en lógica* que son realmente utilizables en la ingeniería de sistemas inteligentes. Por lo tanto, el objetivo de este documento es identificar y clasificar soluciones y arquitecturas que utilizan técnicas de Inteligencia Artificial aplicada, basadas en la integración *de lógica simbólica y no-simbólica (en particular aprendizaje automático con redes neuronales artificiales)*, para proporcionar una visión integral, exhaustiva y organizada de las soluciones disponibles en la literatura, haciéndolas objeto de una SLR cuidadosamente diseñada e implementada sobre el tema.

En tal sentido, el documento está organizado de la siguiente manera. La Sección 2 presenta el método utilizado para la SLR, discutiendo los detalles técnicos desde las preguntas de investigación fundamentales hasta las operaciones técnicas que conducen al material discutido en el documento. La Sección 3 presenta y organiza los resultados obtenidos, luego del análisis de los trabajos primarios, de las tecnologías desarrolladas de acuerdo con una perspectiva basada en modelos híbridos. En la Sección 4 se plantean las limitaciones del estudio con una indicación de la dirección del trabajo futuro.

La Sección 5 plantea la discusión sobre las fuentes y las tecnologías seleccionadas con una visión general de las tecnologías basadas en *Lógica para Deep Learning*, ofreciendo algunos criterios de interpretación sobre su estado de la técnica y perspectivas actuales. Por último, la Sección 6 concluye el artículo con algunas observaciones finales.

Finalmente, todas las siglas utilizadas en la presente revisión, pueden ser consultadas en el Anexo I – Acrónimos y siglas.

2 Método de Investigación

En esta sección se presenta la metodología de investigación utilizada, la cual sigue la estrategia propuesta por Kitchenham et al., (2009). Siguiendo dicha estrategia, en subsecuentes subsecciones del protocolo de revisión, se discuten los criterios de inclusión y exclusión, el proceso de estrategia de búsqueda, el proceso de selección y los procesos de extracción y síntesis de datos.

2.1 Premisas

Hacer una revisión sistemática requiere en primer lugar que los conceptos principales estén definidos con la suficiente claridad, no sólo para precisar su alcance, sino sobre todo para hacer que la revisión en sí sea reproducible, hasta cierto punto. Esto no significa, por supuesto, que nociones como redes neuronales y sistemas basados en lógica deban redefinirse aquí con más detalle de lo que se usa normalmente en la literatura. Pero, al menos, significa que las aceptaciones de términos utilizados en el resto de este documento están bien fundamentadas y son comprensibles, Calegari et al., (2020).

Entonces, al referir a tecnologías de *Deep Learning*, nos referimos a los conceptos vertidos por Goodfellow et al., (2016) y Russell & Norvig, (2010), o a una definición equivalente presentadas en Domingos, (2018) donde las redes neuronales profundas son presentadas como capas de neuronas densas, las cuales se combinan con capas de entrada y salida, y alguna función de activación en sus neuronas; y que para comprimir y parametrizar información, utilizan autoencoders, máquinas de Boltzman o convulsiones.

En términos del método de aprendizaje, nos basaremos en los conceptos vertidos por Goodfellow et al., (2016), donde describen que este enfoque de aprendizaje profundo de IA, permite que las computadoras aprendan de la experiencia y comprendan el mundo en términos de una jerarquía de conceptos, con cada concepto definido a través de su relación con conceptos más simples. Al recopilar el conocimiento de la experiencia, este enfoque evita la necesidad de que los operadores humanos especifiquen formalmente todo el conocimiento que necesita la computadora. Entonces la jerarquía de conceptos permite que la computadora aprenda conceptos complicados construyéndolos a partir de otros más simples. Si se dibuja un gráfico que muestre cómo se construyen estos conceptos uno encima del otro, el gráfico es profundo, con muchas capas.

Menos problemático quizás es definir qué es la lógica: desde Aristóteles, cuya lógica es el instrumento del conocimiento humano, Rijk, (2002), ésta estudia la forma en que sacamos conclusiones y nos expresamos, y se ocupa de cómo formalizarla, Metakides & Nerode, 1996. Dado que existen varios tipos de lógica y se adoptan en general en el contexto amplio de la lógica computacional, esta revisión se enfoca solo en solo la lógica formal de primer orden (LPO) claramente identificable en la literatura, en términos de nombre, definición formal y artículos de referencia.

Sin embargo, el problema principal aquí no concierne ni al *DL* ni a la *LPO*. En cambio, proviene de nuestra elección básica de *enfocarnos en tecnologías basadas en lógica para Deep Learning*, y también de su motivación. De hecho, un punto clave es que las tecnologías basadas en Lógica funcionen realmente en el campo del *Deep Learning*: en consecuencia, en el resto del estudio se ha elegido incluir sólo tecnologías *Deep Learning potenciadas con lógica formal de primer orden* que sean científicamente identificables y técnicamente disponibles. Esto significa, en primer lugar, que debe

existir al menos una publicación científica que defina y describa claramente la tecnología y, por lo tanto, cubra sus aspectos fundamentales; luego, que la tecnología en sí debe mencionarse claramente en la publicación referida, así como técnicamente accesible en el momento de la revisión.

Más precisamente, en este trabajo consideramos como *tecnología Deep Learning basada en lógica* a cualquier arquitectura, *framework* o lenguaje de *software* orientado a agentes que:

- (i) Implique algún modelo lógico claramente definido.
- (ii) Provea alguna reificación¹ tecnológica real.

De hecho, se argumenta que los aspectos tecnológicos están lejos de ser marginales, ya que la tecnología es *la aplicación del conocimiento científico con fines prácticos*, y ahora más que nunca la ingeniería de sistemas inteligentes precisa de manera urgente tecnologías confiables.

Claramente, las tecnologías pueden involucrar lógicas en varios niveles diferentes y de formas dispares, de la misma manera que pueden abordar diversas topologías de una red neuronal. Dentro del alcance de la revisión, en particular, estamos interesados principalmente en *tecnologías que combinan redes neuronales profundas con sistemas basados en lógica de primer orden*.

En La Fig. 1 se muestra el proceso general. Para la fase de planificación, se elaboró una descripción detallada del protocolo de investigación. Las siguientes subsecciones presentan la especificación del objetivo y las preguntas de investigación y detallan los pasos de la fase de ejecución. Durante la etapa de ejecución se implementa el proceso definido durante la fase de planificación. También cabe destacar que el proceso es iterativo y puede requerir revisiones, Kitchenham et al., (2009). Durante la etapa de reporte, se clasifican y describen los resultados obtenidos, se discuten los resultados y se presentan las conclusiones.

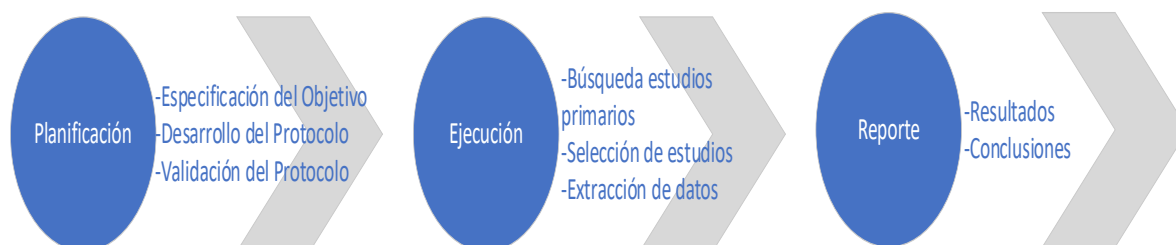


Figura 1. - Diagrama de fases y pasos de la investigación en este estudio

2.2 Especificación del Objetivo

Para la presente revisión, el objetivo se estableció de la siguiente manera:

Identificar y clasificar soluciones y arquitecturas que utilizan técnicas de Inteligencia Artificial aplicada, basadas en la integración *de lógica simbólica y no-simbólica (en particular aprendizaje automático con redes neuronales artificiales)*.

A partir de este objetivo, se deriva un conjunto de preguntas de investigación (PI) y palabras clave (PC).

¹ Técnica de programación orientada a objetos que consiste en tener un tipo de datos para una abstracción.

2.3 Preguntas de Investigación

Todas las SLR se basan en preguntas de investigación fundamentales. Una revisión sistemática intenta identificar, evaluar y sintetizar toda la evidencia empírica que cumple con los criterios de elegibilidad pre especificados para responder una pregunta de investigación determinada. Los investigadores que realizan revisiones sistemáticas utilizan métodos explícitos destinados a minimizar el sesgo, a fin de producir hallazgos más confiables que se puedan utilizar para informar la toma de decisiones.

En particular, la presente revisión sistemática tiene como objetivo responder las siguientes preguntas:

PI 1. *¿Cuál de las deficiencias del DL es abordada/mejorada por la propuesta descrita en el artículo?*

PI 2. *¿En qué parte del proceso de DL se aplica Lógica?*

PI 3 *¿Que herramientas de la lógica se aplican? ¿Qué tipos de tareas se abordan? ¿Sobre cuáles dominios?*

Las preguntas anteriores dirigen el proceso de búsqueda, al articular la forma en que los artículos y las tecnologías se buscan, examinan y condicionalmente seleccionan para la revisión. Luego, se utilizó el método PICOC propuesto por Petticrew & Roberts, 2008 para definir el alcance de la revisión:

Población (P): “Deep Learning”, “First Order Logic”, “Deep Neural Networks”, “Deductive reasoning”, “Relational thinking”, “Inference Engines”, “Meta-interpretive learning”, “Symbolic logic”.

Intervención (I): realizar una revisión sistemática de la literatura sobre arquitecturas de *software*, modelos simbólico-neuronal híbrido basado en DL y Lógica simbólica.

Comparación (C): sin comparación

Salidas (Outputs - O): precisión, interpretabilidad, tecnología, implementación, arquitectura, modelo.

Contexto(C): Contextos relacionados con tecnología aplicada e IA (industria, empresa, negocio, academia)

Límites: Artículo de Revista académica, Artículo en conferencia, Página de libro.

2.4 Fuentes

Las fuentes electrónicas utilizadas en el proceso de búsqueda son de dos tipos:

- (i) Bases de datos bibliográficas generales.
- (ii) Fuentes específicas, como conferencias, simposios y talleres, sobre los temas centrales.

Los primeros se buscan de forma intensiva a través de consultas específicas, mientras que los segundos se exploran extensamente, artículo por artículo. Esto se debe a que a menudo faltan talleres pequeños y especializados en las principales bases de datos bibliográficas utilizadas para nuestra

búsqueda basada en consultas; en cambio, normalmente se puede contar con la indexación completa de las principales conferencias y revistas internacionales.

En consecuencia, el escrutinio exhaustivo de todos los trabajos publicados en los talleres especializados sobre temas cercanos al *Deep Learning* y *lógica de primer orden* es mayoritariamente destinado a complementar la búsqueda intensiva realizada a través de los motores bibliográficos, de modo que resulte en la búsqueda más completa posible en general.

En particular, consideramos ampliamente los artículos publicados en simposios tales como:

- International Conference on Learning Representations
- International Joint Conference on Artificial Intelligence (IJCAI)
- International Conference on Machine Learning (ICML)
- AAAI Conference on Artificial Intelligence

En detalle, las fuentes utilizadas para la búsqueda intensiva (a través de consultas basadas en palabras clave) son las siguientes:

- Google Scholar²
- IEEE Xplore³
- ScienceDirect⁴
- SpringerLink⁵
- ACM Digital Library⁶

Cada una de estas fuentes se consulta con las siguientes combinaciones de palabras clave:

PC1. Todas las formas de decir Deep learning

(“Deep Learning” OR “Neural Networks”)

PC2: Todas las formas de decir lógica

(“First Order Logic” OR “Symbolic logic” OR “deductive reasoning” OR “Relational thinking” OR “inference engines” OR “Meta-interpretive learning” OR “Logic programming”)

PC3: Todas las Intervenciones

(“hybrid model” OR “integrated model”)

PC4: todas las salidas (definidos en la O del PICOC)

(“Accuracy” OR “interpretability” OR “technology” OR “implementation” OR “architecture” OR “model”)

La consulta resultante vendría dada de la siguiente manera: PC1 AND PC2 AND PC3 AND PC4.

Los primeros cincuenta resultados devueltos se consideran candidatos potenciales. Primero se verifica la ocurrencia de un enfoque basado en la *lógica de primer orden en DL* para cada artículo; Luego, en caso de que haya concordancia de términos, se busca y verifica la existencia de la tecnología en todo el documento.

² <http://scholar.google.com>

³ <http://ieeexplore.ieee.org>

⁴ <http://www.sciencedirect.com>

⁵ <https://link.springer.com>

⁶ <http://dl.acm.org>

2.5 Clasificación de artículos

Los artículos resultantes de la actividad de búsqueda anterior se clasifican de acuerdo con tres clases principales:

Artículos primarios: Aquellos artículos que presentan y describen una tecnología de interés.

Artículos auxiliares: Aquellos artículos que mencionan una tecnología de interés, lo que significa que la tecnología mencionada se ha empleado de alguna manera en el desarrollo que contribuye al artículo.

Artículos secundarios: Aquellos artículos que analizan tecnologías de interés.

Dada su relevancia obvia para una encuesta sistemática, los artículos secundarios no se utilizan sólo como fuente directa donde se expliquen tecnologías de interés; en cambio, se busca en sus referencias bibliográficas extensas y organizadas de forma recursiva en busca de posibles elementos de interés.

Este documento tiene como objetivo comprender y representar el estado actual de las tecnologías basadas en *lógica de primer orden disponibles para DL*; por lo tanto, esta revisión se centra en aquellas tecnologías que realmente se pueden utilizar en la ingeniería de sistemas inteligentes.

2.6 Selección de trabajos y criterios de exclusión

Primero eliminamos los duplicados en el conjunto de artículos recuperados. Esta tarea preliminar garantiza que no se analizará el mismo documento más de una vez. Después de eso, todos los artículos se recopilaron en una sola hoja de cálculo de cada estrategia de búsqueda y se llevó a cabo un proceso de dos pasos:

Paso 1: los resultados se analizaron únicamente por título, palabras clave y lugar (revista o conferencia), para excluir artículos que cumplan algunos de los siguientes requisitos:

- No estén en inglés.
- Publicaciones no revisadas por pares.
- No aparecen términos de búsqueda importantes en palabras clave.
- Libros, editoriales, tutoriales, paneles, prefacios, opiniones, cartas, diapositivas y cualquier obra que pueda considerarse literatura gris.
- Año de publicación anterior al año 2015 inclusive.
- Artículos duplicados.
- Artículos que no tienen acceso completo.

Fuera del alcance: estos son trabajos irrelevantes que fueron recuperados debido a una mala ejecución de la búsqueda o debido a errores humanos en las búsquedas manuales. Se considera que este criterio de exclusión es el más eficaz.

Límites: artículos que no pertenecen al dominio de IT / CS / SE / IS (Tecnología de la información / Ciencias de la computación / Ingeniería de *software* / Sistemas de información)

Otros: También se excluyeron las tesis de doctorado o maestría, bajo el supuesto de que publicaciones relevantes, resultantes de la investigación cubierta por las tesis, estaban disponibles e incluidos en el conjunto de artículos recuperados.

Paso 2: De los artículos del Paso 1 se evaluaron sus títulos, resúmenes y palabras clave donde se aplicaron los criterios de inclusión y exclusión.

Finalmente, cada artículo candidato fue leído en su totalidad para decidir si cumple un conjunto de características y criterios de calidad. Luego de la lectura completa de los trabajos, se identificaron otros trabajos relevantes. Concretamente, dos artículos que también se leyeron en profundidad para responder las preguntas descritas anteriormente.

2.7 Extracción de datos

El proceso de extracción de datos es un proceso iterativo e incremental que se ha dividido en varias etapas en las que se han llevado a cabo diferentes actividades. para actualizar dicho conocimiento según se describe en Moher et al., (2010)

Se ha desarrollado un Formulario de Extracción de Datos (FED); una hoja de cálculo para recopilar todos los datos extraídos de los trabajos primarios seleccionados. Se utilizó una plantilla para organizar los elementos de interés.

En primer lugar, se han identificado los resultados obtenidos tras aplicar las cadenas de búsqueda. Para esta tarea se utilizó Zotero® como herramienta de recolección y organización de trabajos. En segundo lugar, se analizó el título, el resumen y las palabras clave de cada trabajo, y se aplicaron los criterios de inclusión y exclusión. Los trabajos se organizaron en otra hoja dentro del mismo documento y cada trabajo se ha marcado como candidato o no, según los anteriores criterios. Finalmente, cada trabajo candidato se leyó en detalle para decidir si son relevantes respecto de las preguntas de investigación.

3 Resultados de la Revisión Sistemática

En esta sección se presentan los datos obtenidos tras el proceso de extracción y el análisis de los trabajos seleccionados. Los resultados del análisis preliminar en la presente revisión, revelaron 65 estudios para mayor consideración.

Los resultados obtenidos después de llevar a cabo este proceso se describen en la Fig. 2, y se detallan a continuación:

- Después de aplicar las cadenas de búsqueda en cada fuente, se han obtenido 93 artículos, de los cuales 2 son de ACM, 45 de GoogleScholar, 10 de IEEE, 17 de ScienceDirect, y 19 de SpringerLink.
 - Después de eliminar los duplicados, quedaron 73 artículos.
 - Una vez que se han aplicado los criterios de inclusión y exclusión al título, el resumen y las palabras clave, quedan 65 artículos.
 - Finalmente, se han analizado en profundidad un total de 24 artículos que son los más relevantes para el presente estudio (25% de los trabajos únicos recuperados).
-

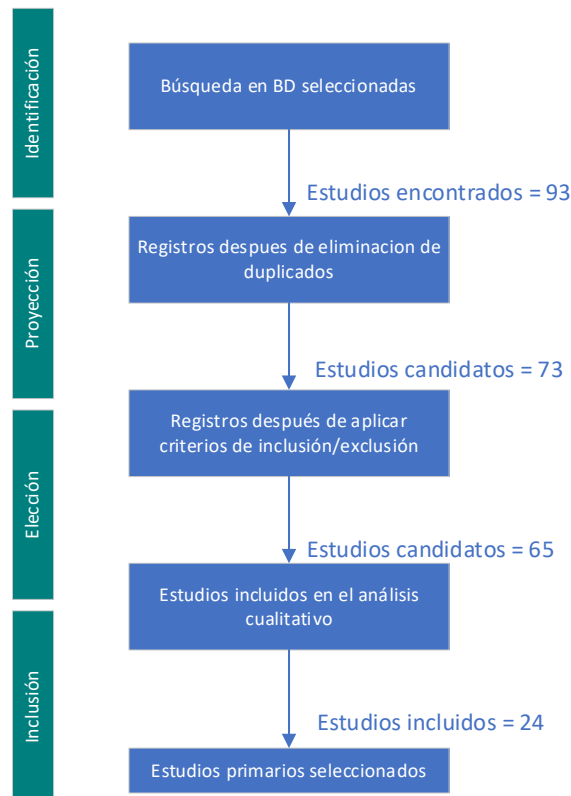


Figura 2. - Procedimiento de búsqueda de estudios primarios

3.1 P11. ¿Cuál de las deficiencias del DL es abordada/mejorada por la propuesta descrita en el artículo?

Esta pregunta de investigación se sub clasifica y aborda basada en las diferentes deficiencias del DL. En tal sentido, se definen las siguientes categorías:

- Ineficiencia de los datos
- Falta de generalización
- Falta de razonamiento
- Falta de explicabilidad o comprensión humana

Cabe destacar que, los trabajos agrupados en cada una de las categorías presentadas, aborda el tema en cuestión y aporta una solución que pretende resolver la deficiencia en cuestión.

3.1.1 Ineficiencia de Datos

Los modelos profundos a menudo requieren grandes cantidades de datos de entrenamiento, lo que puede limitar su aplicación en situaciones en las que los datos no están tan fácilmente disponibles. Basados en esta idea, se dan los primeros pasos hacia un enfoque práctico para entrenar modelos secuenciales profundos utilizando una combinación de datos y conocimiento temporal previo, Xie et al., (2021).

Para abordar este problema dentro de los sistema de DL, Melin et al., (2018) proponen un modelo que utiliza lógica difusa con el objetivo de crear aproximaciones matemáticas para resolver

ciertos y determinados problemas. Los autores sostienen que este tipo de lógica produce resultados precisos a partir de datos imprecisos.

En la misma dirección Pal & Kar, (2019) han utilizado lógica difusa de tipo 2 general para obtener los pesos en la red neuronal y tratar de lidiar con la incertidumbre en los datos reales. En este trabajo se plantea que los sistemas de lógica difusa de orden superior son capaces de lidiar con una gran cantidad de incertidumbres en aplicaciones del mundo real. Los autores plantean un modelo basado en *zSlices*, donde se aplican pesos difusos entre la capa de salida oculta y el error difuso en la capa de salida. El concepto de *zSlices* basados en intervalos se ha utilizado para obtener los pesos difusos generales de tipo 2 para la arquitectura de una red neuronal. Al cortar un conjunto difuso de tipo 2 general en la tercera dimensión, denotado como z en algún nivel, digamos z_i , obtenemos *zSlices*. Un *zSlice* \check{Z}_i es un conjunto difuso de tipo 2 de intervalo con grado de membresía z_i dentro del rango $0 \leq z_i \leq 1$. Aquí, los conjuntos difusos de tipo 2 generalizados son conjuntos difusos de tipo 2 de intervalo basados en *zSlices*, los cuales son capaces de capturar la incertidumbre intra-intermedia que se encuentra dentro de los pesos de la arquitectura de la red neuronal de tres capas con aprendizaje de retro propagación.

En de Penning et al., (2014) se plantea la utilización de conocimiento del dominio, y se sostiene que el comportamiento aprendido se puede extraer para actualizar el dicho conocimiento existente para su validación, informes, mantenimiento, evolución y retroalimentación. Además, el enfoque permite codificar el conocimiento del dominio en el modelo y se ocupa de la incertidumbre en los datos del mundo real.

En Li & Srikumar, (2019) también se sugiere que el entrenamiento *end-to-end* exige de una gran cantidad de ejemplos de entrenamiento, donde entrenar una red típica para la traducción automática puede requerir millones de pares de oraciones. En este trabajo se razona sobre las dificultades y los costos de curar grandes cantidades de datos anotados y, en consecuencia, plantean la problemática de que no se disponga de conjuntos de datos masivos para nuevas tareas, dominios o idiomas.

Como solución a este problema de datos insuficientes o incompletos en Borges et al., 2006 se plantea la inserción de conocimientos previos donde advierten que, en todos los casos, las redes lograron un desempeño efectivo al aprender operadores temporales.

Por su parte, Diligenti et al., (2017) sugiere que el aprendizaje profundo permite desarrollar representaciones de características y entrenar modelos de clasificación de una manera totalmente integrada. Sin embargo, aprender redes profundas es bastante difícil, y mejora en arquitecturas superficiales sólo si una gran cantidad de datos de entrenamiento están disponibles. Inyectar conocimientos previos al *learner* es una forma basada en principios de reducir la cantidad de datos de formación necesarios, ya que este no necesita inducir el conocimiento a partir de los datos en sí.

El modelo presentado en Riegel et al., (2020) es diferenciable *end-to-end*, y el aprendizaje minimiza una función de pérdida novedosa que captura la contradicción lógica, lo que proporciona resistencia al conocimiento inconsistente. También permite la suposición de mundo abierto al mantener límites en los valores de verdad que pueden tener semántica probabilística, lo que produce resistencia al conocimiento incompleto.

En el artículo presentado en Marra et al., (2019), presentan LYRICS, que es una herramienta para IA implementada en TensorFlow (TF). LYRICS proporciona un lenguaje de entrada que permite definir un conocimiento arbitrario previo como lógica de primer orden (LPO). En este trabajo se muestra la generalidad del enfoque presentando algunos casos de uso del lenguaje, incluida la verificación de modelos, el aprendizaje supervisado y la clasificación colectiva. La introducción de conocimientos previos en el proceso de aprendizaje es un paso fundamental para superar estas limitaciones, dado que no requiere que el proceso de entrenamiento induzca las reglas del conjunto de entrenamiento, por lo que se reduce la cantidad de datos de entrenamiento requeridos. Además, el uso de conocimientos previos se puede utilizar para expresar el comportamiento deseado del *learner* en cualquier entrada, proporcionando mejores garantías de comportamiento en un entorno adverso o no controlado.

3.1.2 Falta de generalización

Sinha et al., (2020) estudian la tarea de la generalización lógica, en el contexto del razonamiento relacional utilizando *Graph Neural Networks* (GNN). En particular, estudian cómo las GNN pueden inducir reglas lógicas y generalizar combinando estas reglas de formas novedosas después del entrenamiento.

Para Ramirez et al., (2019) la hibridación de métodos puede *aumentar el rendimiento* de un sistema y aprovechar los beneficios que ofrecen estas técnicas para resolver problemas complejos.

Del mismo modo, Wang & Pan, (2020) demuestran la efectividad y generalización del modelo propuesto en múltiples tareas de extracción de información. Para mejorar el rendimiento de extracción en el dominio del procesamiento natural del lenguaje (PNL), proponen incorporar conocimiento del dominio como reglas lógicas que se integran en el sistema de aprendizaje de representación a través de un marco unificado.

La solución propuesta por Q. Li et al., (2021), exhibe una fuerte generalización sistemática con una precisión general del 72%, superando a los métodos neuronales de *end-to-end* en casi un 33%.

Alzaeemi et al., (2019) basan su trabajo en la performance de los modelos híbridos, haciendo foco en la introducción de lógica para el cálculo de parámetros y para decrementar la cantidad de neuronas en las capas ocultas.

El trabajo presentado en Barceló et al., (2020) tiene como objetivo responder a la pregunta de cuáles son los clasificadores de nodos que pueden ser capturados por arquitecturas GNN (*Graph Neural Network*) como AC-GNN (*Aggregate-Combine, Graph Neural Network*). Para empezar a responder a esta pregunta, proponen centrarse en clasificadores lógicos, es decir, en fórmulas unarias expresables en lógica de predicados de primer orden (LPPO): dicha fórmula clasifica cada nodo v dentro del grafo, según si la fórmula es válida para tal nodo v o no, y validan experimentalmente sus hallazgos mostrando que la expresividad teórica de los AC-GNN (*Aggregate-Combine-Readout, Graph Neural Network*), así como las diferencias con los AC-GNN, se pueden observar cuando la red aprende de los ejemplos. Para los AC-GNN, un punto de partida significativo para medir su poder expresivo es la lógica FOC_2 , el fragmento de lógica LPO que solo permite fórmulas con dos variables, pero a su vez permite utilizar cuantificadores de conteo de la forma $\exists^{\geq N} \varphi$, lo cual establece que existen al menos N nodos que satisfacen la fórmula φ . En particular, muestran que en los datos de gráficos sintéticos que se ajustan a las fórmulas FOC_2 , los AC-GNN luchan por ajustarse a los datos de entrenamiento, mientras que los AC-GNN pueden generalizar incluso a grafos de tamaños que no son vistos durante el entrenamiento.

3.1.3 Falta de razonamiento

Para soportar esta falencia Manhaeve et al., (2018), presentan un *framework* donde las redes neuronales y la programación lógica probabilística se integran de una manera que explota la expresividad y las fortalezas de ambos mundos y se puede llevar adelante un entrenamiento *end-to-end* basándose en ejemplos.

En el trabajo propuesto por Han et al., (2021), los autores utilizan enfoques de razonamiento lógico simbólico, que incluyen el aprendizaje meta-interpretativo y programación lógica de primer orden para aportar conocimientos básicos y remediar el lago de información de supervisión.

En el artículo presentado por Cai et al., (2017) se propone un método de razonamiento basado en *Deep Neural Network* (DNN). El concepto básico de este método es entrenar redes neuronales con retroalimentación profunda y hacer que sean capaces de manipular símbolos. En particular, este método permite a las redes neuronales realizar el razonamiento con variables en la lógica de predicados. Además, amplían la aplicación de este método al descubrimiento de axiomas y permiten que las redes neuronales aprendan el *sentido común* de los datos.

3.1.4 Falta de explicabilidad

Para hacer frente a esta deficiencia, AmirHosseini & Hosseini, (2019) proponen mejorar las compensaciones entre la precisión y la interpretabilidad del sistema de inferencia difuso (FIS), ajustando los parámetros del modelo mediante un algoritmo evolutivo que explora el espacio de búsqueda de manera más inteligente que otros algoritmos propuestos.

Mahdavifar & Ghorbani, (2020) proponen un sistema experto de redes neuronales integradas en profundidad (DeNNeS), que extrae reglas refinadas de una arquitectura de red neuronal profunda (DNN) entrenada para sustituir la base de conocimientos de un sistema experto. Posteriormente, la base de conocimientos se utiliza para clasificación de incidentes *invisibles* de seguridad, y para informar al usuario final de la regla correspondiente que hizo esa inferencia.

Por su parte Cocarascu et al., (2018), durante el entrenamiento, utilizan un codificador automático para clasificar las características de los ejemplos de entrada, como más o menos representativos. Estos ejemplos de formación están etiquetados como pertenecientes a una de dos clases dadas (resultados). Luego se usa esa clasificación para seleccionar un subconjunto de características de mayor rango, de modo que la restricción de los ejemplos de entrenamiento de estas características sea coherente (es decir, sin dos restricciones que tengan las mismas características, pero diferentes resultados). La restricción resultante de los ejemplos de entrenamiento se asigna luego a un *framework* de argumentación abstracto que, a su vez, se asigna a un conjunto de reglas lógicas, de la variedad de la programación lógica.

En el trabajo desarrollado en Csiszár et al., (2020), los autores proponen una sugerencia sobre cómo aún se puede crear un marco teórico sintetizando los mundos de la lógica continua y los sistemas de toma de decisión de múltiple criterio (TDMC), y examinan las principales propiedades del operador de preferencia en sistemas nilpotentes⁷.

Este marco coherente admite una posible aplicación de los resultados en el campo de la inteligencia artificial, como un paso importante hacia la interpretabilidad de los modelos neuronales.

El objetivo del proyecto publicado en Schmid & Finzel, (2020) es combinar enfoques de aprendizaje profundo de caja negra, con aprendizaje automático interpretable para la clasificación de diferentes tipos de imágenes médicas, para combinar la precisión predictiva del aprendizaje profundo y la transparencia y comprensibilidad de modelos interpretables.

En este trabajo, los autores plantean como objetivo desarrollar un enfoque para una asociación humana-IA equilibrada, al hacer que las decisiones basadas en el aprendizaje automático en la medicina sean transparentes, comprensibles y corregibles. El principal resultado del proyecto será un *framework* para una interfaz de explicación que se basa en explicaciones mutuas.

En el estudio presentado en Dombi & Csiszár, (2021), los autores demuestran que los sistemas lógicos nilpotentes ofrecen un marco matemático apropiado para una hibridación de modelos neuronales y lógicos nilpotentes continuos, lo que ayuda a mejorar la interpretabilidad y la seguridad del aprendizaje automático.

Uno de los mayores desafíos es la creciente necesidad de abordar el problema de interpretabilidad para mejorar la transparencia del modelo, rendimiento y seguridad.

La combinación de redes neuronales profundas con reglas lógicas estructuradas y herramientas de decisión de múltiple criterio, donde los operadores lógicos se aplican a los clústeres creados en la primera capa, contribuye a la reducción de la naturaleza de caja negra de los modelos neuronales.

A modo de resumen, el gráfico de barras de la Fig. 3 presenta la distribución de trabajos según la deficiencia abordada.

⁷ En matemática, un elemento x de un anillo R se dice que es *nilpotente* si existe algún entero positivo n tal que $x^n = 0$. El término se utiliza para describir elementos que desaparecen cuando se elevan a una potencia.

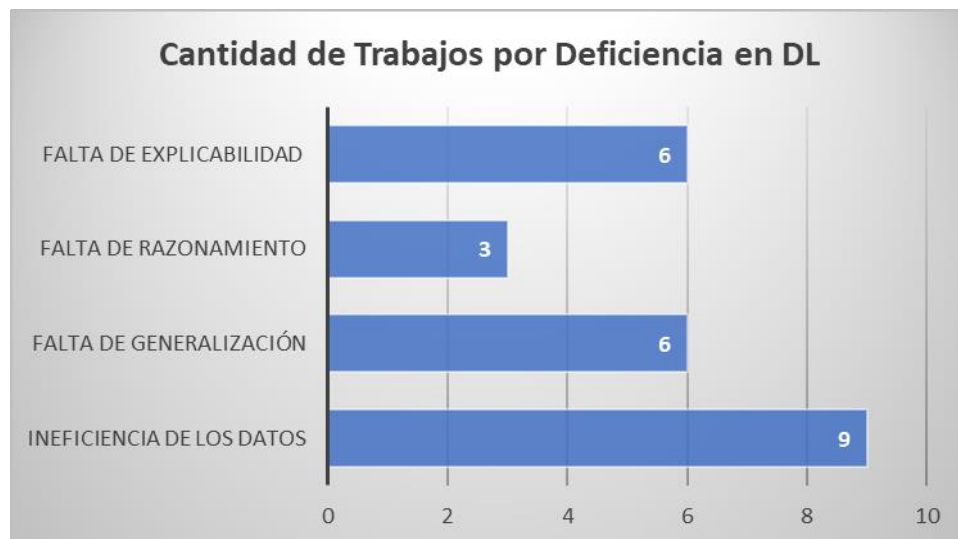


Figura 3. Cantidad de artículos por cada deficiencia de DL abordada

3.2 PI2. ¿En qué parte del proceso de DL se aplica Lógica?

Con esta pregunta de investigación, se intenta comprender cómo se aplican las herramientas basadas en lógica en el proceso de desarrollo de los modelos basados en IA no-simbólica. Concretamente, se busca entender si las distintas variantes de la lógica se utilizan para, por ejemplo, encontrar los hiper-parámetros óptimos, para encontrar las características, para mejorar los datos de entrenamiento, o para acelerar el aprendizaje.

3.2.1 Encontrar los hiper-parámetros óptimos

En el trabajo desarrollado en Pal & Kar, (2019), los autores actualizan los pesos de la red neuronal con la ayuda de conjuntos difusos de tipo 2 generales, para hacer la red más robusta en la captura de incertidumbres debido a la asignación de algunos grados de pertenencia a valores nítidos. Los conjuntos difusos tipo 2 se usan para modelar incertidumbre e imprecisión, y son esencialmente conjuntos difusos en los que los grados de pertenencia son conjuntos difusos de tipo 1. La utilización de sistemas difusos tipo 2 se justifica cuando existe un alto nivel de incertidumbre, en casos donde se busca encontrar mejores resultados que la lógica difusa tradicional, Díaz et al., (2009).

En el trabajo presentado en AmirHosseini & Hosseini, (2019), los autores desarrollan un modelo híbrido que busca optimizar los parámetros utilizando un enfoque de evolución difusa-diferencial (*fuzzy-DE*). Utilizan también, para el proceso de diagnóstico de metástasis hepática, un motor de inferencia difuso, donde aplican reglas difusas como base de conocimientos, incluidos los conjuntos difusos y la membresía difusa del sistema de inferencia, de acuerdo con las variables difusas de entrada.

Por otro lado, Manhaeve et al., (2018) presentan un enfoque para entrenar conjuntamente los parámetros de hechos probabilísticos y redes neuronales en programas DeepProbLog. El aprendizaje se realiza mediante el uso de ProbLog para calcular el gradiente de la pérdida que luego se usa en métodos estándar basados en gradiente descendente, para optimizar los parámetros tanto en el programa de lógica probabilística como en las redes neuronales.

En el trabajo presentado en MahdaviFar & Ghorbani, (2020), los autores utilizan programación lógica dentro de un sistema experto para la optimización de parámetros de la DNN. El sistema experto

de DNN se integra para detectar ataques cibernéticos, que aprende patrones subyacentes en el conjunto de datos e ignora *outliers*. Debido a la naturaleza estadística de las redes neuronales, éstas podrían ser una buena sustitución a las reglas de la base de conocimientos elaboradas manualmente. Sus algoritmos de aprendizaje podrían ser instrumentos para construir reglas *if-then* y alimentarlas a los sistemas expertos. A la inversa, la salida de los sistemas expertos también podría ser una entrada a la red neuronal.

En el trabajo desarrollado por Csiszár et al., (2020), los autores utilizan lógica difusa, como una herramienta de lógica continua, para poder razonar sobre los valores de los pesos y parámetros de la red. En este artículo, los autores proponen un *framework* para modelar el pensamiento humano mediante el uso de herramientas de ambos campos: operadores lógicos difusos y operadores de agregación y preferencia. En este *framework*, la agregación, la preferencia y los operadores lógicos son descritos por la misma función generadora unaria. De manera similar a la implicación definida como una composición de la disyunción y el operador de negación, los operadores de preferencia se introdujeron como una composición del operador de agregación y el operador de negación. En teoría clásica, la preferencia es una relación binaria estrechamente relacionada a las implicaciones:

$$xRy \Leftrightarrow \text{“}y \text{ no es peor que } x\text{”}$$

Las preferencias entre alternativas también pueden describirse mediante un valor relación de preferencia p , tal que el valor $p(x, y)$ se normaliza, y se introduce como el grado en el que el enunciado “*y no es peor que x*” es verdadero.

$$p(x, y) = \text{verdad de } (y \geq x).$$

Aquí, p es una función continua que es estrictamente decreciente en la primera variable, y estrictamente creciente en la segunda variable, mientras que $p(x, y) = n(p(y, x))$ también debe ser válido, siendo $n(x)$ el operador de negación.

En el trabajo desarrollado por Dombi & Csiszár, (2021), los autores presentan un modelo neuronal nilpotente, donde operadores lógicos nilpotentes⁸ combinados con herramientas de decisión multicriterio, se implementan en las capas ocultas de las redes neuronales. En este modelo solo se deben aprender los pesos de la primera capa (parámetros de los hiperplanos que separan el espacio de decisión).

En el modelo neuronal nilpotente, las funciones de activación (*Squashing Function*⁹) en la primera capa son funciones de pertenencia que representan valores de verdad de las desigualdades, normalizando así las entradas. Al mismo tiempo, las funciones de activación en las capas ocultas modelan la función de corte (para evitar el problema del gradiente de desaparición) en los operadores lógicos nilpotentes. La idea básica del uso de lógica continua es la sustitución del espacio de valores de verdad (T, F) por un intervalo compacto como $[0; 1]$. Esto significa que las entradas y salidas de las puertas lógicas extendidas son números reales del intervalo unitario, que representan valores de verdad de desigualdades. Los cuantificadores $\forall x$ y $\exists x$ se reemplazan por Sup_x e Inf_x , y las conectivas lógicas son funciones continuas. Basados en esta idea, el pensamiento humano y el lenguaje natural se puede modelar de una manera sofisticada.

Los sistemas lógicos nilpotentes ofrecen un nuevo enfoque para diseñar redes neuronales utilizando lógica continua, ya que las funciones de pertenencia (que representan el valor de verdad de una desigualdad), y también los operadores lógicos nilpotentes son modelados por perceptrones.

Schmid & Finzel, (2020) introducen la programación lógica inductiva (ILP) como un enfoque poderoso de aprendizaje automático interpretable que, naturalmente, permite combinar el razonamiento

⁸ Un sistema conectivo es nilpotente, si la conjunción c es una t -norma nilpotente, y la disyunción d es una t -conorma nilpotente.

⁹ Aproximación del rectificador ReLU. Es una función que dirige la entrada hacia uno de los extremos de un pequeño intervalo. En las redes neuronales, estos se pueden usar en los nodos de una capa oculta para aplastar la entrada. Esto introduce la no linealidad a la NN y permite que la NN sea efectiva.

y el aprendizaje. La programación lógica inductiva permite aprender modelos compuestos de tales reglas lógicas, mediante la combinación del razonamiento y el aprendizaje de forma natural.

Las teorías de fondo pueden explotarse durante el aprendizaje, y las reglas aprendidas se pueden combinar con reglas de inferencia predefinidas opcionalmente. Se ha demostrado que las reglas aprendidas con ILP pueden apoyar la toma de decisiones humanas en dominios complejos.

3.2.2 Encontrar las características

En Melin et al., (2018) se diseñó un modelo híbrido que utiliza redes neuronales modulares y lógica difusa para proporcionar el diagnóstico de riesgo de hipertensión de una persona. Con el objetivo de realizar una integración de los resultados de los módulos, las redes neuronales modulares utilizan integradores de respuesta como la lógica difusa. En esta investigación se utilizan sistemas difusos como integradores de respuesta, porque de esta forma se puede manejar la incertidumbre en las decisiones.

En el modelo presentado en Sinha et al., (2020), los autores usan lógica para predecir relaciones entre nodos de una GNN (*Graph Neural Network*). En lugar de razonar en un solo grafo de conocimiento estático durante el entrenamiento y las pruebas, consideran el entorno en el que el modelo debe aprender a generalizar en grafos no vistos durante la evaluación.

Definen un gran conjunto de reglas que se basan en lógica proposicional. Luego dividen el conjunto de reglas en subconjuntos superpuestos, que se usan para definir los subconjuntos únicos. Finalmente, dentro de cada subconjunto generan varios grafos de conocimiento que son gobernados por el conjunto de reglas de inicial.

El modelo que proponen en Cocarascu et al., (2018) genera programas lógicos, que capturan el razonamiento desde un conjunto de ejemplos S . Estas reglas permiten hacer exactamente las mismas predicciones para nuevos ejemplos que un predictor basado en un *autoencoder* + ANN (Artificial Neural Net). Al mismo tiempo, estas reglas resumen la estructura del conjunto de ejemplos S . En particular, describen de manera concisa en términos lógicos cuál debería ser la predicción y cuáles características deberían influir en ella. Las reglas son genéricas para S , pero cuando se hace una predicción para un nuevo ejemplo, sus características se agregan como hechos a las reglas.

En Ramirez et al., (2019) se utiliza una unidad de módulo básico para realizar la clasificación de señales dirigida. En la arquitectura resultante, se define la integración de varios módulos, donde cada unidad de módulo básico se compone de tres clasificadores diferentes basados en los siguientes modelos: algoritmo KNN (*K-Nearest Neighbor*) difuso, perceptrón multicapa con descenso de gradiente y momento (MLP-GDM) y perceptrón multicapa con retropropagación de gradiente conjugada escalada (MLP-SCG). Las salidas de los clasificadores se combinan mediante un sistema difuso para la integración de los resultados. Diseñaron dos sistemas de lógica difusa, el sistema de inferencia difuso Mamdani tipo 1 (FIS tipo 1) y un sistema difuso de intervalo tipo 2 (IT2FIS). El objetivo es realizar una comparación entre FIS tipo 1 e IT2FIS en el modelo híbrido. Los conjuntos difusos tipo 2 se usan para modelar incertidumbre e imprecisión, y son esencialmente conjuntos difusos en los que los grados de pertenencia son conjuntos difusos de tipo 1. La utilización de sistemas difusos tipo 2 se justifica cuando existe un alto nivel de incertidumbre, en casos donde se busca encontrar mejores resultados que la lógica difusa tradicional.

La diferencia fundamental entre la lógica difusa tipo 1 y tipo 2 está en su definición: Un conjunto difuso A en U puede ser representado como un conjunto de pares ordenados de elementos, y su valor de pertenencia está dado por:

$$A = \{x, \mu_A(x) \mid x \in U\}$$

Donde U es el universo del discurso continuo (por ejemplo, todos los números reales de $U = \mathbb{R}$). Un conjunto difuso tipo 2 es denotado por \tilde{A} y es caracterizado por una función de pertenencia:

$$\mu_{\bar{A}}(x,u) \text{ donde } x \in X \text{ y } u \in \int_x^u \subseteq [0,1] \text{ y } 0 \leq \mu_{\bar{A}}(x,u) \leq 1, \text{ es definido mediante la ecuación:}$$

$$\bar{A} = \{(x,u, \mu_{\bar{A}}(x,u)) / x \in X, \forall u \in \int_x \subseteq [0,1]\}$$

Entonces se puede afirmar que existe incertidumbre en la lógica difusa T2 tanto en la variable primaria como en la secundaria, mientras que en la lógica T1 no existe incertidumbre en la variable primaria, Díaz et al., (2009).

En Wang & Pan, (2020), los autores utilizan reglas lógicas para hacer cumplir correlaciones complejas en el espacio de salida, e integrar estas reglas en el sistema de representación distribuida de aprendizaje con un mecanismo de aprendizaje para lograr inferencia conjunta. El módulo lógico se compone de un conjunto de reglas lógicas representadas por LPO. Estas reglas especifican explícitamente las relaciones complejas en el espacio de etiquetas de salida, que no se pueden manejar con restricciones simples. Estas dependencias pueden considerarse restricciones complejas que pueden expresarse de manera eficiente como reglas lógicas.

En Riegel et al., (2020), los autores proponen un *framework* novedoso que proporciona propiedades clave tanto de las redes neuronales (aprendizaje) como de la lógica simbólica (conocimiento y razonamiento). Cada neurona tiene un significado como componente de una fórmula en una lógica ponderada de valor real, produciendo una representación desenredada altamente interpretable. La inferencia es más bien omnidireccional, que se centra en variables objetivo predefinidas, y corresponde al razonamiento lógico, incluido el teorema de lógica clásica de primer orden que se demuestra como un caso especial.

En Barceló et al., (2020) se plantea la utilización de Lógica FOC₂ (Lógica de predicados de primer orden con cuantificadores de conteo), un fragmento bien estudiado de lógica de primer orden, el cual está estrechamente relacionado con la prueba Weisfeiler-Lehman (WL) y, por lo tanto, con los GNN.

La idea principal detrás de las GNN (*Graph Neural Network*) es que las conexiones entre neuronas no son arbitrarias, sino que reflejan la estructura de los datos de entrada. Este enfoque está motivado por redes neuronales convolucionales y recurrentes y generaliza ambas.

En este artículo los autores dan un paso hacia la comprensión del poder expresivo, al establecer conexiones entre GNN y formalismos lógicos conocidos. Se sugiere además que, estas conexiones son conceptualmente importantes, ya que permiten comprender los aspectos de comportamiento inherentemente procedimentales de algunos fragmentos de GNN en términos del aspecto más declarativo de los lenguajes lógicos.

La arquitectura AC-GNN se amplía de una manera muy simple al permitir lecturas globales, donde en cada capa también se calcula un vector de características para todo el grafo, y lo combinan con agregaciones locales; A estos los definen como *GNN de lectura combinada agregada* (ACR-GNN: *Aggregate-Combine-Readout, Graph Neural Network*).

En este trabajo demuestran que cada fórmula de FOC-2 puede ser capturada por un ACR-GNN.

Este estudio relaciona el poder de los GNN con el de los clasificadores expresados en la lógica de predicados de primer orden (FO), sobre grafos (no dirigidos) donde cada vértice tiene características únicas.

En Q. Li et al., (2021), los autores adoptan programas funcionales para entender el significado semántico de los conceptos, por lo que ven a la semántica del aprendizaje como una inducción del programa. La semántica de un concepto es tratada como una función.

3.2.3 Mejorar los datos de entrenamiento

En el trabajo realizado por Alzaeemi et al., (2019), los autores buscan encontrar los valores de verdad de los valores de entrada de una red neuronal de función de base radial (RBFNN - *Radial Basis*

Function Neural Network) para las cláusulas, a través de programación lógica con 2SAT (2-satisfiability)¹⁰.

En de Penning et al., (2014) logran aprovechar la integración neuronal-simbólica, utilizando las redes para realizar un aprendizaje y adaptación robusta, y la extracción de conocimiento simbólico para representar las *relaciones temporales* explícitamente y por razonamiento cualitativo.

La solución propuesta en T. Li & Srikumar, (2019) compila sistemáticamente declaraciones lógicas en grafos de cálculo que aumentan una red neuronal sin parámetros extra que aprender, o rediseño manual. Los autores sostienen que pueden *combatir la sed de datos de las redes neuronales* aprovechando el conocimiento del dominio expresado como lógica de primer orden. En tal sentido, las redes neuronales son provistas del conocimiento del dominio, el cual queda codificado y expresado como reglas lógicas de primer orden (LPO).

En Diligenti et al., (2017), los autores sostienen que la regularización basada en semántica (SBR: *Semantic Based Regularization*) se utiliza como marco subyacente para representar el conocimiento previo, expresado como una colección de cláusulas LPO, y donde cada tarea a aprender corresponde a un predicado en la base de conocimiento. Dicha base correlaciona las tareas que se deben aprender y se traduce en un conjunto de restricciones que se integran en el proceso de aprendizaje a través de *backpropagation*.

La regularización semántica es una estadística relacional de aprendizaje, que integra la capacidad de aprender de ejemplos y reglas lógicas. El conocimiento previo en SBR se expresa a través de un conjunto de cláusulas LPO.

El aprendizaje relacional estadístico es particularmente adecuado para inyectar conocimiento lógico en el aprendizaje, porque transforma el conocimiento en un conjunto de restricciones continuas que se pueden integrar en las funciones de costos que generalmente se consideran en el aprendizaje automático.

En el trabajo desarrollado por Han et al., (2021), los autores llevan a cabo razonamiento lógico simbólico para hacer un análisis *causa-efecto* no supervisado, de entidades detectadas con anomalías a través del aprendizaje meta-interpretativo. Utilizan LPO para embeber conocimiento previo, y sistemas de razonamiento dentro de la red de grafos.

Finalmente, el enfoque que presenta el trabajo en Xie et al., (2021), se trata de embeber conocimiento simbólico en una DNN, expresado como lógica temporal lineal (LTL), y usar esta información para guiar el entrenamiento del modelo. Específicamente, construyen incrustaciones semánticas de autómatas generados a partir de la fórmula LTL a través de una red neuronal gráfica.

3.2.4 Acelerar el aprendizaje

En el trabajo presentado por Borges et al., (2006), los autores proponen una arquitectura que combina redes neuronales recurrentes de entrada-salida y un sistema neural-simbólico para el aprendizaje de conocimiento *simbólico temporal*. Mediante un enfoque experimental, han demostrado que la integración del conocimiento simbólico puede mejorar las características de una red neuronal, en particular la tolerancia al ruido, rendimiento y convergencia de un algoritmo de aprendizaje.

En Marra et al., (2019), los autores sugieren que los predicados y funciones del conocimiento de LPO se pueden vincular a cualquier grafo computacional de *TensorFlow* (TF), y las fórmulas se convierten en un conjunto de restricciones de valor real, que participan en el problema de optimización general. Esto permite conocer el peso de los *learners*, bajo las limitaciones impuestas por el

¹⁰ Problema computacional que asigna valores a variables, cada una de las cuales tiene dos valores posibles, para satisfacer un sistema de restricciones sobre pares de variables. Puede involucrar restricciones en más de dos variables, y de problemas de satisfacción de restricciones, que pueden permitir más de dos opciones para el valor de cada variable.

conocimiento previo. El marco es extremadamente general, ya que no impone restricciones en cuanto a qué modelos o conocimientos se pueden integrar.

Este documento presenta LYRICS, un entorno de TF basado en un lenguaje declarativo para la integración de conocimientos previos en el aprendizaje automático, que permite la plena expresividad de LPO para definir conocimiento.

LYRICS tiene su raíz en marcos como la Regularización basada en semántica (SBR) construida sobre Kernel Machines y Logic Tensor Networks (LTN) que se pueden aplicar a redes neuronales. Estos marcos transforman las cláusulas LPO en un conjunto de restricciones que se optimizan conjuntamente durante el aprendizaje. En particular, cualquier teoría lógica de primer orden con muchas clasificaciones se puede expresar en el *framework*, lo cual permite declarar dominios de diferente tipo, con constantes, predicados y funciones, proporcionando una integración muy estrecha de aprendizaje y lógica, ya que cualquier grafo computacional puede vincularse a un predicado LPO. Esto permite limitar al *learner* tanto durante el entrenamiento como durante la inferencia. Dado que el *framework* es agnóstico para los *learners* que están vinculados a los predicados, se puede utilizar en una amplia gama de aplicaciones, incluida la clasificación, el aprendizaje automático generativo o adversario, el aprendizaje secuencia a secuencia, la clasificación colectiva, etc. Las funciones LPO permiten el mapeo entre individuos de los dominios de entrada a un individuo del dominio de salida.

En el trabajo desarrollado en Cai et al., (2017), se propone un método de razonamiento simbólico basado en DNN, y este método se aplica al descubrimiento de axiomas, el cual hace uso del concepto de manipulación simbólica. Específicamente, se basa en la capacidad de aprendizaje de las DNN y la capacidad de razonamiento de un sistema lógico: el sistema lógico genera ejemplos de entrenamiento, que indican cómo manipular símbolos a partir de una serie dada de datos, y luego las DNN intentan aprender tales ejemplos, puntuarlos y realizar abstracciones de posibles axiomas. En particular, este método permite a las redes neuronales profundas realizar un razonamiento simple con variables en la lógica de predicados.

Las redes neuronales de alimentación profunda pueden colaborar con el sistema lógico y aprender a manipular símbolos. En particular, son capaces de terminar algunas tareas de razonamiento con variables. Esta capacidad se puede utilizar para aprender reglas lógicas y se puede utilizar más para descubrir axiomas.

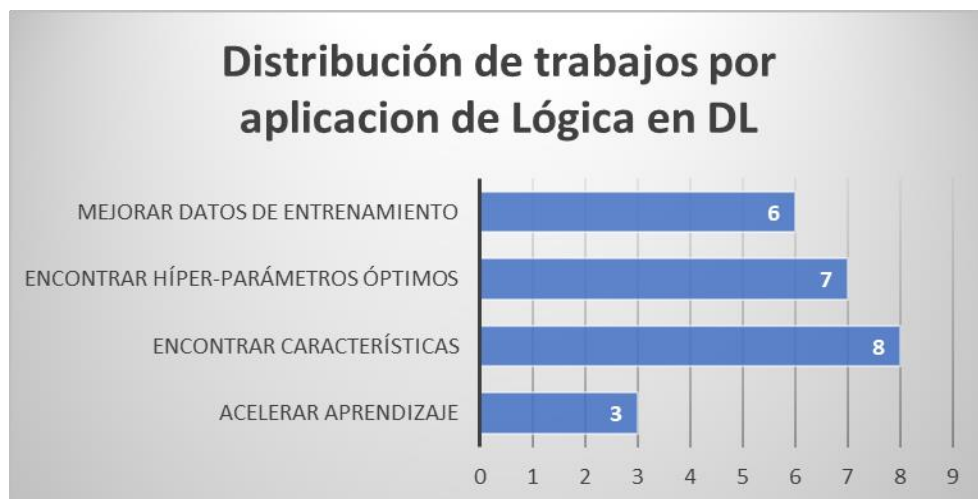


Figura 4. Aplicación de Lógica en el proceso de DL

3.3 PI3. ¿Qué herramientas de la lógica se aplican? ¿Qué tipos de tareas se abordan? ¿Sobre cuáles dominios?

Esta pregunta de investigación fue abordada en base a los diferentes tipos de lógica utilizada en cada trabajo.

3.3.1 Programación Lógica

En el trabajo presentado en Alzaeemi et al., (2019), los autores presentan una solución de aplicación general, a través del uso de programación lógica tipo 2SAT, donde se busca la optimización de la red (parámetros y neuronas en la capa oculta).

En Manhaeve et al., (2018), los autores utilizan programación lógica para resolver tareas de interpretación y clasificación de imágenes.

Por su parte, en el trabajo realizado por Cocarascu et al., (2018), los autores utilizan programación lógica para tareas de clasificación, en una solución de aplicación general.

También se utiliza programación lógica en el trabajo realizado en Mahdavifar & Ghorbani, (2020), aplicada dentro de un *sistema experto*, para tareas de clasificación en la detección de *cyber-ataques*.

3.3.2 Lógica Difusa

En Melin et al., (2018), los autores utilizan lógica difusa (*Mamdani Fuzzy Inference*), y para optimizar el sistema de lógica difusa utilizaron un algoritmo genético. En este trabajo aplicado a Medicina, se busca hacer mediciones de presión sanguínea para hacer predicciones de riesgos de Hipertensión.

En Pal & Kar, (2019), los autores plantean una solución de aplicación general para realizar predicciones en el cual utilizan lógica difusa de tipo 2.

El trabajo realizado por AmirHosseini & Hosseini, (2019), los autores utilizan lógica difusa aumentada con algoritmos evolutivos, para tareas de clasificación. En este trabajo de aplicación médica, proveen una solución para clasificación de tumores en imágenes de tomografía computarizada del hígado.

En Ramirez et al., (2019) también utilizan lógica difusa del tipo 1 y 2. Esta solución de aplicación médica busca hacer clasificaciones de arritmias cardiacas.

3.3.3 Lógica Temporal

En de Penning et al., (2014), los autores utilizan lógica temporal aumentada con Redes bayesianas. La arquitectura descrita utiliza la lógica temporal como teoría T y una máquina de Boltzmann restringida (RBM - *Restricted Boltzmann Machines*) como red neuronal N. mediante este trabajo buscan predecir reducción de emisiones de CO2 utilizando Agentes en Sistemas de transporte inteligentes.

3.3.4 Lógica de Primer Orden (LPO)

En T. Li & Srikumar, (2019) se plantea una solución de Aplicación general para realizar predicciones, utilizando Lógica de primer orden.

En Sinha et al., (2020) para tareas de predicción, se utiliza lógica de primer orden para evaluar la generalización lógica en una GNN.

En el trabajo presentado por Diligenti et al., (2017) para tareas de clasificación, se utiliza lógica de primer orden, para la clasificación de Imágenes.

En Riegel et al., (2020) plantean una solución con la capacidad general de resolución de problemas de un demostrador de teoremas completo, utilizando lógica de primer orden para tareas de predicción.

Para realizar extracción de información de textos, Wang & Pan, (2020) utilizan una combinación de lógica de primer orden con lógica probabilística para clasificación.

En Han et al., (2021), los autores proponen una solución de aplicación médica, para la identificación de discos en imagen de columnas. Para este trabajo de clasificación utilizan una combinación entre programación lógica de primer orden con aprendizaje meta-interpretativo.

En Marra et al., (2019) se plantea una solución de Aplicación general para realizar clasificaciones, utilizando Lógica de primer orden aumentado con Redes Bayesianas.

En Barceló et al., (2020) para tareas de clasificación, utilizan *LPO – FOC-2 (Lógica de primer orden con 2 variables) & Counting*, para aportar una solución de aplicación general. Si bien el trabajo está orientado a distinguir colores en nodos de un grafo, se han hecho pruebas también sobre una base de datos para identificación de proteínas, lo cual plantea una solución de aplicación real muy interesante.

3.3.5 Lógica continua nilpotente

También para la solución de aplicación general, pero para tareas de predicción, en Csiszár et al., (2020) utilizan lógica difusa nilpotente aumentada con lógica continua a través de la toma decisiones multicriterio.

En Dombi & Csiszár, (2021) utilizan lógica continua nilpotente combinada con herramientas de decisión multicriterio, para la solución de aplicación general.

3.3.6 Programación Lógica Inductiva

Para realizar predicción de expresiones matemáticas, Q. Li et al., (2021) utilizan programación lógica inductiva, aumentada con redes Bayesianas.

Otra solución de aplicación medica propuesta por Schmid & Finzel, (2020), para tareas de clasificación, utilizan programación lógica inductiva aumentada con conocimiento del dominio. El objetivo de este trabajo es la detección de tumores a través de un modelo de aprendizaje e Inferencia, partiendo de una base de datos de imágenes.

3.3.7 Lógica Temporal Lineal

Para aprender actividades de reconocimiento e imitación de secuencias y series temporales aplicadas a robótica, Xie et al., (2021) aplican lógica temporal lineal (LTL), para tareas de Predicción.

3.3.8 Lógica Proposicional

Borges et al., (2006) presentan un *framework* de aplicación general para realizar predicciones, utilizando lógica proposicional.

3.3.9 Lógica de predicados

En Cai et al., (2017) para realizar predicciones de expresiones matemáticas lógica de predicados, en particular para realizar el descubrimiento de axiomas.

A modo de resumen para la pregunta 3, se presenta en la Fig. 5 la distribución de trabajos encuadrados según el tipo de lógica utilizada. En la Fig. 6 se presenta la distribución de trabajos según las tareas de predicción o clasificación. Por último, en la Fig. 7 se presenta del mismo modo la distribución de trabajos por dominios de aplicación.



Figura 5. Distribución de trabajos por tipo de lógica utilizada.



Figura 6. Distribución de trabajos por objetivo.

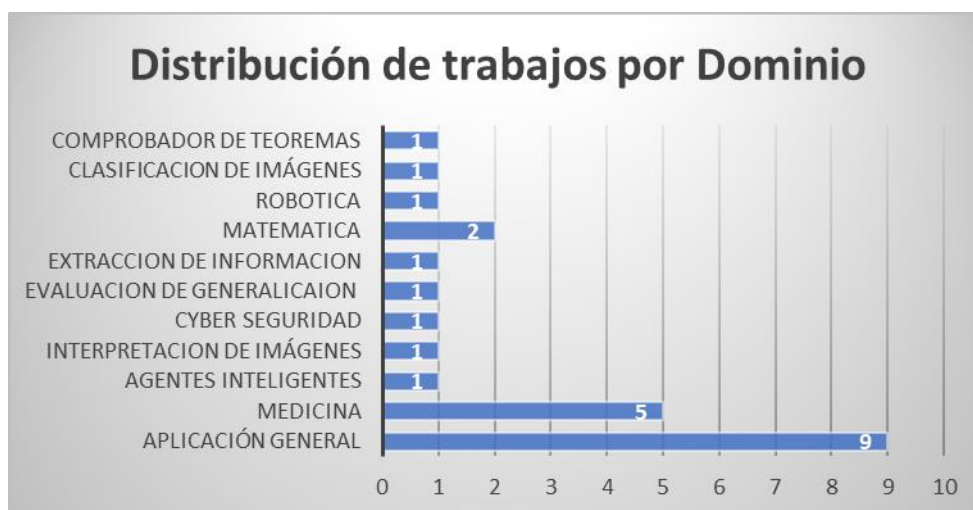


Figura 7. Distribución de trabajos por dominio de aplicación.

4 Limitaciones del estudio

En la siguiente sección se presentan las limitaciones del estudio, y se introduce una indicación sobre el trabajo a futuro.

Al aplicar las cadenas de búsqueda, los motores de búsqueda han listado artículos con contenidos relevantes para la presente revisión, pero estos no estaban disponibles para una lectura y análisis en profundidad, los cuales debieron ser excluidos.

Dada su naturaleza de no ser difundida de forma habitual mediante la publicación, sino a través de canales limitados y de difícil acceso a ellas, los artículos encuadrados como bibliografía gris pueden haber quedado fuera del alcance de la revisión al no ser encontrados por los motores de búsqueda.

Presencia de publicación segmentada, la cual es una forma distinta de publicación redundante que generalmente se caracteriza por la similitud de hipótesis, metodología o resultados, pero no por la similitud del texto. Estos aspectos de las publicaciones no son detectados objetivamente por las aplicaciones de *software* y, por lo tanto, representan una seria amenaza para el análisis de la revisión.

Como idea de trabajo a futuro, se ha comenzado a desarrollar una herramienta de búsqueda y análisis semántico de artículos, a través de la cual, con una clasificación inicial mínima entre artículos relevantes e irrelevantes sobre una temática dada, y el uso de un clasificador del tipo *few shots learning*, la herramienta sea capaz de identificar artículos relevantes para el análisis dentro de un resultado de búsqueda mucho mayor.

5 Discusión

Se han analizado 24 artículos seleccionados que abordan temas de IA no-simbólica aumentada con IA simbólica sobre ecosistemas de *software*, en diferentes dominios de aplicación.

Es muy interesante observar como las soluciones resultantes se vuelven una herramienta de asistencia que, en algunos casos, se alimenta e interactúa con especialistas del dominio logrando mejorar a partir de esta interacción. Es importante destacar que los artículos analizados en la presente revisión, presentan combinaciones de técnicas que propenden a mejorar los rendimientos de los sistemas resultantes.

Haciendo un análisis de las tendencias de elecciones tecnológicas, observado desde el punto de vista de las falencias de la IA no-simbólica abordadas en la pregunta P11, se desprende el siguiente análisis:

Al abordar el problema de **Falta de explicabilidad** (25% de los casos), la preferencia de los investigadores es la utilización de *Programación Lógica* (50% de los casos), normalmente aumentada con sistemas expertos o utilización de conocimiento del dominio. La tendencia sigue con la utilización *Lógica continua nilpotente* apoyada por herramientas de decisión multicriterio (33% de los casos), y en menor medida la utilización de *Lógica difusa* aumentada con *Algoritmos Evolutivos* (17% de los casos). La tendencia respecto a los objetivos, es hacia tareas de *clasificación* (83%), mientras que solo el 17% hace foco en tareas de *predicción*. Finalmente, para esta deficiencia, el análisis muestra que el 83% de los estudios, busca *encontrar los hiper-parámetros óptimos*, mientras que solo el 17% se enfoca en encontrar las características. Todos los trabajos que utilizan Programación Lógica y Lógica Difusa, se enfocan en tareas de Clasificación; Mientras que para los trabajos que implementan *Lógica continua nilpotente*, la tendencia es 50% predicción y 50% clasificación.

Al abordar el problema de **Falta de generalización** (25% de los casos), la tendencia es más pareja respecto a la elección tecnológica dado que, el 50% de los casos aplica LPO, el 33% utiliza *Programación Lógica* (dentro de este grupo un solo trabajo utilizo *Programación Lógica Inductiva*) y el 17% utilizo *Lógica Difusa* aumentada con *Lógica difusa tipo 2*. La tendencia respecto a los objetivos, también tiene una distribución homogénea, dado que el 50% de los trabajos dirigió sus esfuerzos hacia tareas de *clasificación*, mientras que el otro 50% lo hizo orientado a tareas de *predicción*. De los trabajos que utilizaron LPO como herramienta base, el 67% enfocó su análisis en tareas de *clasificación*, mientras que el 23% restante lo enfocó hacia *predicción*.

Se encontró una alta correlación entre los trabajos que utilizan LPO los cuales se centran en *encontrar las características del modelo* en su totalidad; mientras que el 50% de los trabajos que aplican *Programación Lógica* también se concentran en *encontrar las características del modelo*, el 50% restante se enfoca en mejorar los datos de entrenamiento. Finalmente, en el trabajo que se utilizó Lógica Difusa, los autores se enfocaron en tareas de clasificación con la finalidad de encontrar las características del modelo.

Al abordar el problema de **Falta de razonamiento** (13% de los casos), no se distingue una distribución homogénea. En el trabajo que utilizó LPO como herramienta base, se aumentó el modelo con *Aprendizaje Meta Interpretativo*, para resolver tareas de *clasificación* donde proponen *mejorar los datos de entrenamiento*. El trabajo que implementó *Programación Lógica*, se enfocó más en encontrar los *hiper-parámetros óptimos del modelo* también para tareas de *clasificación*, mientras que el trabajo que utilizó *Lógica de Predicados* como herramienta de razonamiento, se enfocó en la *aceleración del aprendizaje* en tareas de *predicción*. Esta es quizás el área donde menos trabajos se han realizado.

Al abordar el problema de **Ineficiencia de los datos** (37% de los casos), encontramos que el 44% de los trabajos utilizaron LPO como herramienta base de razonamiento, y solo un caso aumento el modelo combinando LPO con *Redes Bayesianas*. La distribución respecto al objetivo es homogénea, dado que el 50% enfocó sus esfuerzos en tareas de *clasificación*, mientras que el otro 50% lo dirigió a tareas de *predicción*.

En relación a la aplicación de LPO, el 50% de los trabajos se enfocó en mejorar los datos de entrenamiento, el otro 25% en *encontrar las características*, y el 25% restante en *acelerar el aprendizaje*.

El 22% de los autores utilizaron *lógica difusa* combinada con *algoritmos genéticos* y *lógica difusa* de tipo 2. El 50% de los trabajos se enfocó en *encontrar las características del modelo*, mientras que el otro 50% se centró en *encontrar los hiper-parámetros óptimos*. Aquí hay una fuerte correlación con tareas de *predicción*, dado que todos los trabajos se concentraron en ello.

También con un 22% de adopción, la tendencia se mantiene con el uso de *Lógica Temporal* (uno de los trabajos aplico *Lógica temporal lineal*). Solo un trabajo aumento el modelo lógico utilizando además *Redes Bayesianas*. Algo a destacar en este segmento es que la correlación hacia el objetivo y la aplicación son del 100% para tareas de *predicción* y *mejora de los datos de entrenamiento*.

Para finalizar solo un trabajo (11%), hizo uso de *Lógica Proposicional*, para tareas de *predicción* aplicado a *acelerar el aprendizaje*.

Este trabajo proporciona una base para identificar brechas y oportunidades de investigación en el área de la IA, más precisamente en el mundo de DL. Proporciona un estado global del arte a través de estudios validados.

Finalmente, el análisis de los diferentes estudios sienta las bases para realizar una revisión sistemática de la literatura centrada en soluciones en el campo de la IA aplicada a los ecosistemas de *software*, con especial énfasis en los ecosistemas tecnológicos.

6 Conclusiones

Dada la popularidad reciente que ha obtenido la IA y el papel potencial que podrían desempeñar las tecnologías basadas en lógica en la ingeniería de sistemas inteligentes complejos, se desprende que la combinación de técnicas, dentro de IA, basadas en diferentes tipos de lógicas que se combinen y potencien las capacidades del DL, serían el camino a seguir.

Las tecnologías resultantes de esta revisión sistemática, se analizan y evalúan desde dos perspectivas diferentes – a saber, *Deep Learning* y Lógica.

A través del análisis de los trabajos presentados a lo largo de este estudio, se han visto combinaciones de lo más variadas entre sistemas lógicos, con alguna forma de red neuronal (DNN, GNN, CNN).

Esta investigación logró clasificar, organizar y explicar las diferentes formas en que las deficiencias del DL son abordadas por propuesta basadas en lógica simbólica. El estudio también determinó en qué etapas del proceso de DL dichas propuestas son aplicadas. Y complementariamente, el estudio permitió determinar cuáles son las herramientas de la lógica que se aplican preferentemente, para cada área y cada dominio.

Si bien no se observa un patrón arquitectónico claro, los esfuerzos por encontrar un *modelo de propósito general* que combine ambos mundos dirigen las tendencias y esfuerzos de investigación.

En cuanto a los ecosistemas tecnológicos, si bien este concepto se enmarca en el campo de los ecosistemas de la ingeniería de *software*, presenta diferentes matices que necesitan soluciones enfocadas a la evolución del dominio, y la inclusión del factor humano al mismo nivel que el componente de *software*.

Finalmente, debido al enfoque científico, bien fundamentado y reproducible de esta revisión, se deduce que representa una herramienta confiable para evaluar el estado actual del arte, con la esperanza que pueda usarse para comprender las direcciones futuras en esta área.

Anexo I - Acrónimos y siglas

IA	= Inteligencia Artificial
MI	= Motor de Inferencia
LPO	= Lógica de primer orden
PI	= Pregunta de Investigación
PC	= Palabra Clave
TF	= TensorFlow (Librería para desarrollo de redes neuronales)
BD	= Base de Datos
PNL	= Procesamiento Natural del Lenguaje
LPPO	= Lógica de predicados de primer orden
FOC ₂	= Lógica de predicados de primer orden con cuantificadores de conteo
TDMC	= Toma de decisión de múltiple criterio
ILP	= Inductive Logic Programming (Programación lógica inductiva)
FIS	= Fuzzy Inference System (Sistema de inferencia difuso)
IT2FIS	= Sistema de inferencia difuso de intervalo de tipo 2
LTL	= Lógica Temporal Lineal
DL	= Deep Learning
ANN	= Artificial Neural Net
NN	= Neural Network
DNN	= Deep Neural Network
GNN	= Graph Neural Network
RN	= Relational Network
RBM	= Restricted Boltzmann Machines
FIS	= Fuzzy Inference System
AC-GNN	= Aggregate-Combine, Graph Neural Network (Agregar-Combinar, Red Neuronal de Grafos)
ACR-GNN	= Aggregate-Combine-Readout, Graph Neural Network (Agregar-Combinar-Lectura, Red Neuronal de Grafos)
SBR	= Semantic Based Regularization (Regularización basada en semántica)
SLR	= Systematic Literature Review (Revisión Sistemática de la Literatura)

Referencias

- Alzaeemi, S. A., Mansor, M. A., Kasihmuddin, M. S. M., & Sathasivam, S. (2019). 2 satisfiability logic programming in radial basis function neural networks. *AIP Conference Proceedings*, 2184(1), 060045.
- AmirHosseini, B., & Hosseini, R. (2019). An improved fuzzy-differential evolution approach applied to classification of tumors in liver CT scan images. *Medical & Biological Engineering & Computing*, 57(10), 2277-2287. <https://doi.org/10.1007/s11517-019-02009-7>
- Barceló, P., Kostylev, E. V., Monet, M., Pérez, J., Reutter, J., & Silva, J. P. (2020). The logical expressiveness of graph neural networks. The logical expressiveness of graph neural networks, hal-03356968f. <https://hal.archives-ouvertes.fr/hal-03356968>
- Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58, 82-115.
- Battaglia, P., Pascanu, R., Lai, M., Rezende, D., & Kavukcuoglu, K. (2016). Interaction networks for learning about objects, relations and physics. *Advances in Neural Information Processing Systems*.
- Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Gulcehre, C., Song, F., Ballard, A., Gilmer, J., Dahl, G., Vaswani, A., Allen, K., Nash, C., Langston, V., ... Pascanu, R. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv:1806.01261 [cs, stat]*. <http://arxiv.org/abs/1806.01261>
- Borges, R. V., Lamb, L. C., & D'Avila Garcez, A. S. (2006). Combining Architectures for Temporal Learning in Neural-Symbolic Systems. *2006 Sixth International Conference on Hybrid Intelligent Systems (HIS'06)*, 46-46.
- Cai, C.-H., Ke, D., Xu, Y., & Su, K. (2017). Symbolic manipulation based on deep neural networks and its application to axiom discovery. *2017 International Joint Conference on Neural Networks (IJCNN)*, 2136-2143.
- Calegari, R., Ciatto, G., Mascardi, V., & Omicini, A. (2020). Logic-based technologies for multi-agent systems: A systematic literature review. *Autonomous Agents and Multi-Agent Systems*, 35(1), 1.
- Cocarascu, O., Cyras, K., & Toni, F. (2018). *Explanatory predictions with artificial neural networks and argumentation*.
- Csiszár, O., Csiszár, G., & Dombi, J. (2020). How to implement MCDM tools and continuous logic into neural computation?: Towards better interpretability of neural networks. *Knowledge-Based Systems*, 210, 106530.
- de Penning, L., d'Avila Garcez, A. S., Lamb, L. C., Stuiver, A., & Meyer, J.-J. Ch. (2014). Applying Neural-Symbolic Cognitive Agents in Intelligent Transport Systems to reduce CO2 emissions. *2014 International Joint Conference on Neural Networks (IJCNN)*, 55-62.
- Díaz, G. A., Sotelo, J. A. L., & Bravo, E. C. (2009). *Aplicación de la lógica difusa tipo dos en una planta didáctica en control de procesos industriales, respecto de las variables nivel y flujo*. 7(13), 21.
-

- Diligenti, M., Roychowdhury, S., & Gori, M. (2017). Integrating prior knowledge into deep learning. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 920-923.
- Dombi, J., & Csiszár, O. (2021). Interpretable Neural Networks Based on Continuous-Valued Logic and Multi-criteria Decision Operators. En J. Dombi & O. Csiszár (Eds.), *Explainable Neural Networks Based on Fuzzy Logic and Multi-criteria Decision Tools* (pp. 147-169). Springer International Publishing.
- Domingos, P. (2018). How the Quest for the Ultimate Learning Machine will remake our World. En *The Master Algorithm How*.
- Evans, R., Saxton, D., Amos, D., Kohli, P., & Grefenstette, E. (2018). Can neural networks understand logical entailment? En *ArXiv*. <https://arxiv.org/abs/1802.08535>
- Garnelo, M., Arulkumaran, K., & Shanahan, M. (2016). Towards Deep Symbolic Reinforcement Learning. *arXiv:1609.05518 [cs]*. <http://arxiv.org/abs/1609.05518>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. The MIT Press.
- Han, Z., Wei, B., Xi, X., Chen, B., Yin, Y., & Li, S. (2021). Unifying neural learning and symbolic reasoning for spinal medical report generation. *Medical Image Analysis*, 67, 101872.
- Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, 51(1), 7-15.
- Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*.
- Li, Q., Huang, S., Hong, Y., Zhu, Y., Wu, Y. N., & Zhu, S.-C. (2021). A HINT from Arithmetic: On Systematic Generalization of Perception, Syntax, and Semantics. *arXiv preprint arXiv:2103.01403*.
- Li, T., & Srikumar, V. (2019). Augmenting neural networks with first-order logic. *arXiv preprint arXiv:1906.06298*.
- MahdaviFar, S., & Ghorbani, A. A. (2020). DeNNeS: Deep embedded neural network expert system for detecting cyber attacks. *Neural Computing and Applications*, 32(18), 14753-14780.
- Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., & De Raedt, L. (2018). Deepproblog: Neural probabilistic logic programming. *Advances in Neural Information Processing Systems*, 31, 3749-3759.
- Marcus, G. (2018). Deep Learning: A Critical Appraisal. *arXiv:1801.00631 [cs, stat]*. <http://arxiv.org/abs/1801.00631>
- Marra, G., Giannini, F., Diligenti, M., & Gori, M. (2019). Lyrics: A general interface layer to integrate logic inference and deep learning. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*, 283-298.
- McCarthy, J. (1987). Generality in artificial intelligence. *Communications of the ACM*.
- Melin, P., Miramontes, I., & Prado-Arechiga, G. (2018). A hybrid model based on modular neural networks and fuzzy systems for classification of blood pressure and hypertension risk diagnosis. *Expert Systems with Applications*, 107, 146-164.
- Metakides, G., & Nerode, A. (1996). *Principles of Logic and Logic Programming*. Elsevier.
-

- Moher, D., Liberati, A., Tetzlaff, J., & Altman, D. G. (2010). Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. *Int J Surg*, 8(5), 336-341.
- Pal, S. S., & Kar, S. (2019). A Hybridized Forecasting Method Based on Weight Adjustment of Neural Network Using Generalized Type-2 Fuzzy Set. *International Journal of Fuzzy Systems*, 21(1), 308-320.
- Petticrew, M., & Roberts, H. (2008). *Systematic Reviews in the Social Sciences: A Practical Guide*. John Wiley & Sons.
- Ramirez, E., Melin, P., & Prado-Arechiga, G. (2019). Hybrid model based on neural networks, type-1 and type-2 fuzzy systems for 2-lead cardiac arrhythmia classification. *Expert Systems with Applications*, 126, 295-307.
- Riegel, R., Gray, A., Luus, F., Khan, N., Makondo, N., Akhalwaya, I. Y., Qian, H., Fagin, R., Barahona, F., & Sharma, U. (2020). Logical neural networks. *arXiv preprint arXiv:2006.13155*.
- Rijk, L. M. de. (2002). *Aristotle: Semantics and Ontology: Volume I: General Introduction. The Works on Logic*. BRILL.
- Rocktäschel, T., & Riedel, S. (2017). End-to-End Differentiable Proving. *arXiv:1705.11040 [cs]*. <http://arxiv.org/abs/1705.11040>
- Russell, S., & Norvig, P. (2010). *Artificial Intelligence A Modern Approach Third Edition*. En *Pearson*.
- Santoro, A., Raposo, D., Barrett, D. G. T., Malinowski, M., Pascanu, R., Battaglia, P., & Lillicrap, T. (2017). A simple neural network module for relational reasoning. *Advances in Neural Information Processing Systems*.
- Schmid, U., & Finzel, B. (2020). Mutual Explanations for Cooperative Decision Making in Medicine. *KI - Künstliche Intelligenz*, 34(2), 227-233.
- Schmidhuber, J. (2015). Deep Learning in neural networks: An overview. En *Neural Networks*.
- Sinha, K., Sodhani, S., Pineau, J., & Hamilton, W. L. (2020). Evaluating logical generalization in graph neural networks. *arXiv preprint arXiv:2003.06560*.
- Tsividis, P. A., Pouncy, T., Xu, J. L., Tenenbaum, J. B., & Gershman, S. J. (2017). Human learning in atari. *AAAI Spring Symposium - Technical Report*.
- Wang, W., & Pan, S. J. (2020). Integrating deep learning with logic fusion for information extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05), 9225-9232.
- Xie, Y., Zhou, F., & Soh, H. (2021). Embedding Symbolic Temporal Knowledge into Deep Sequential Models. *arXiv preprint arXiv:2101.11981*.
- Yann LeCun, Yoshua Bengio, G. H. (2015). Deep learning (2015), Y. LeCun, Y. Bengio and G. Hinton. *Nature*.
-