



UNIVERSIDAD
NACIONAL
DE LOJA



Facultad de Energía, las Industrias y los Recursos Naturales No Renovables

Carrera de Ingeniería en Sistemas

“DETECCIÓN DE TÉCNICAS DE APRENDIZAJE PROFUNDO APLICADAS EN LAS DIFERENTES ÁREAS DEL CONOCIMIENTO, EMPLEANDO EL MÉTODO DE REVISIÓN SISTEMÁTICA DE LITERATURA.”

TESIS DE GRADO PREVIA A LA
OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS

Autor:

Suntaxi Sarango Martha Cristina

Director:

Ing. Pablo Fernando Ordoñez Ordoñez, Mg.Sc.

LOJA – ECUADOR
2019

Certificación

Ing. Pablo Fernando Ordoñez Ordoñez Mg. Sc.

Director de Tesis.

Certifico:

Que el presente trabajo previo a la obtención del título de Ingeniera en Sistemas titulado:
“DETECCIÓN DE TÉCNICAS DE APRENDIZAJE PROFUNDO APLICADAS EN LAS DIFERENTES ÁREAS DEL CONOCIMIENTO, EMPLEANDO EL MÉTODO DE REVISIÓN SISTEMÁTICA DE LITERATURA” de autoría de la egresada **Martha Cristina Suntaxi Sarango**, ha sido dirigido, revisado y corregido en todas sus partes durante su ejecución, cumpliendo con los requisitos establecidos por las normas generales para la graduación en la Universidad Nacional de Loja, tanto en aspectos de forma como de contenido; por lo cual autorizo su presentación sustentación y defensa ante el tribunal designado para el efecto.

Loja, 27 de marzo del 2019

Atentamente



Ing. Pablo Fernando Ordoñez Ordoñez Mg. Sc.

DIRECTOR DE TESIS

Autoría

Yo, **MARTHA CRISTINA SUNTAXI SARANGO** declaro ser autora del presente trabajo de titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación del Trabajo de Titulación en el Repositorio Institucional – Biblioteca Virtual.

Firma:



Cédula: 1104798796

Fecha: 27 de marzo del 2019

CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.

Yo, **Martha Cristina Suntaxi Sarango**, declaro ser el autor de la tesis titulada: **DETECCIÓN DE TÉCNICAS DE APRENDIZAJE PROFUNDO APLICADAS EN LAS DIFERENTES ÁREAS DEL CONOCIMIENTO, EMPLEANDO EL MÉTODO DE REVISIÓN SISTEMÁTICA DE LITERATURA**, como requisito para optar al grado de: **INGENIERA EN SISTEMAS**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el repositorio Digital Institucional:

Los usuarios pueden consultar el contexto de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad. La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los once días del mes de junio del 2019.



Firma:

Autora: Martha Cristina Suntaxi Sarango

Cédula: 1104798796

Dirección: Loja (Malacatos, El Carmen)

Correo Electrónico: mcsuntaxis@unl.edu.ec

Teléfono: 2107298 **Celular:** 0993920789

DATOS COMPLEMENTARIOS

Director de Tesis: Ing. Pablo Fernando Ordoñez Ordoñez Mg. Sc.

Tribunal de Grado: Ing. Hernán Leonardo Torres Carrión Mg. Sc.

Ing. Luis Antonio Chamba Eras Mg. Sc.

Ing. Edison Leonardo Coronel Romero Mg. Sc.

Dedicatoria

Es mi deseo como sencillo gesto de agradecimiento dedicarles mi Trabajo de Titulación a:

La memoria de mi tía Josefina quien, con cariño compartió mi infancia y siempre está presente en mis recuerdos.

Mis abuelitos Genoveva y Eduardo, por apoyarme, cuidarme y siempre estar presentes en los momentos importantes de mi vida.

Mis padres Mariana Judith y Gregorio, por darme las herramientas para buscar y conseguir mis metas, por las aventuras compartidas y por todo el amor y apoyo incondicional que me han brindado siempre.

Mis hermanos Víctor y Fernanda y a mis sobrinos por los gratos momentos compartidos

Mis primos Eduardo, Luis, a mis amigos Jaritza, Pablo, Stalin, Byron, Víctor, André y demás compañeros por estar presentes en esta etapa y ser parte de mi crecimiento personal y profesional.

M. Cristina Suntaxi Sarango

Agradecimiento

Gracias a mis padres por ser los principales promotores de mis sueños, gracias a ellos por cada día confiar y creer en mí y en mis expectativas, a mi madre Mariana Judith por siempre desear y anhelar lo mejor para mi vida y por ser ejemplo de trabajo, esfuerzo y dedicación, a mi padre Gregorio por cada consejo y por cada una de sus palabras que guiaron mi vida, a mis abuelitos quienes fueron mi guía moral y mis compañeros en todo este trayecto.

Mi sincero agradecimiento a la Universidad Nacional de Loja, a la Facultad de Energía las Industrias y los Recursos Naturales no Renovables y en especial a la Carrera de Ingeniería en Sistemas por haberme permitido fórmame, y en ella, gracias a todos los docentes que fueron participes de este proceso, ya sea de manera directa o indirecta.

De manera muy especial quiero agradecer al Ing. Pablo Fernando Ordoñez Ordoñez, quien con sus conocimientos y acertada dirección permitió que el presente trabajo de titulación se realizara con éxito.

Finalmente quiero agradecer a la vida por este nuevo triunfo, gracias a todas las personas que me apoyaron y creyeron en la realización de este trabajo.

M. Cristina Suntaxi Sarango

Tabla de Contenidos

Índice General

CERTIFICACIÓN.....	1
AUTORÍA.....	III
DEDICATORIA.....	V
AGRADECIMIENTO	VI
TABLA DE CONTENIDOS	VII
ÍNDICE GENERAL.....	VII
ÍNDICE DE FIGURAS.....	X
ÍNDICE DE TABLAS.....	XII
1. TÍTULO	1
2. RESUMEN.....	2
2.1. ABSTRACT.....	3
3. INTRODUCCIÓN.....	4
4. REVISIÓN DE LITERATURA.....	6
4.1. REVISIÓN SISTEMÁTICA DE LITERATURA	6
4.1.1. <i>Protocolo de SLR</i>	7
4.2. MACHINE LEARNING	9
4.2.1. <i>Paradigmas de aprendizaje</i>	11
4.3. DEEP LEARNING	12
4.4. REDES NEURONALES ARTIFICIALES	13
4.4.1. <i>Elementos de una red neuronal artificial</i>	14
4.4.2. <i>Neurona Artificial</i>	15
4.4.3. <i>Funciones de activación</i>	16
4.4.4. <i>Arquitectura de redes neuronales</i>	17
4.5. REDES NEURONALES PROFUNDAS.....	19
4.5.1. <i>Perceptrón multicapa (MLP)</i>	19
4.5.1.1. Perceptrón	20
4.5.1.2. Perceptrón multicapa (MLP).....	20
4.5.2. <i>Autoencoders apilados (SAE)</i>	21
4.5.2.1. Autoencoder (AE).....	22
4.5.2.2. Autoencoder Apilado (SAE)	22
4.5.3. <i>Redes de Creencia Profunda (DBN)</i>	23
4.5.3.1. Máquinas de Boltzmann Restringidas (RBM)	23
4.5.3.2. Redes de Creencia Profunda (DBN)	24
4.5.4. <i>Redes Convolucionales (CNN)</i>	25
4.5.5. <i>Redes Long Short-Term Memory (LSTM)</i>	26
4.5.5.1. Redes Neuronales Recurrentes RNN	26
4.5.5.2. Redes Long Short-Term Memory (LSTM)	26
4.6. TECNOLOGÍAS	27
4.6.1. <i>Keras</i>	27
4.6.1.1. Modelos	28
4.6.1.2. Capas	28
4.6.1.3. Inicializadores	29
4.6.1.4. Funciones de pérdida	29
4.6.1.5. Optimizadores.....	29

4.6.2. <i>Python</i>	30
4.6.3. <i>Jupyter</i>	30
5. MATERIALES Y MÉTODOS	31
5.1. OBJETIVO 1:	31
5.1.1. <i>Planificación de la Revisión</i>	32
5.1.2. <i>Ejecución de la Revisión</i>	32
5.2. OBJETIVO 2	35
5.2.1. <i>Extracción de datos</i>	35
5.2.2. <i>Pre procesamiento y análisis exploratorio</i>	36
5.2.3. <i>Elegir los modelos</i>	38
5.2.4. <i>Entrenamiento y ajuste de parámetros</i>	38
5.2.5. <i>Hardware y software</i>	39
5.3. OBJETIVO 3	41
6. RESULTADOS.....	42
6.1. OBJETIVO 1:	42
6.1.1. <i>Identificación de la necesidad de una revisión.</i>	42
6.1.2. <i>Desarrollo de un protocolo de Revisión.</i>	42
6.1.3. <i>Selección de estudios primarios</i>	45
6.1.4. <i>Evaluación de Calidad.</i>	46
6.1.5. <i>Extracción de Datos.</i>	46
6.1.6. <i>Síntesis de datos.</i>	46
6.1.7. <i>Discusión de la SLR</i>	57
6.2. OBJETIVO 2	59
6.2.1. <i>Pre procesamiento y análisis exploratorio</i>	59
6.2.2. <i>Elegir los modelos</i>	63
6.2.3. <i>Entrenamiento y ajuste de parámetros red LSTM</i>	64
6.2.3.1. <i>Caso base</i>	64
6.2.3.2. <i>Dropout</i>	65
6.2.3.3. <i>Tamaño del Lag</i>	66
6.2.3.4. <i>Número de Bloques</i>	66
6.2.3.5. <i>Stateful</i>	67
6.2.3.6. <i>Número de Épocas</i>	67
6.2.3.7. <i>Función de activación</i>	67
6.2.3.8. <i>Número de capas</i>	68
6.2.4. <i>Modelo ARIMA</i>	69
6.2.5. <i>Red neuronal superficial</i>	69
6.3. OBJETIVO 3	72
7. DISCUSIÓN	73
7.1. DESARROLLO DE LA PROPUESTA ALTERNATIVA	73
7.2. VALORACIÓN TÉCNICA, ECONÓMICA Y CIENTÍFICA.....	76
7.2.1. <i>Valoración Técnica</i>	76
7.2.2. <i>Valoración Económica</i>	76
7.2.3. <i>Valoración Científica</i>	78
8. CONCLUSIONES	79
9. RECOMENDACIONES.....	80
10. BIBLIOGRAFÍA	81
11. ANEXOS	87
ANEXO 1: EXTRACCIÓN DE DATOS	88
Área de Conocimiento: INGENIERÍA	88

Área de Conocimiento: INDUSTRIA	100
Área de Conocimiento: INVESTIGACIÓN.....	101
Área de Conocimiento: SALUD	104
Área de Conocimiento: FINANZAS.....	110
ANEXO 2: TABLA INTEGRAL DE LAS PREGUNTAS DE INVESTIGACIÓN	120
ANEXO 3: CÓDIGO COMPLETO DEL MODELO DESARROLLADO EN KERAS	122
ANEXO 4: CARTA DE ACEPTACIÓN DEL ARTÍCULO	125
ANEXO 5: ARTÍCULO INDEXADO EN KNOWLEDGE E.....	126
ANEXO 6: CERTIFICADO DE PARTICIPACIÓN EN EL SIIPRIN'2018.....	140
ANEXO 7: CERTIFICADO DE PARTICIPACIÓN EN EL II CICLO DE CONFERENCIAS CIS-UNL 2019	141
ANEXO 8: DECLARACIÓN DE CONFIDENCIALIDAD.....	142
ANEXO 9: LICENCIA CREATIVE COMMONS.....	143

Índice de Figuras

FIGURA 1. ESTRUCTURA DEL DOCUMENTO DE TRABAJO DE TITULACIÓN (IMAGEN PROPIA)	5
FIGURA 2. PROCESO DE CONSTRUCCIÓN DE MODELOS ML (IMAGEN PROPIA)	11
FIGURA 3. RELACIÓN ENTRE IA, ML Y DL (TOMADA DE [12]).....	13
FIGURA 4. RED NEURONAL TOTALMENTE CONECTADA (TOMADA DE [14])	14
FIGURA 5. ESQUEMA GENERAL DE UNA NEURONA ARTIFICIAL (TOMADA DE [15])	15
FIGURA 6. FUNCIONES DE ACTIVACIÓN (A) LINEAL, (B) BINARIA, (C) SIGMOIDE, (D) TANGENTE, (E) ReLU Y (F) SOFGPLUS (IMAGEN PROPIA).....	17
FIGURA 7. ARQUITECTURA DE UNA RED NEURONAL MONOCAPA (IMAGEN PROPIA)	18
FIGURA 8. ARQUITECTURA DE UNA RED NEURONAL MULTICAPA (IMAGEN PROPIA)	18
FIGURA 9. ARQUITECTURA DE UNA RED NEURONAL RECURRENTE (IMAGEN PROPIA).....	19
FIGURA 10. DIFERENCIA RED NEURONAL TRADICIONAL Y RED NEURONAL PROFUNDA (TOMADA DE [18])	19
FIGURA 11. NEURONA ARTIFICIAL O PERCEPTRÓN (TOMADA DE [16]).....	20
FIGURA 12. RED NEURONAL DE PERCEPTRÓN MULTICAPA (IMAGEN PROPIA)	21
FIGURA 13. ARQUITECTURA DE UN AUTOENDODER (TOMADA DE [19])	22
FIGURA 14. ARQUITECTURA DE UN AUTOENCODER APILADO (TOMADA DE [20])	23
FIGURA 15. ARQUITECTURA DE RED TIPO MÁQUINA DE BOLTZMANN RESTRINGIDA (TOMADA DE [20])	24
FIGURA 16. ARQUITECTURA DE UNA RED DE CREENCIA PROFUNDA (TOMADA DE [21])	24
FIGURA 17. ARQUITECTURA DE UNA RED NEURONAL CONVOLUCIONAL (TOMADA DE [22])	25
FIGURA 18. ARQUITECTURA DE RED LSTM (TOMADA DE [11])	27
FIGURA 19. ETAPAS DE LA SLR SEGÚN KITCHENHAM (IMAGEN PROPIA)	31
FIGURA 20. PROCESO DE SELECCIÓN DE ESTUDIOS PRIMARIOS (IMAGEN PROPIA).....	33
FIGURA 21. ETAPAS DE LA CONSTRUCCIÓN DEL MODELO ML (IMAGEN PROPIA)	35
FIGURA 22. ESTUDIOS PRIMARIOS SELECCIONADOS EN CADA BASE DE DATOS CIENTÍFICA	45
FIGURA 23. SÍNTESIS DEL PROCESO DE SELECCIÓN DE ESTUDIOS PRIMARIOS.	47
FIGURA 24. RESULTADO DE ARTÍCULOS SELECCIONADOS POR AÑO DE PUBLICACIÓN	48
FIGURA 25. RESULTADO DE ARTÍCULOS SELECCIONADOS POR ÁREA DE CONOCIMIENTO.....	49
FIGURA 26. RESULTADO DE APLICACIONES DEL DL EN EL ÁREA DE INGENIERÍA.....	51
FIGURA 27. RESULTADO DE APLICACIONES DEL DL EN EL ÁREA DE INDUSTRIA.....	52
FIGURA 28. RESULTADO DE APLICACIONES DEL DL EN EL ÁREA DE INVESTIGACIÓN	52
FIGURA 29. RESULTADO DE APLICACIONES DEL DL EN EL ÁREA DE SALUD	53
FIGURA 30. RESULTADO DE APLICACIONES DEL DL EN EL ÁREA DE FINANZAS.....	53
FIGURA 31. RESULTADO DE LOS MODELOS DL APLICADOS EN LOS ESTUDIOS PRIMARIOS	56
FIGURA 32. RESULTADO DE LOS MODELOS DL DE ACUERDO AL TIPO DE DATASET EN LA QUE SE APLICAN	56
FIGURA 33. SERIE DE TIEMPO DE LAS VENTAS DE ILE	60
FIGURA 34. HISTOGRAMA DEL TOTAL DE VENTAS (2011-2018)	61
FIGURA 35. CAPTURA DE ESTADÍSTICAS OBTENIDAS EN R CON EL COMANDO SUMMARY()	61
FIGURA 36. SERIE DE TIEMPO ESTACIONARIA DE LAS VENTAS DE ILE	62
FIGURA 37. CAPTURA DE LA EJECUCIÓN DEL TEST DE DICKEY-FULLER.....	62
FIGURA 38. HISTOGRAMA DE VENTAS ESTACIONARIAS.....	63
FIGURA 39. SERIE DE TIEMPO DE LAS VENTAS DE ILE ESCALADA Y ESTACIONARIA	63
FIGURA 40. CAPTURA DE LOS RESULTADOS CON LOS PARÁMETROS BASE	64
FIGURA 41. CAPTURA DE LAS ÉPOCAS DE ENTRENAMIENTO DEL CASO BASE	65
FIGURA 42. ESTRUCTURA DE LA RED Y PARÁMETROS DEL CASO BASE	65
FIGURA 43. CAPTURA DE LOS RESULTADOS DEL CASO BASE	65
FIGURA 44. CAPTURA DE LOS RESULTADOS MODIFICANDO EL DROPOUT	66
FIGURA 45 CAPTURA DE LOS RESULTADOS MODIFICANDO EL LAG.....	66
FIGURA 46. CAPTURA DE LOS RESULTADOS MODIFICANDO EL NÚMERO DE BLOQUES POR CAPA	66
FIGURA 47. CAPTURA DE LOS RESULTADOS MODIFICANDO EL PARÁMETRO STATEFUL	67
FIGURA 48. CAPTURA DE LOS RESULTADOS MODIFICANDO EL NÚMERO DE ÉPOCAS.....	67
FIGURA 49. CAPTURA DE LOS RESULTADOS MODIFICANDO LA FUNCIÓN DE ACTIVACIÓN	68
FIGURA 50. CAPTURA DE LOS RESULTADOS MODIFICANDO EL NÚMERO DE CAPAS	68
FIGURA 51. RESULTADOS DE LA LSTM EN KERAS.....	69
FIGURA 52 CAPTURA DEL MODELO ARIMA (CON <i>auto.arima</i> Y EN R)	69

FIGURA 53. CAPTURA DE LOS RESULTADOS MODIFICANDO LOS DIFERENTES PARÁMETROS DE LA RED	70
FIGURA 54. CAPTURA DE LA ESTRUCTURA DE LA RED	70
FIGURA 55. RESULTADOS DEL PERCEPTRÓN UNICAPA EN KERAS	71
FIGURA 56. CAPTURA DE LA MEMORIA DEL CONGRESO SIIPRIN'2018	72

Índice de Tablas

TABLA I. EXTRACCIÓN DE DATOS	34
TABLA II. FRAGMENTO DEL DATASET VENTAS DE ILE	36
TABLA III. PREGUNTAS DE INVESTIGACIÓN	43
TABLA IV. FUENTES DE BÚSQUEDA	43
TABLA V. CADENAS DE BÚSQUEDA	44
TABLA VI. CRITERIOS DE INCLUSIÓN	44
TABLA VII. CRITERIOS DE EXCLUSIÓN	45
TABLA VIII. RESULTADOS DEL PROCESO DE SELECCIÓN DE ESTUDIOS PRIMARIOS.....	45
TABLA IX. LISTA DE VERIFICACIÓN DE EVALUACIÓN DE CALIDAD	46
TABLA X. TRABAJOS SELECCIONADOS POR CADA BASE DE DATOS CIENTÍFICA	48
TABLA XI. TRABAJOS SELECCIONADOS POR CADA ÁREA DE CONOCIMIENTO	49
TABLA XII. IDENTIFICACIÓN DE APLICACIONES DEL DL EN LAS DIFERENTES ÁREAS DE CONOCIMIENTO (PI_2)	50
TABLA XIII. IDENTIFICACIÓN DE MODELOS / ALGORITMOS (PI_3)	54
TABLA XIV. FRAGMENTO DE LAS VENTAS AJUSTADA POR PRODUCTO	59
TABLA XV. FRAGMENTO DEL DATASET AJUSTADO A LA VENTA TOTAL.....	60
TABLA XVI. VALORES USADOS EN CADA PARÁMETRO DE LA RED LSTM	68
TABLA XVII. ERROR RMSE ALCANZADO POR LOS MODELOS DESARROLLADOS	71
TABLA XVIII. TALENTO HUMANO	76
TABLA XIX. RECURSOS HARDWARE Y SOFTWARE.....	77
TABLA XX. PRESTACIÓN DE SERVICIOS A ADQUIRIR	77
TABLA XXI. MATERIALES DE OFICINA	78
TABLA XXII. PRESUPUESTO TOTAL DEL PROYECTO.....	78
TABLA XXIII. RESULTADOS DEL ARTÍCULO EP01.....	88
TABLA XXIV. RESULTADOS DEL ARTÍCULO EP02	88
TABLA XXV. RESULTADOS DEL ARTÍCULO EP03	89
TABLA XXVI. RESULTADOS DEL ARTÍCULO EP04	89
TABLA XXVII. RESULTADOS DEL ARTÍCULO EP05	90
TABLA XXVIII. RESULTADOS DEL ARTÍCULO EP06	90
TABLA XXIX. RESULTADOS DEL ARTÍCULO EP07	91
TABLA XXX. RESULTADOS DEL ARTÍCULO EP08	91
TABLA XXXI. RESULTADOS DEL ARTÍCULO EP09	92
TABLA XXXII. RESULTADOS DEL ARTÍCULO EP10	92
TABLA XXXIII. RESULTADOS DEL ARTÍCULO EP11	93
TABLA XXXIV. RESULTADOS DEL ARTÍCULO EP12	93
TABLA XXXV. RESULTADOS DEL ARTÍCULO EP13	94
TABLA XXXVI. RESULTADOS DEL ARTÍCULO EP14	94
TABLA XXXVII. RESULTADOS DEL ARTÍCULO EP15	95
TABLA XXXVIII. RESULTADOS DEL ARTÍCULO EP16	95
TABLA XXXIX. RESULTADOS DEL ARTÍCULO EP17	96
TABLA XL. RESULTADOS DEL ARTÍCULO EP18	96
TABLA XLI. RESULTADOS DEL ARTÍCULO EP19	97
TABLA XLII. RESULTADOS DEL ARTÍCULO EP20	97
TABLA XLIII. RESULTADOS DEL ARTÍCULO EP21	98
TABLA XLIV. RESULTADOS DEL ARTÍCULO EP22	98
TABLA XLV. RESULTADOS DEL ARTÍCULO EP23	99
TABLA XLVI. RESULTADOS DEL ARTÍCULO EP24	99
TABLA XLVII. RESULTADOS DEL ARTÍCULO EP25	100
TABLA XLVIII. RESULTADOS DEL ARTÍCULO EP26	100
TABLA XLIX. RESULTADOS DEL ARTÍCULO EP27	101
TABLA L. RESULTADOS DEL ARTÍCULO EP28	101
TABLA LI. RESULTADOS DEL ARTÍCULO EP29	102
TABLA LII. RESULTADOS DEL ARTÍCULO EP30	102
TABLA LIII. RESULTADOS DEL ARTÍCULO EP31	103
TABLA LIV. RESULTADOS DEL ARTÍCULO EP32	103

TABLA LV. RESULTADOS DEL ARTÍCULO EP33	104
TABLA LVI.RESULTADOS DEL ARTÍCULO EP34.....	104
TABLA LVII. RESULTADOS DEL ARTÍCULO EP35	105
TABLA LVIII. RESULTADOS DEL ARTÍCULO EP36	105
TABLA LIX. RESULTADOS DEL ARTÍCULO EP37	106
TABLA LX. RESULTADOS DEL ARTÍCULO EP38	106
TABLA LXI. RESULTADOS DEL ARTÍCULO EP39	107
TABLA LXII. RESULTADOS DEL ARTÍCULO EP40	107
TABLA LXIII. RESULTADOS DEL ARTÍCULO EP41	108
TABLA LXIV. RESULTADOS DEL ARTÍCULO EP42.....	108
TABLA LXV. RESULTADOS DEL ARTÍCULO EP43.....	109
TABLA LXVI. RESULTADOS DEL ARTÍCULO EP44.....	109
TABLA LXVII. RESULTADOS DEL ARTÍCULO EP45.....	110
TABLA LXVIII. RESULTADOS DEL ARTÍCULO EP46.....	110
TABLA LXIX. RESULTADOS DEL ARTÍCULO EP47.....	111
TABLA LXX. RESULTADOS DEL ARTÍCULO EP48.....	111
TABLA LXXI. RESULTADOS DEL ARTÍCULO EP49.....	112
TABLA LXXII. RESULTADOS DEL ARTÍCULO EP50.....	112
TABLA LXXIII. RESULTADOS DEL ARTÍCULO EP51.....	113
TABLA LXXIV. RESULTADOS DEL ARTÍCULO EP52	113
TABLA LXXV. RESULTADOS DEL ARTÍCULO EP53	114
TABLA LXXVI. RESULTADOS DEL ARTÍCULO EP54	114
TABLA LXXVII. RESULTADOS DEL ARTÍCULO EP55	115
TABLA LXXVIII. RESULTADOS DEL ARTÍCULO EP56	115
TABLA LXXIX. RESULTADOS DEL ARTÍCULO EP57	116
TABLA LXXX. RESULTADOS DEL ARTÍCULO EP58	116
TABLA LXXXI. RESULTADOS DEL ARTÍCULO EP59	116
TABLA LXXXII. RESULTADOS DEL ARTÍCULO EP60	117
TABLA LXXXIII. RESULTADOS DEL ARTÍCULO EP61	117
TABLA LXXXIV. RESULTADOS DEL ARTÍCULO EP62	118
TABLA LXXXV. RESULTADOS DEL ARTÍCULO EP63.....	118
TABLA LXXXVI. RESULTADOS DEL ARTÍCULO EP64	119
TABLA LXXXVII. RESULTADOS INTEGRADOS DE PI_2 Y PI_3	120

1. Título

“DETECCIÓN DE TÉCNICAS DE APRENDIZAJE PROFUNDO APLICADAS EN LAS DIFERENTES ÁREAS DEL CONOCIMIENTO, EMPLEANDO EL MÉTODO DE REVISIÓN SISTEMÁTICA DE LITERATURA.”

2. Resumen

Deep Learning es un campo de investigación que se ha vuelto muy popular en los últimos años para el aprendizaje de datos, ya que alcanza abstracciones de alto nivel a través del modelado de varias capas de procesamiento. Deep Learning ha inspirado a un gran número de investigadores que emplean algoritmos profundos para detectar patrones y extraer características de los datos, con el propósito de obtener resultados más precisos que los modelos comúnmente usados. Es por ello que, gracias a la creciente aplicación del Deep Learning, el presente Trabajo de Titulación tiene como objetivo determinar las técnicas y/o modelos de aprendizaje profundo que se aplican actualmente en las áreas de conocimiento de ingeniería, medicina, investigación, industria y finanzas. Para ello se desarrolló una Revisión Sistemática de Literatura en base al protocolo de Bárbara Kitchenham resultando en la selección de 64 estudios primarios de los cuales se extrajo el problema de investigación y el modelo Deep Learning aplicado.

En la Revisión Sistemática de Literatura se obtuvo como resultado que, el modelo Deep Learning LSTM es ideal para problemas de predicción de ventas, para validar este resultado se emplearon los datos de ventas de ILE (Industria Lojana de Especerías) en desarrollar un modelo de predicción de ventas empleando la red LSTM, la cual se comparó con los modelos de Machine Learning red neuronal simple y el modelo ARIMA, empleando como métrica de evaluación el error cuadrático medio, los experimentos demostraron que a pesar del pequeño tamaño de los datos el modelo LSTM obtiene menor error que los otros modelos.

Finalmente, el DL tiene una amplia gama de aplicación, entre ellas la predicción de ventas, donde se encontró que el modelo LSTM tiene un menor error que los modelos Machine Learning, a pesar de que los datos son limitados. Por tanto, el modelo LSTM es más eficaz que los modelos Machine Learning, aunque para validar la eficacia de DL en trabajos futuros se recomienda comparar los resultados con otros modelos DL.

2.1. Abstract

Deep Learning is a field of research that has become very popular in recent years for data learning, as it reaches high-level abstractions through the modeling of several layers of processing. Deep Learning has inspired a large number of researchers who use deep algorithms to detect patterns and extract data characteristics, in order to obtain more accurate results than commonly used models. That is why, thanks to the growing application of Deep Learning, the present Degree Work aims to determine the techniques and / or deep learning models that are currently applied in the areas of knowledge of engineering, medicine, research, industry and finance. For this, a Systematic Literature Review was developed based on the protocol of Barbara Kitchenham, resulting in the selection of 64 primary studies from which the research problem and the applied Deep Learning model were extracted.

In the Systematic Literature Review it was obtained that, the Deep Learning LSTM model is ideal for problems of sales prediction, to validate this result the sales data of ILE (Lojana's Industry of Especerías) were used in developing a prediction model of sales using the LSTM network, which was compared with the models of Machine Learning simple neural network and the ARIMA model, using as an evaluation metric the mean square error, the experiments showed that despite the small size of the data the LSTM model gets less error than the other models.

Finally, the DL has a wide range of applications, among them the sales prediction, where it was found that the LSTM model has a lower error than the Machine Learning models, although the data is limited. Therefore, the LSTM model is more efficient than Machine Learning models, although to validate the effectiveness of DL in future work, it is recommended to compare the results with other DL models.

3. Introducción

El aprendizaje profundo, denominado en inglés Deep Learning (DL) surge por la necesidad de procesar las masivas cantidades de datos que se generan a diario y para las cuales los modelos actuales de machine learning no dan abasto, el objetivo del DL es alcanzar abstracciones de alto nivel, a través del modelado de redes neuronales con varias capas de procesamiento, con estos modelos es posible extraer cualidades específicas de los datos agregando capas de neuronas a una red denominada profunda por la cantidad de capas ocultas, para ello se necesitan grandes cantidades de datos y de procesamiento, características que son alcanzables en la actualidad. DL también cuenta con la capacidad de capturar las características de los datos sin intervención humana, esta característica es ideal para el campo de la visión artificial y el procesamiento del lenguaje natural, por lo que este campo del machine learning no es nuevo, pero tuvo mayor acogida recientemente gracias a los avances en procesamiento y se encuentra en constante desarrollo, ya que ha demostrado que es la forma más eficiente de hacer con un computador una simulación semejante a nuestro cerebro [1].

La elaboración del presente Trabajo de Titulación tiene como objetivo principal determinar y demostrar las técnicas o modelos de aprendizaje profundo que se aplican actualmente en las diferentes áreas de conocimiento.

Para dar cumplimiento al objetivo principal del Trabajo de Titulación se definió los siguientes objetivos específicos: a) Aplicar la técnica de revisión sistemática cualitativa de Bárbara Kitchenham para obtener los avances y conocimientos actuales de aplicaciones del Deep Learning en diferentes áreas de conocimiento; b) Demostrar el uso del Deep Learning con el resultado de la revisión sistemática cualitativa de Bárbara Kitchenham; c) Escribir los resultados logrados en un artículo para su envío a una revista indexada e inicie su proceso de publicación.

La estructura del documento Trabajo de Titulación (véase Figura 1) se describe a continuación:

En la sección Revisión de Literatura, se elaboró cuatro capítulos del área de estudio que ayudaron a sustentar los conocimientos aplicados en la ejecución del trabajo de titulación, también incluye un capítulo con las tecnologías utilizadas en el desarrollo del modelo de deep learning. La sección de Metodología, permitió detallar todos los

recursos de hardware y software empleados en la ejecución de este trabajo, y el desarrollo de los métodos aplicados para conseguir los objetivos específicos; el método de revisión sistemática de literatura basado en el protocolo propuesto por Kitchenham [2], el método para el desarrollo de modelos predictivos que incluye la extracción y preparación de los datos, finalmente el proceso de redacción y publicación de los resultados. La sección Resultados, sirvió para presentar todos los datos relevantes obtenidos en la ejecución del Trabajo de Titulación. En la sección Discusión, se realizó un análisis de los resultados obtenidos y como se cumplió con los objetivos. La sección Conclusiones, permitió expresar los resultados más relevantes rescatados luego de haber culminado el trabajo. Finalmente, en la sección Recomendaciones, se planteó aspectos a considerar para el desarrollo de futuros trabajos.

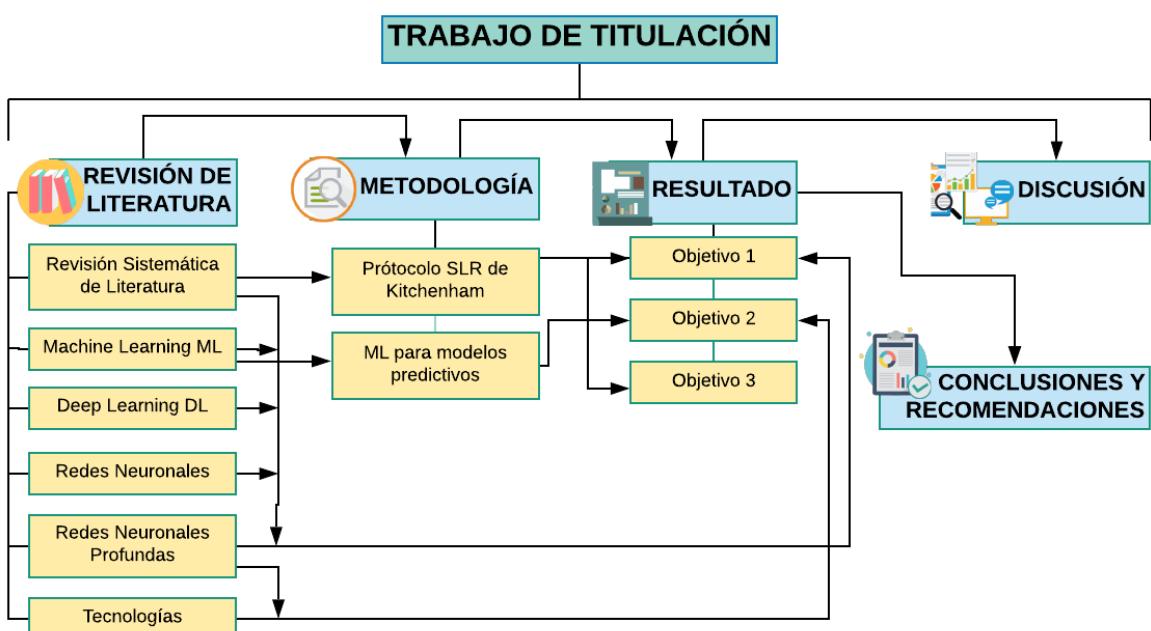


Figura 1. Estructura del documento de trabajo de titulación (Imagen propia)

4. Revisión de Literatura

En esta sección se aborda temas y conceptos que permiten al lector entender el objetivo de estudio del Trabajo de Titulación. En el capítulo 1 se revisan conceptos de revisiones bibliográficas, y el método de revisión sistemática de literatura, el cual se empleó para explorar en los documentos científicos los problemas de investigación en los que se aplican modelos del DL, el capítulo 2 expresa los conceptos de machine learning, una rama de la inteligencia artificial de la cual se deriva DL cuyos conceptos se aborda en el capítulo 3. El DL emplea redes neuronales artificiales por lo que en el capítulo 4 se revisaron diversos conceptos y características de redes neuronales, los elementos que las componen y como se organizan. Los modelos de DL que emplean redes neuronales profundas se explican el capítulo 5. En el capítulo 6 de tecnologías se habla acerca de las herramientas tecnológicas empleadas para diseñar el modelo de predicción de ventas empleando redes neuronales profundas.

4.1. REVISIÓN SISTEMÁTICA DE LITERATURA

La rápida progresión del conocimiento científico mediado por la aparición creciente de publicaciones en un medio como el Internet requiere la realización de revisiones de la literatura científica para tener un conocimiento actualizado sobre un tema de interés. Con la revisión bibliográfica se aproxima al conocimiento de un tema y ayuda a identificar el conocimiento científico actual que implican los vacíos en ciertas áreas de investigación. Es una sinopsis que involucra diferentes investigaciones y artículos que nos da una idea sobre cuál es el estado actual de la cuestión a investigar [3].

Según [3] la Revisión Bibliográfica se ha definido como “la operación de recuperar un conjunto de documentos o referencias bibliográficas que se publican en el mundo sobre un tema, un autor, una publicación o un trabajo específico”. Para [4], la revisión bibliográfica es “la selección de los documentos disponibles sobre el tema, que contienen información, ideas, datos y evidencias por escrito sobre un punto de vista en particular para cumplir ciertos objetivos o expresar determinadas opiniones sobre la naturaleza del tema y la forma en que se va a investigar, así como la evaluación eficaz de estos documentos en relación con la investigación que se propone”.

Existen diversos tipos de revisiones bibliográficas que no se abordaron en el desarrollo de este trabajo de titulación, para el estudio propuesto se empleó una revisión sistemática de literatura (SLR), aplicando la metodología expuesta por Bárbara Kitchenham.

Según Kitchenham [2], “una revisión sistemática de literatura es un medio para identificar, evaluar e interpretar toda la investigación disponible que sea relevante para una pregunta de investigación en particular, un área temática o un fenómeno de interés.” Nuestro fenómeno de interés es DL que ha ganado popularidad en los últimos años, con este estudio se pretende resumir la evidencia existente sobre DL, identificar los problemas de investigación alcanzados con el fin de sugerir áreas de investigación y proporcionar un marco de referencia para nuevas actividades de investigación.

4.1.1. Protocolo de SLR

Según Kitchenham [2], el protocolo de la SLR se compone de las siguientes partes:

- **Planificación de la revisión.** Su objetivo es determinar si hay una verdadera necesidad de la revisión y desarrollar un plan para ejecutar la misma. Esta fase se divide en dos actividades:
 - Identificación de la necesidad de la revisión: su objetivo es evitar la duplicidad para que cada nueva revisión sea original. Si existen revisiones previas debería evaluarse su calidad con base en su objetivo, fuentes de búsqueda, criterios de inclusión/exclusión, criterios de evaluación de la calidad, métodos de extracción y síntesis de datos.
 - Desarrollo de un protocolo de revisión: este protocolo especifica los métodos a ser utilizados en la SLR. Se pretende evitar los sesgos producidos por cambios en la pregunta de investigación o por una selección subjetiva de los estudios primarios. Este protocolo contiene: los antecedentes de la revisión, las preguntas de investigación, la estrategia de búsqueda, los criterios y procedimientos para la selección de estudios y para la evaluación de su calidad, la estrategia de extracción de datos y los métodos para su síntesis.

- **Conducción de la revisión.**
 - Identificación de la investigación: consiste en la identificación de estudios primarios relevantes a través de una estrategia de búsqueda que debe ser exhaustiva y libre de sesgo.
 - Selección de los estudios primarios: consiste en determinar la relevancia de cada estudio identificado, respecto a la pregunta de investigación planteada. Esta tarea se realiza con base en los criterios de inclusión/exclusión establecidos en el protocolo de revisión.
 - Evaluación de la calidad de los estudios: su propósito es obtener información adicional acerca de los estudios seleccionados para refinar los criterios de inclusión/exclusión. En este contexto, la calidad es definida en relación a la manera en que cada estudio reduce la probabilidad de sesgos y maximiza la validez interna y externa de sus resultados.
 - Extracción de datos y monitoreo: el objetivo es diseñar formularios para realizar la recolección de datos a partir de los estudios primarios de manera precisa y libre de sesgo.
 - Síntesis de datos: el propósito es resumir y agregar los datos que han sido extraídos de los estudios primarios seleccionados. Puede realizarse a través de la tabulación o descripción de las características de los estudios y sus resultados (síntesis no-cuantitativa) o de una síntesis cuantitativa (meta-análisis) a través de la aplicación de técnicas estadísticas.
- **Reporte de la revisión.** Como última fase de la RS, se debe generar un reporte que permita a los investigadores y profesionales entender las implicaciones de los resultados de la RS, al mismo tiempo que pueda valorar su validez. Se sugiere comunicar los resultados a través de diferentes medios: reportes técnicos, una sección en una tesis, artículos técnicos en revistas y conferencias, artículos no técnicos en revistas para profesionales, en notas de prensa, y en páginas Web. El procedimiento propone una estructura guiada por el protocolo de investigación donde se documenta información resumida de los componentes del protocolo y una discusión breve de resultados y conclusiones.

4.2. MACHINE LEARNING

Machine Learning o aprendizaje automático (ML) es un campo de las ciencias de la computación y parte de la Inteligencia Artificial (IA). IA es la disciplina que se ocupa del diseño de sistemas de computación inteligentes, es decir, sistemas que exhiben las características asociadas a la inteligencia humana, como comprensión del lenguaje, aprendizaje, razonamiento y resolución de problemas [5].

Existen diversos conceptos de ML:

- En [6], Arthur Samuel menciona que el ML “les da a las computadoras la habilidad de aprender sin ser explícitamente programadas”.
- Según [6], es la práctica de usar algoritmos para parsear datos, aprender de ellos y luego ser capaces de hacer una predicción o sugerencia sobre algo.
- En [7] sugieren que “el aprendizaje se refiere a un amplio espectro de situaciones en las cuales el aprendiz incrementa su conocimiento o sus habilidades para cumplir una tarea”. Para el aprendizaje automático se considera la resolución de problemas como un tipo de aprendizaje que consiste, una vez resuelto un tipo de problema, en ser capaz de reconocer la situación problemática y reaccionar usando la estrategia aprendida.
- En [9] el machine learning es un método de análisis de datos que automatiza la construcción de modelos analíticos. Se basa en la idea de que los sistemas pueden aprender de datos, identificar patrones y tomar decisiones con mínima intervención humana. El principal objetivo del ML es desarrollar la capacidad de generalizar y asociar, es decir los modelos ML deben actuar en escenarios familiares o nuevos con precisión y exactitud. El aprendizaje se mejora con la experiencia y el entrenamiento a partir de los datos de entrada.

En base a los conceptos antes expuestos se puede discernir que el ML da las pautas para la construcción de máquinas capaces de analizar datos o experiencias que le permita responder con precisión a escenarios similares.

Para [7] no hay un proceso estándar en la construcción de modelos ML, pero generalmente los pasos a seguir son (véase Figura 2):

- Recolectar datos (dataset): Dada la problemática a resolver, se debe investigar y obtener datos para alimentar el modelo. Es importante considerar la calidad y cantidad de información, ya que impactará directamente en el funcionamiento del modelo.
- Pre procesamiento: Se realiza un análisis exploratorio para detectar si hay correlaciones entre las distintas características seleccionadas. La selección de características, impactara directamente en los tiempos de ejecución y en los resultados. Se balancean los datos y se separan en dos grupos: uno para entrenamiento y otro para evaluación del modelo, la fracción más común es 80/20 pero puede variar según el caso y el volumen de datos. En esta etapa se realizan otras tareas como: normalización, eliminar duplicados y corrección de errores.
- Elección del modelo: se elige el modelo a aplicar en base al problema.
- Entrenamiento: Se emplea los datos de entrenamiento para ejecutar el modelo. los “pesos” se inician aleatoriamente, los pesos son los valores que multiplican o afectan a las relaciones entre las entradas y las salidas, se irán ajustando automáticamente por el algoritmo con cada entrenamiento.
- El test o evaluación: se realiza con entradas que el modelo desconoce para verificar la precisión del modelo ya entrenado.
- Ajustar parámetros: se aplica en el caso de no obtener resultados satisfactorios en el test, se realiza nuevamente el entrenamiento modificando los parámetros del modelo. Cada algoritmo tiene sus propios parámetros a ajustar, por ejemplo, en las redes neuronales artificiales se define la cantidad de neuronas por capa, este parámetro se modifica hasta reducir el error al mínimo.

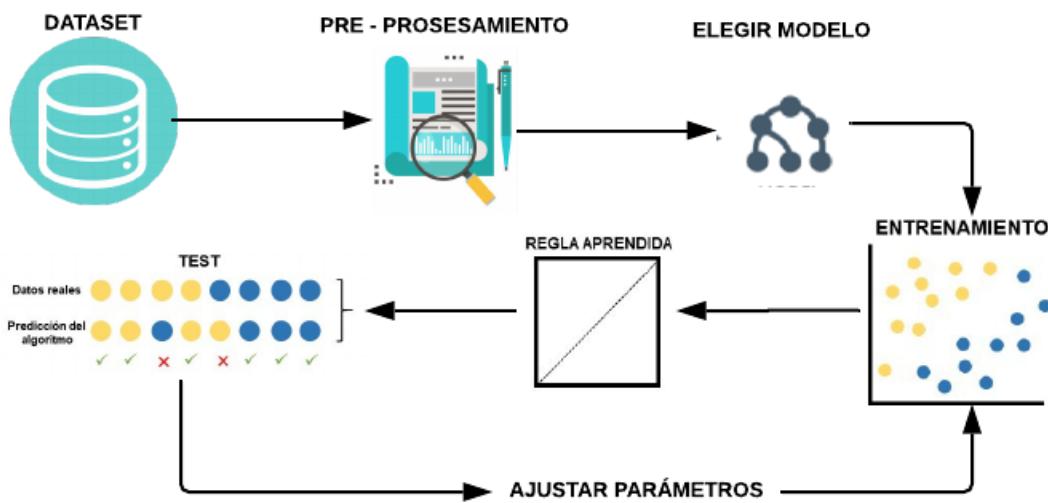


Figura 2. Proceso de construcción de modelos ML (Imagen propia)

4.2.1. Paradigmas de aprendizaje

Los algoritmos se dividen principalmente en tres grandes categorías: supervised learning (aprendizaje supervisado), unsupervised learning (aprendizaje no supervisado) y reinforcement learning (aprendizaje por refuerzo). Existen otras categorías poco empleadas en el estado del arte, a continuación, se presenta un concepto breve de los paradigmas más comunes [5].

Supervised learning: Depende de datos previamente etiquetados, estas etiquetas o rótulos son colocadas por seres humanos para asegurar la efectividad y calidad de los datos, por ejemplo, una data de animales contiene imágenes de animales y la etiqueta que corresponde a la especie, ya sea una imagen de un gato y la etiqueta “gato”, con estas etiquetas los algoritmos de ML buscan la función que permita asignar la especie o etiqueta a imágenes de animales nuevas que el modelo desconoce.

Unsupervised learning: En esta categoría lo que sucede es que al algoritmo se le despoja de cualquier etiqueta, de modo que no cuenta con ninguna indicación previa, en cambio, se le provee de una enorme cantidad de datos con las características propias de un objeto para que pueda determinar qué es, a partir de la información recopilada. Por ejemplo, en marketing, para conocer las costumbres de consumo de las personas, en este caso la data contiene información de gran cantidad de personas y los productos que consumen, los algoritmos de ML buscan la función que permita agrupar a las personas por el consumo.

Reinforcement Learning La máquina es capaz de aprender con base a pruebas y errores en un número de diversas situaciones. Aunque conoce los resultados desde el principio, no sabe cuáles son las mejores decisiones para llegar a obtenerlos. Lo que sucede es que el algoritmo progresivamente va asociando los patrones de éxito, para repetirlos una y otra vez hasta perfeccionarlos y volverse infalible; por ejemplo, máquinas entrenadas para jugar videojuegos como el Tetris, el objetivo es despejar las filas combinando fichas, por tanto, se entrena con las posibles combinaciones del juego, mientras realiza una combinación la máquina evalúa el resultado, si es positivo, aprende automáticamente a repetir esa decisión en el futuro, mientras que, si el resultado es negativo, evitará volver a tomar la misma.

4.3. DEEP LEARNING

Deep Learning (DL) es un concepto muy amplio, lo que conlleva a que no tenga una definición exacta, pero los estudios [8][9][10][11] coinciden en que DL comprende un conjunto de algoritmos que intentan modelar abstracciones de alto nivel, a través del modelado de varias capas de procesamiento. Este campo de investigación tuvo sus inicios en los años 80 y es una rama del aprendizaje automático donde los algoritmos se emplean en el entrenamiento de redes neuronales profundas para conseguir mayor precisión que el ML (véase Figura 3), en los últimos avances se ha llegado a un punto en el que DL supera a las personas en algunas tareas, por ejemplo, reconocer objetos presentes en imágenes [8].

La gran ventaja del DL es no requerir la caracterización de los datos para el aprendizaje, al contrario, el sistema reconoce sin ayuda la forma de capturar los datos para procesarlos automáticamente. La máquina puede reconocer patrones en los datos de entrada que a simple vista no son perceptibles al humano, por sus limitaciones sensoriales [10].

Según [13] además del desarrollo de mejores algoritmos, existen dos razones por las que últimamente resulta más útil emplear modelos profundos:

- **Datos etiquetados.** Para entrenar modelos DL se requieren grandes cantidades de datos. Por ejemplo, para desarrollo de un vehículo autónomo se necesitan millones de imágenes y miles de horas de video.

- Potencia de cálculo. Un modelo DL requiere de una potencia de cálculo significativa pero actualmente se cuenta con tecnologías como GPU, la programación paralela, los clústeres y el cálculo en la nube. Esto permite reducir el entrenamiento de una red neuronal profunda de semanas a horas.

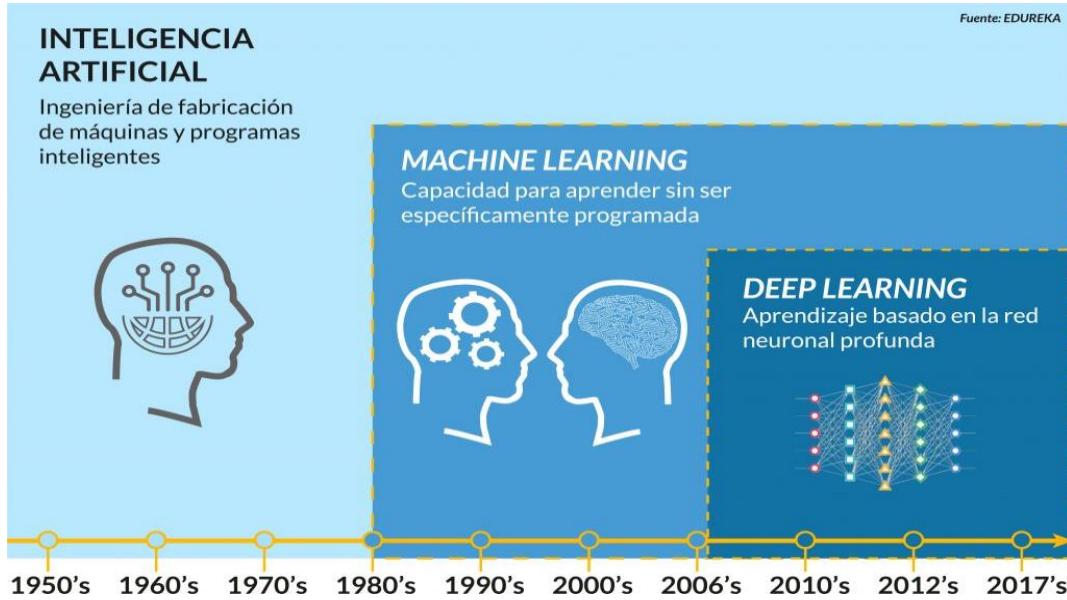


Figura 3. Relación entre IA, ML y DL (tomada de [12])

4.4. REDES NEURONALES ARTIFICIALES

Los modelos DL emplean redes neuronales artificiales por lo que se consideró necesario incluir estos conceptos para entender los modelos que se revisaran en el capítulo 5.

Las redes neuronales artificiales pretenden imitar el modelo biológico del cerebro, a través de modelos matemáticos organizados en redes interconectadas masivamente en paralelo y con organización jerárquica. Intentan interactuar con los objetos del mundo real del mismo modo que el sistema nervioso biológico [13].

Debido a su constitución y fundamentos las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro, estos modelos ofrecen ventajas para la resolución de problemas, lo cual ha aumentado su aplicación, entre las ventajas más significativas de las redes neuronales encontramos las expuestas por [14]:

- **Aprendizaje adaptativo.** Son un modelo ML y por lo tanto cuentan con la capacidad de aprender a realizar tareas basadas en una experiencia inicial.

- **Auto organización.** Pueden crear su propia representación de información con los datos que recibe en la etapa de aprendizaje
- **Tolerancia a fallos.** Se consideran dos tipos de tolerancia a fallos; la tolerancia a fallos de la red donde, de ocurrir una destrucción parcial de la red, algunas capacidades se pueden retener. Y la segunda la tolerancia a fallos respecto a los datos donde, las redes neuronales son capaces de aprender a reconocer patrones en datos distorsionados incompletos o con ruido.
- **Operación en tiempo real.** Admiten la computación paralela, pero para ello el hardware debe estar diseñado para permitir esta capacidad.
- **Fácil inserción.** Actualmente existen chips especializados para redes neuronales artificiales que permiten mejorar la capacidad de ciertas tareas que aplican estos modelos.

4.4.1. Elementos de una red neuronal artificial.

En la Figura 4 se observa un esquema de red neuronal artificial interconectada la cual se compone de capas. Las capas son las unidades estructurales de la red neuronal, dentro de una capa las neuronas suelen ser del mismo tipo. La capa de entrada captura los datos o señales del entorno, la capa oculta realiza las operaciones matemáticas, procesa los datos y los dirige a la capa de salida la cual entrega la respuesta de la red a los datos capturados por la capa de entrada [14].

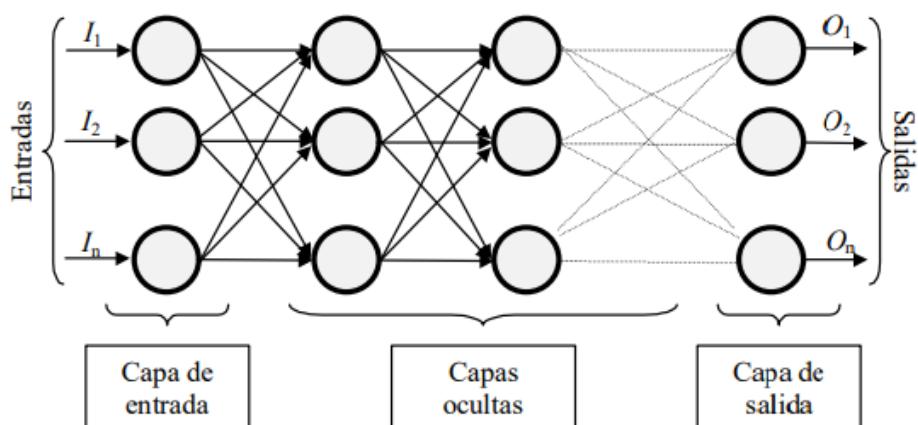


Figura 4. Red neuronal totalmente conectada (tomada de [14])

4.4.2. Neurona Artificial.

Las redes neuronales artificiales son modelos matemáticos compuestos por nodos (neuronas) interconectados entre sí y organizados en capas [17]. La estructura de neurona artificial empleada generalmente se muestra en la Figura 5 donde se aprecian cuatro elementos funcionales descritos por [17]:

- **Receptor.** Es el que recibe las señales de entrada x_i , provenientes de la capa de entrada o de otras neuronas y que son atenuadas o amplificadas con un peso w_i que constituye la conectividad entre la neurona anterior y la siguiente.
- **Sumador.** Efectúa la suma ponderada de las señales de entrada y el peso w , esta acción se describe en la ecuación 1.

$$S = \sum w_i x_i \quad (1)$$

- **Función de activación.** Aplica una función no lineal de umbral (ya sea de escalón o una curva logística) a la salida del sumador para decidir si la neurona se activa, disparando una salida o no (en el siguiente apartado se describen las diferentes funciones de activación más empleadas).
- **Salida.** Produce la señal de acuerdo al elemento anterior y constituye la salida de la neurona [15].

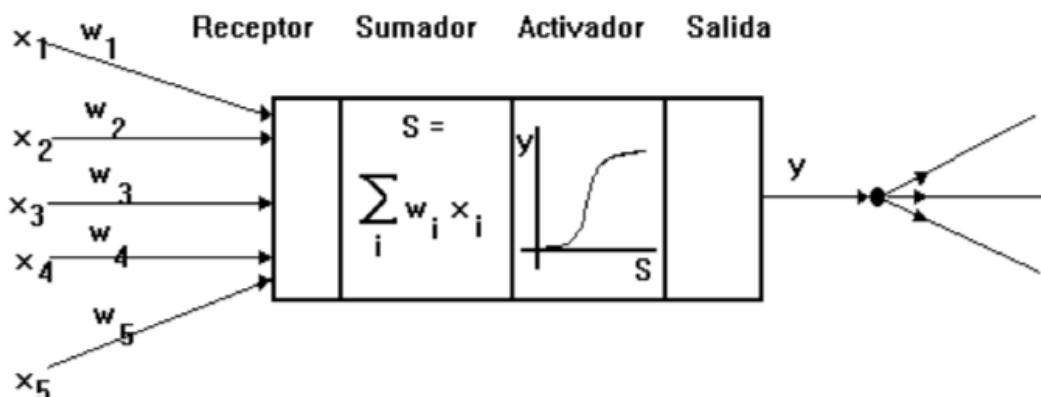


Figura 5. Esquema general de una neurona artificial (tomada de [15])

4.4.3. Funciones de activación.

Combina la función de propagación con el estado actual de la neurona para determinar si la neurona se activa o no. La función de activación es una función de x , donde x es la resta entre la entrada global (in_i) y el umbral (θ_i) (véase ecuación 2).

$$x = in_i - \theta_i \quad (2)$$

Existe variedad de funciones, a continuación, se listan las más comunes [9][16]:

- **Función lineal.** Devuelve como salida su valor de entrada. Su rango es $-\infty, \infty$ (véase Figura 6 (a)).

$$f(x) = x \quad (3)$$

- **Escalón binario.** Es una función sencilla determinada por un rango entre 0 y 1 (véase Figura 6 (b)).

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (4)$$

- **Sigmoide.** Una de las funciones más utilizadas para modelar redes neuronales, cuyo rango es continuo en los valores 0, 1 (véase Figura 6 (c)).

$$f(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

- **Tangente hiperbólica.** Esta función es utilizada por redes con salidas continuas. Cumple con un rango $(-1, 1)$, es monótona, y se aproxima a la función identidad en su origen (véase Figura 6 (d)).

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (6)$$

- **Unidad lineal rectificadora (ReLU).** Esta función acelera el cálculo al no necesitar exponenciales como la función tangente y sigmoide pero el principal inconveniente de esta función es el riesgo de que los pesos de algunas neuronas tomen valores que no activen la neurona (véase Figura 6 (e)).

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (7)$$

- **Función softplus:** Aproximación suavizada de la función de activación rectificadora (véase Figura 6 (f)).

$$f(x) = \ln(1 + e^x) \quad (8)$$

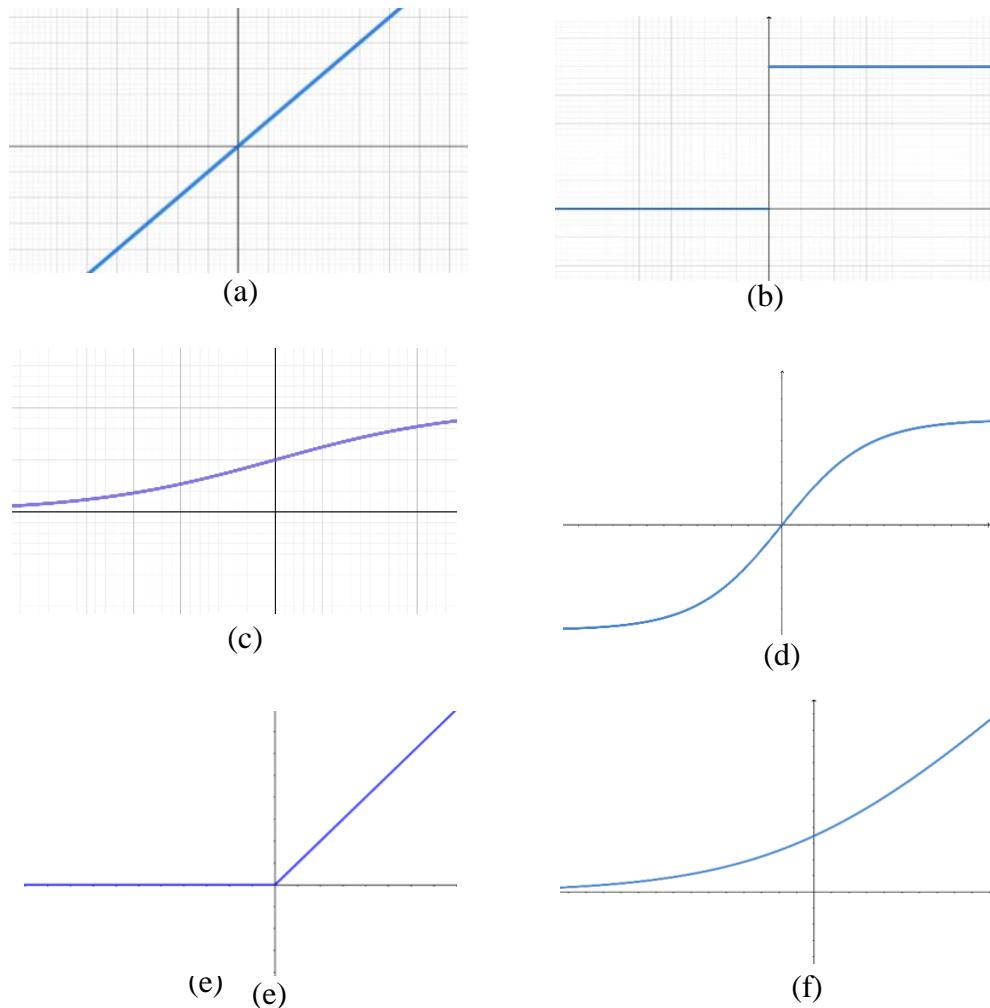


Figura 6. Funciones de activación (a) Lineal, (b) Binaria, (c) Sigmoid, (d) Tangente, (e) ReLU y (f) softplus (Imagen propia)

4.4.4. Arquitectura de redes neuronales

Consiste en la organización de las neuronas en la red, es decir la forma en que se alinean las capas de neuronas entre la entrada y salida de dicha red. Los parámetros fundamentales de la red son: número de capas, número de neuronas por capa, grado de conectividad y tipo de conexiones entre neuronas [14].

En [17] identifican las siguientes arquitecturas, en función de cómo se realiza la interconexión entre neuronas:

- **Redes monocapa.** Red neuronal más sencilla (véase Figura 7). Se establecen conexiones entre las neuronas que pertenecen a la única capa que constituye la red. Generalmente se emplean en la regeneración de información de entrada que se presenta a la red de forma incompleta o distorsionada.

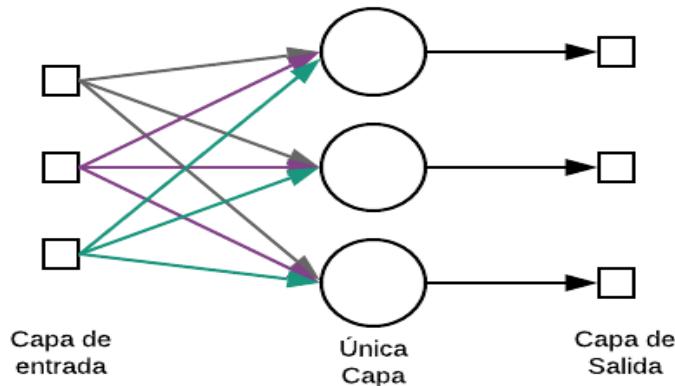


Figura 7. Arquitectura de una red neuronal monocapa (Imagen propia)

- **Redes multicapa.** Disponen de un conjunto de neuronas agrupadas en varias capas entre la capa de entrada y salida (capas ocultas). Este tipo de redes pueden estar total o parcialmente conectadas (véase Figura 8).

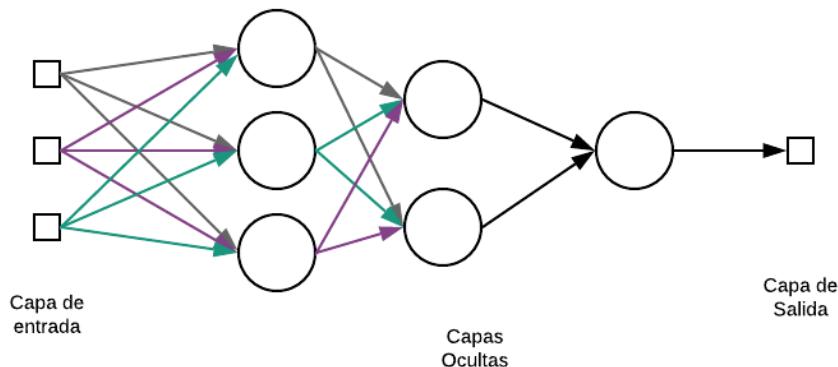


Figura 8. Arquitectura de una red neuronal multicapa (Imagen propia)

- **Redes recurrentes.** Esta red se diferencia de las anteriores en la existencia de lazos de realimentación en la red, por lo que no presentan una estructura de capas definida, además permiten conexiones arbitrarias entre las neuronas estableciendo temporalidad y permitiendo a la red tener memoria (véase Figura 9). Esta estructura las hace adecuadas para problemas no lineales.

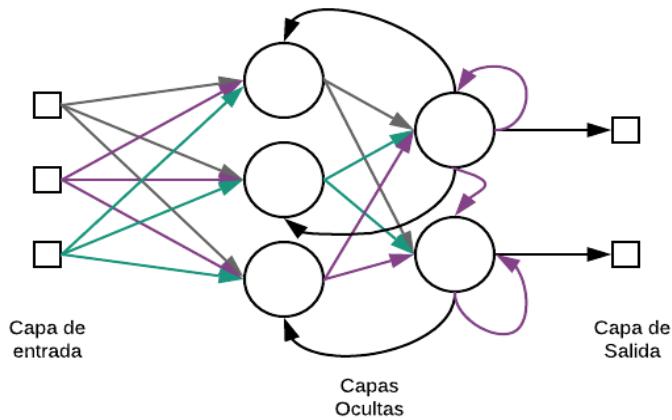


Figura 9. Arquitectura de una red neuronal recurrente (Imagen propia)

4.5. REDES NEURONALES PROFUNDAS

Los modelos del DL emplean arquitecturas de redes neuronales, por lo que se suelen denominar redes neuronales profundas, el concepto del DL hace referencia al número de capas de procesamiento, por tanto, una red neuronal profunda se diferencia de una red neuronal tradicional por el número de capas ocultas empleadas (véase Figura 10) [9].

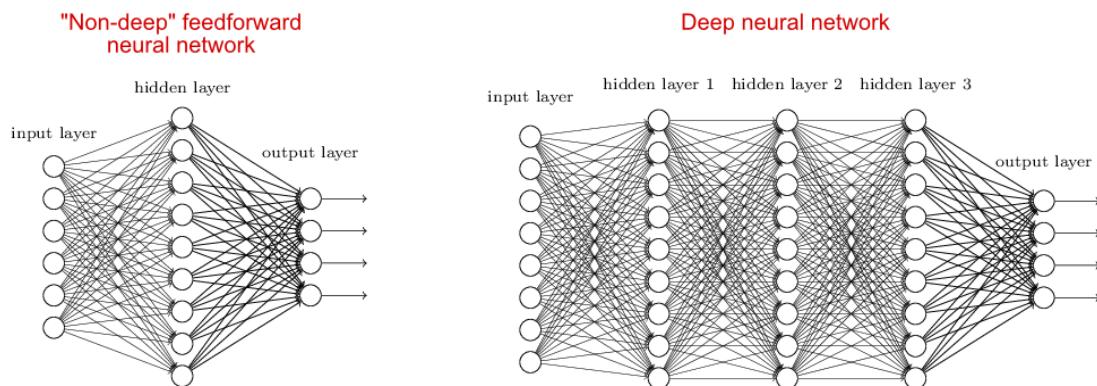


Figura 10. Diferencia red neuronal tradicional y red neuronal profunda (tomada de [18])

Actualmente existen diversas arquitecturas de redes neuronales profundas pero las más representativas según el estado del arte son:

4.5.1. Perceptrón multicapa (MLP)

Surge con la idea de solucionar problemas que no se adaptan a funciones linealmente separables. Un MLP se compone de varias capas de perceptrones, a continuación, se explica los conceptos de perceptrón antes de continuar detallando el modelo MLP.

4.5.1.1. Perceptrón

El perceptrón es uno de los modelos más sencillos y comunes de neurona artificial, como se presenta en la Figura 11, esta neurona tiene asociada a cada entrada un peso que es un factor multiplicativo representado por w_1 , w_2 y w_3 . A la suma ponderada de las entradas se le aplica la función matemática de activación. Si el resultado supera un umbral preestablecido, la salida es 1 caso contrario la salida será 0. Este concepto se observa en la ecuación (2) donde, denotado por b , se define un sesgo equivalente al valor umbral con signo negativo [16].

$$\text{Salida} = \begin{cases} 0 & \text{si } f\left(\sum w \cdot x\right) - b \leq 0 \\ 1 & \text{si } f\left(\sum w \cdot x\right) - b > 1 \end{cases} \quad (9)$$

Este tipo de neuronas con una sola capa, es útil cuando tratamos problemas lineales como por ejemplo compuertas OR, AND, NOT, para modelar problemas más complejos que no son linealmente separables como por ejemplo la compuerta XNOR se requiere de mayor número de capas en el perceptrón o neurona.

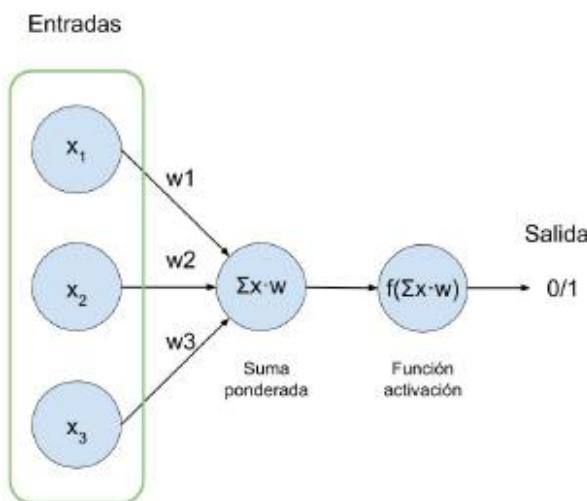


Figura 11. Neurona artificial o perceptrón (tomada de [16])

4.5.1.2. Perceptrón multicapa (MLP)

Es un modelo de red neuronal con alimentación hacia adelante, se compone por varias capas entre la entrada y la salida (véase Figura 12). Para entrenar este tipo de redes se suele emplear el algoritmo de backpropagation o retro propagación que consiste en la propagación de los valores de entrada hacia delante, se calcula el error a la salida entre

la salida esperada y la salida obtenida, y se propagan hacia atrás con el objetivo de actualizar los pesos mediante el gradiente de la función de error [19].

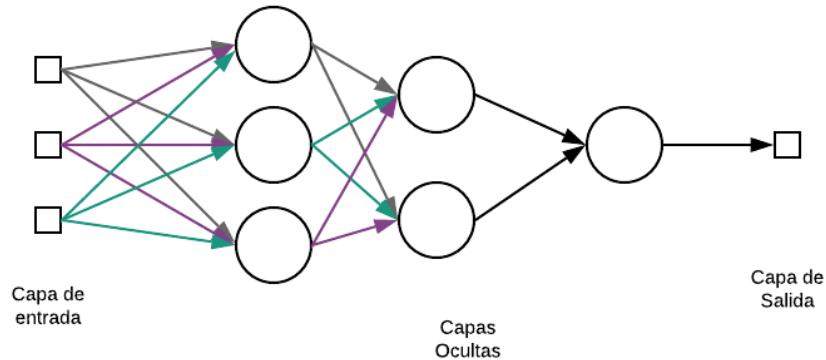


Figura 12. Red neuronal de perceptrón multicapa (Imagen propia)

Para calcular el error comúnmente se emplea la función del error cuadrático medio que se define en la ecuación (10), donde n es cada una de las iteraciones del aprendizaje. El perceptrón multicapa aprende encontrando el mínimo de la función de error.

$$e(n) = \frac{1}{2} \sum_{i=1}^{n_c} (s_i(n) - y_i(n))^2 \quad (10)$$

Para minimizar el error total se emplea métodos de gradiente estocástico, que consisten en minimizar el error en cada patrón $e(n)$. La ecuación (11) muestra el comportamiento del método de gradiente descendiente, cada peso de la red w se actualizará para cada patrón de entrada n , finalmente α representa la tasa de aprendizaje.

$$w(n) = w(n-1) - \alpha \frac{\partial e(n)}{\partial w} \quad (11)$$

4.5.2. Autoencoders apilados (SAE)

Este modelo emplea varios autoencoders entrenados uno a uno, usando cada autoencoder entrenado para entrenar el siguiente, con el objetivo de extraer características más complejas. Empezaremos exponiendo el concepto de autoencoder para proceder a la configuración de SAE.

4.5.2.1. Autoencoder (AE)

Es una arquitectura de red neuronal artificial muy parecida a la MLP (retroalimentadas), cuya principal diferencia se encuentra en la capa oculta, donde se presentan menos neuronas que en la capa de salida como se muestra en la Figura 13. Este tipo de arquitecturas pueden tener un mayor número de capas ocultas, pero manteniendo simetría (el número de capas suele ser impar). Este tipo de redes tiene como objetivo aprender la representación comprimida y distribuida de un conjunto de datos, dicho de otro modo, la salida es una reconstrucción de la entrada, pero al tener menos neuronas en las capas ocultas comprime y reduce la dimensionalidad de los datos. Para entrenar un autoencoder se emplea el algoritmo de backpropagation pero de manera no supervisada, por tanto, la red recibe los datos, los comprime y los regresa a sus dimensiones, los pesos se actualizan teniendo en cuenta que las salidas deseadas son las mismas entradas, en base al error global [10].

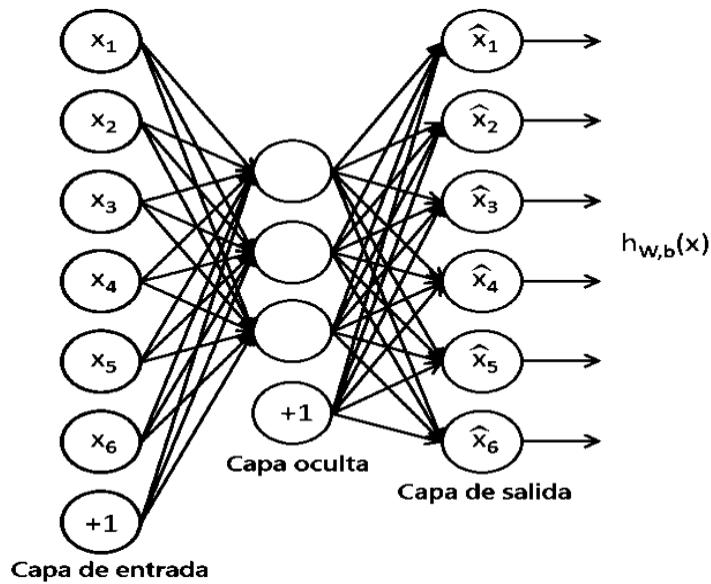


Figura 13. Arquitectura de un Autoencoder (tomada de [19])

4.5.2.2. Autoencoder Apilado (SAE)

Como su nombre lo indica esta red se compone de múltiples autoencoders en pila que proporcionan un método eficaz de pre entrenamiento. En la Figura 14, la capa oculta verde (t) actúa como la capa de entrada al autoencoder rojo ($t + 1$), por lo tanto, la capa de entrada al autoencoder t es la entrada a toda la red [11]. El proceso de entrenamiento de este tipo de red es:

- Entrena el primer autoencoder t (verde) de forma individual utilizando el método de propagación hacia atrás con todos los datos de entrenamiento disponibles.
- Para el segundo autoencoder $t + 1$ (rojo), se descarta la salida del autoencoder t , se entrena con una muestra de entrada a la capa de entrada de t , que se propaga hacia adelante a la capa de salida del autoencoder $t + 1$. Los pesos de $t + 1$ se actualizan empleando propagación hacia atrás. En resumen, se elimina la capa de salida del autoencoder anterior y se sustituye con otro autoencoder y se entrena con propagación hacia atrás, este proceso se repite para todas las capas.
- La red entra a un proceso de ajuste fino que consiste en añadir una o más capas enteramente conectadas con la última capa (amarilla) y se entrena empleando propagación hacia atrás.

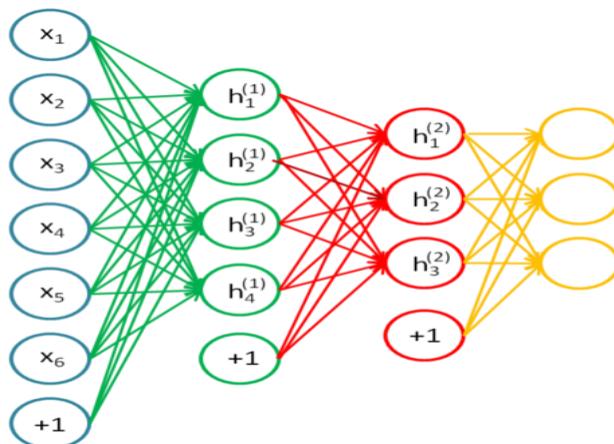


Figura 14. Arquitectura de un Autoencoder apilado (tomada de [20])

4.5.3. Redes de Creencia Profunda (DBN)

El DBN es una red multicapa profunda que incluye muchas capas ocultas, en la que cada par de capas conectadas es una RBM. De esta forma, un DBN se representa con una pila de RBMs.

4.5.3.1. Máquinas de Boltzmann Restringidas (RBM)

Es un tipo de red neuronal generativa, sus funciones de activación tienen un comportamiento no determinístico y pueden aprender una distribución de probabilidad sobre su propio conjunto de entradas. Se componen de una capa de sesgo (bias) oculta

(en la Figura 15 la capa celeste) y visible (capa roja) que a diferencia de las redes retroalimentadas las conexiones entre las capas visibles y ocultas pueden ser visible a oculta y oculta a visible, además de que se encuentran completamente conectadas, pero de manera ordenada entre la capa actual y la siguiente, es por ello que se denomina máquina de Boltzmann restringida [19].

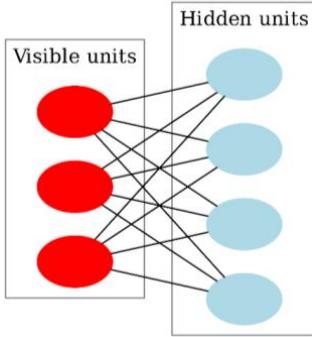


Figura 15. Arquitectura de red tipo Máquina de Boltzmann Restringida (tomada de [20])

4.5.3.2. Redes de Creencia Profunda (DBN)

Al igual que con los SA, las redes de creencia profunda apilan máquinas de Boltzmann, en este caso como se observa en la Figura 16, la capa oculta de la RBM en azul (t) actúa como la capa visible de la RBM ($t + 1$) y la capa de entrada de la primera RBM es la capa de entrada para toda la red. El procedimiento de entrenamiento es similar al de un SA, pero en lugar de autoencoders emplea RBM y la propagación hacia atrás es reemplazada por divergencia contrastiva, que es un algoritmo en el que las unidades ocultas y visibles se actualizan en paralelo para obtener una reconstrucción de los datos y se vuelve a actualizar las unidades ocultas de nuevo [11][19].

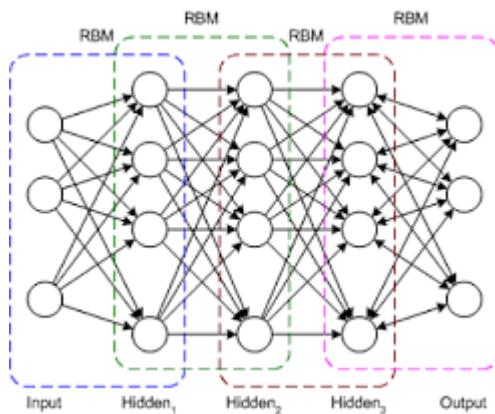


Figura 16. Arquitectura de una red de creencia profunda (tomada de [21])

4.5.4. Redes Convolucionales (CNN)

Este tipo de redes se adaptan eficazmente al reconocimiento de imágenes, son de tipo jerárquicas y al igual que una DBN extrae características en sus capas ocultas; el nombre de la red se deriva de su arquitectura, esta se conforma de capas convolucionales que extraen las características y capas de submuestreo que se encargan de la abstracción del objeto y una capa totalmente conectada que representa los datos de objetivo (véase Figura 17) [19][16].

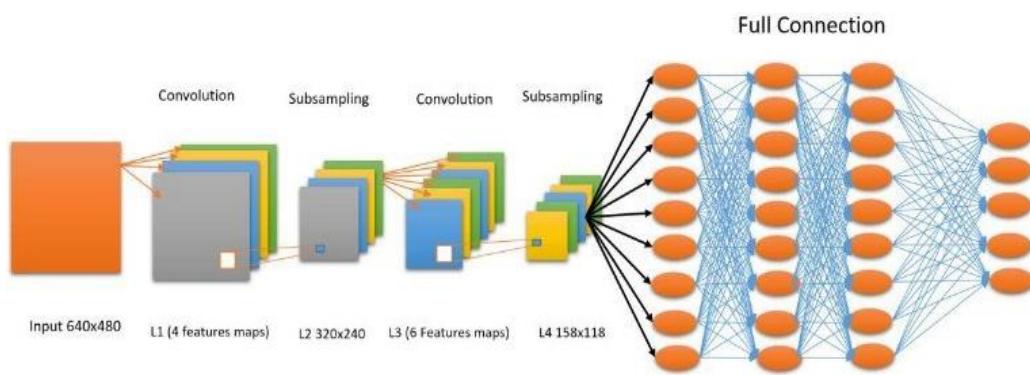


Figura 17. Arquitectura de una red neuronal convolucional (tomada de [22])

Estas redes presentan las siguientes propiedades:

- **Capa convolucional:** En esta capa se receptan los datos de entrada (por lo general una imagen) y sobre estos se aplica un filtro que devuelve un mapa de características, reduciendo el tamaño de los parámetros. La convolución proporciona herramientas para trabajar con tamaños variables, se basa en un operador matemático que transforma dos funciones f y g , en una tercera que representa la magnitud en que se superponen f y una versión trasladada y rotada de g .
- **Capa de reducción o pooling:** Se coloca después de la capa convolucional, se encarga de reducir la cantidad de parámetros a analizar, reduciendo las dimensiones y quedándose con las características más comunes, esta reducción también implica perdida de información; pero provoca menor sobrecarga en el cálculo de próximas capas y reduce el sobreajuste.
- **Capa clasificadora:** Una vez que la imagen ha pasado por las capas convolucionales, pooling y se han extraído sus características más destacadas, los datos llegan a la fase de clasificación. Para ello, emplean capas

completamente conectadas en las que cada pixel se trata como una neurona independiente. Las neuronas de esta fase funcionan como las de un perceptrón multicapa, donde la salida de cada neurona se calcula multiplicando la salida de la capa anterior por el peso de la conexión, y aplicando a este dato una función de activación [19].

4.5.5. Redes Long Short-Term Memory (LSTM)

Son del tipo de redes recurrentes, y surgen para solucionar el problema del enorme crecimiento del gradiente y su desvanecimiento con el tiempo, que se presenta al entrenar redes recurrentes y que dificulta la memorización de dependencias a largo plazo [11].

4.5.5.1. Redes Neuronales Recurrentes RNN

Una RNN es una red donde las funciones de transición y de salida f_h, f_o se definen de la siguiente manera:

$$\begin{aligned} h_t &= f_h(x_t, h_{t-1}) = \phi_h(\mathbf{W}^T h_{t-1} + \mathbf{U}^T x_t) \\ y_t &= f_o(h_t, x_t) = \phi_o(\mathbf{V}^T h_t) \end{aligned} \quad (12)$$

En donde las matrices \mathbf{W} es de transición, \mathbf{U} de entrada y \mathbf{V} de salida, las funciones usuales de las redes neuronales se definen por ϕ_h, ϕ_o y generalmente son de tipo sigmoide o de tangente hiperbólica. Entonces, los parámetros para esta red pueden ser estimados usando el gradiente estocástico de descenso (SGD) en donde el gradiente se calcula usando retro propagación a lo largo del tiempo [11].

4.5.5.2. Redes Long Short-Term Memory (LSTM)

Las redes LSTM aplican el mismo principio de las RNN, pero incorporan una serie de pasos para decidir qué información va a ser almacenada y cual borrada a través de tres compuertas que controlan el modo en que fluye la información (véase Figura 18):

- **Puerta de entrada:** Controla la entrada de nueva información a la memoria.
- **Puerta de olvido:** Decide que datos son importantes y cuales se descartan para dejar sitio a nuevos datos.
- **Puerta de salida:** Controla cuando se utiliza el resultado de los recuerdos almacenados en la celda que dispone de mecanismos de optimización basados en el error de salida de la red resultante [11].

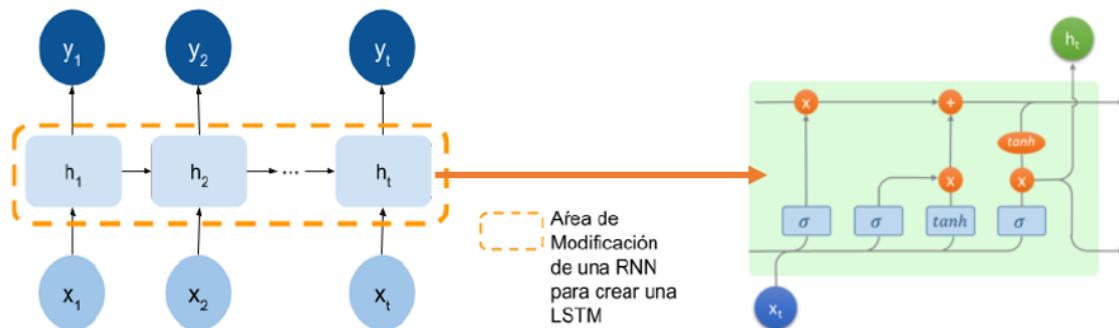


Figura 18. Arquitectura de red LSTM (tomada de [11])

4.6. TECNOLOGÍAS

4.6.1. Keras

Es una librería de redes neuronales desarrollada en Python, esta librería puede ejecutarse sobre Tensorflow o Theano (bibliotecas para el desarrollo de machine learning), se caracteriza por su modularidad, minimalismo y extensibilidad que permiten un prototipado fácil y rápido, soporta arquitecturas neuronales convolucionales, recurrentes, además de esquemas de conectividad arbitrarios. Según el sitio Web [23] actualmente Keras es compatible con las versiones 2.7-3.6 de Python y permite correr algoritmos en GPU (unidad de procesamiento gráfico) y CPU (unidad central de procesamiento).

La librería Keras se compone de: modelos, capas, inicializadores, funciones de pérdida y optimizadores, los cuales se desarrollan a continuación [23]:

4.6.1.1. Modelos

Hay dos tipos de modelos disponibles para implementar en Keras: el modelo secuencial y la clase `Model`, ambos presentan varios métodos en común como `summary()`, `get_config()` o `to_json()` que devuelven información básica sobre el modelo y los modelos `getter` y `setter` propios de los pesos del modelo.

- **Modelo Secuencial:** Los métodos básicos del modelo secuencial son:
 - `compile`: Calcula automáticamente la función gradiente y prepara el modelo para el entrenamiento.
 - `fit`: Entrena el modelo para un número de ciclos determinado por `(nb_epoch)`, actualiza gradiente cada `n` ejemplos valor que se determina por `batch_size`.
 - `evaluate`: Calcula la función de pérdida lote por lote.
 - `predict`: Genera predicciones para unos valores de entrada

4.6.1.2. Capas

La librería Keras presenta las siguientes funciones para diseñar las capas de la red neuronal:

- Funciones básicas de las capas:
 - `Dense`: Capas regulares totalmente conectadas.
 - `Activation`: Aplica una función de activación a la salida de las capas. Las principales que ofrece Keras son: `softmax`, `softplus`, `softsign`, `relu`, `tanh`, `sigmoid`, `hard_sigmoid`, `linear`.
 - `Dropout`: Establece al azar una fracción de unidades de entrada a 0 en cada actualización durante la fase de entrenamiento, para evitar el sobreajuste.
 - `Reshape`: Transforma la salida en una forma concreta
- Capas convolucionales: Keras presenta diversas funciones para desarrollar capas convolucionales según las dimensiones de su entrada.
- Capas recurrentes: Existen 3 tipos de capas recurrentes: `simpleRNN`, `LSTM`, `GRU`.

4.6.1.3. Inicializadores

Los inicializadores definen la manera de inicializar los pesos de las capas, a través del argumento `init`, los más comunes son: uniform, normal, identity, zero, glorot_normal, glorot_uniform.

4.6.1.4. Funciones de pérdida

Es uno de los parámetros necesarios para compilar un modelo, Keras proporciona las siguientes funciones:

- `mean_squared_error(y_true, y_pred)`: Calcula el error cuadrático medio.
- `mean_absolute_error`: Calcula el error medio absoluto.
- `binary_crossentropy`: Calcula la pérdida logarítmica.
- Regularizadores: permiten aplicar penalizaciones en las capas durante la optimización.

4.6.1.5. Optimizadores

Es otro de los valores necesarios para compilar un modelo, los principales son:

- `SGD`: es un método de aproximación estocástica del método de descenso de gradiente.
- `Adagrad`: algoritmo de gradiente que adapta la ratio de aprendizaje a los parámetros.
- `Adadelta`: es una extensión del algoritmo Adagrad.
- `Adam`: Se basa en el gradiente de primer orden de funciones estocásticas objetivas, este método es fácil de implementar, eficiente en consumo de recursos computacionales.

Toda la información aquí recopilada se encuentra disponible en: www.keras.io/ documentación oficial de la librería Keras.

4.6.2. Python

Python es un lenguaje de programación que cuenta con estructuras de datos eficientes y de alto nivel. Hasta la fecha¹ Python se encuentra en la versión 3.6. La biblioteca estándar está a libre disposición para las principales plataformas desde el sitio Web de Python, <https://www.python.org/>. El mismo sitio contiene distribuciones, enlaces de módulos libres de Python, programas, herramientas y documentación.

4.6.3. Jupyter

Jupyter es un entorno de trabajo interactivo que permite desarrollar códigos en Python de manera dinámica, esta herramienta integra en un mismo documento bloques de código, texto, gráficas e imágenes. Comúnmente Jupyter es utilizado en análisis numérico, estadística y ML, entre otros campos de la informática y las matemáticas. La documentación, así como enlaces a módulos y programas se encuentra disponible en: <https://jupyter.org/>.

¹ Fecha: 24/04/2019

5. Materiales y Métodos

En esta sección, se describe por cada objetivo los métodos y recursos empleados en la ejecución del Trabajo de Titulación.

5.1. Objetivo 1:

Aplicar la técnica de revisión sistemática cualitativa de Bárbara Kitchenham para obtener los avances y conocimientos actuales de aplicaciones del Deep Learning en diferentes áreas de conocimiento

Para resolver este objetivo se utilizó el protocolo de revisiones sistemáticas de Bárbara Kitchenham [2], basado en las preguntas de investigación, los criterios de inclusión-exclusión, estrategia de búsqueda, evaluación de calidad y extracción y síntesis de datos, en la Figura 19, se especifican las etapas empleadas.

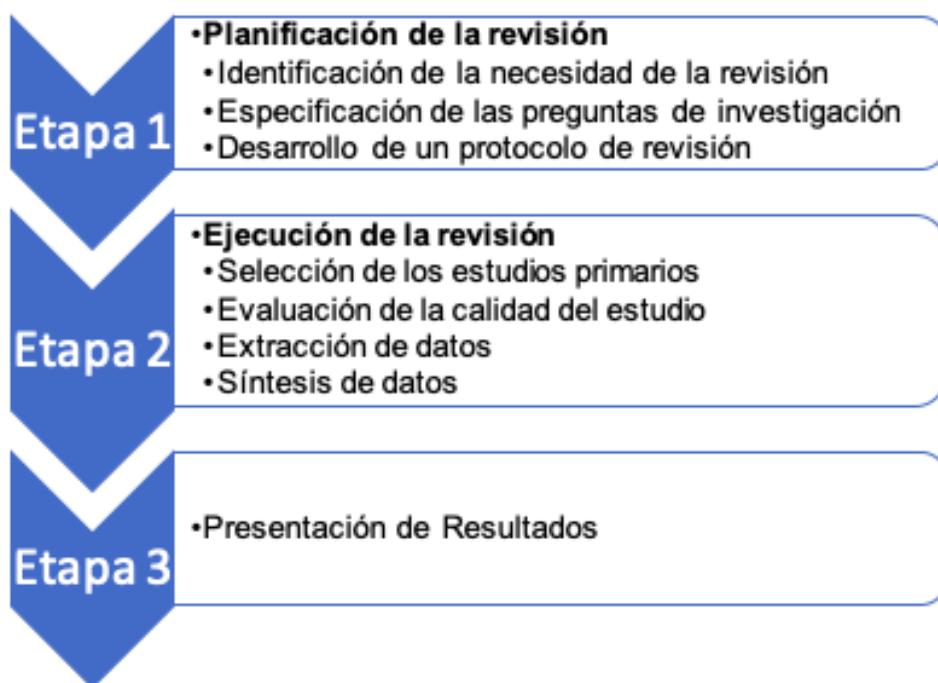


Figura 19. Etapas de la SLR según Kitchenham (Imagen propia)

5.1.1. Planificación de la Revisión

5.1.1.1. Identificación de la necesidad de revisión

En esta etapa se buscó trabajos cuyo objetivo se asemeje al planteado en esta investigación o que responda a las preguntas de investigación planteadas para determinar si existen revisiones de literatura acerca de la aplicación del DL, además se revisó si existen variedad de estudios que apliquen DL para determinar si es pertinente este estudio.

5.1.1.2. Desarrollo del protocolo de investigación

Se desarrolló el protocolo de revisión para reducir la posibilidad de sesgos, en el protocolo se incluye las preguntas de investigación, fuentes y cadenas de búsqueda, criterios de inclusión y exclusión (véase Resultados sección 1.2).

5.1.2. Ejecución de la Revisión

5.1.2.1. Selección de estudios primarios

Todo el protocolo de revisión y selección de los estudios primarios fue ejecutado siguiendo el algoritmo de la Figura 20, para facilitar el manejo de la bibliografía se empleó la herramienta Mendeley (<https://www.mendeley.com>) (véase Resultados sección 1.3).

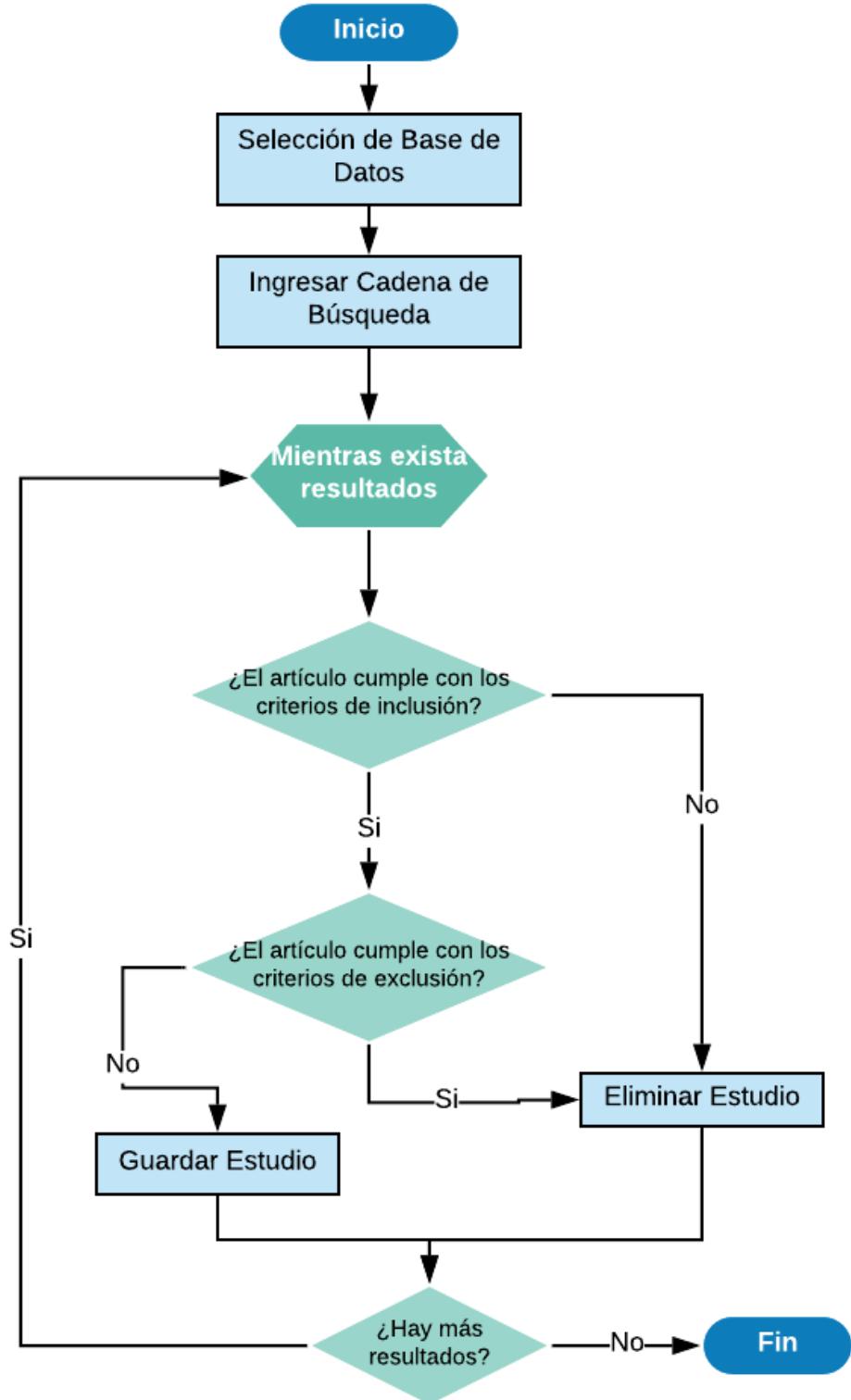


Figura 20. Proceso de selección de estudios primarios (Imagen propia)

5.1.2.2. Evaluación de Calidad

Se planteó criterios para determinar que estudios deben ser excluidos cuando no cumplen con un nivel mínimo de calidad (véase Resultados sección 1.4).

5.1.2.3. Extracción y síntesis de datos

La extracción y síntesis de datos permite registrar y recopilar con precisión la información de los estudios primarios. La Tabla I se formuló para extraer soluciones a las preguntas de investigación (PI), se incluyó datos generales sobre el documento (título, nombre del autor, año de publicación). En la síntesis de datos cada PI se evaluó por separado, al ser una investigación cualitativa se emplearon tablas relacionales, gráficas y análisis de agrupación que nos permitió agrupar los estudios que más se asemejan, considerando como parámetros las preguntas de investigación (véase Resultados secciones 1.5 y 1.6).

TABLA I.
EXTRACCIÓN DE DATOS

#	Descripción	Detalle	
1	Información bibliográfica	Título	Nombre del estudio
		Autor	Nombre del autor o autores
		Referencia	Número referencia correspondiente a la bibliografía
		Año	En el que fue publicado
2	Aplicación	Objetivo principal del documento seleccionado	
3	Modelos	Modelos del DL empleados para procesar los datos	
4	Funciones	Funciones empleadas en el modelo DL: función de activación, funciones adicionales en el entrenamiento	
4	Conclusiones Relevantes	Perspectiva personal del documento	

5.2. Objetivo 2

Demostrar el uso de Deep Learning con el resultado de la revisión sistemática cualitativa de Bárbara Kitchenham.

Los resultados del objetivo 1 condujeron a la generación de un modelo para pronóstico de ventas. En la SLR se encontró que el algoritmo de mayor aplicación al pronóstico de ventas fue el modelo LSTM; para lo cual se trabajó el problema de pronóstico de ventas como una serie temporal y siguiendo los pasos generales para la construcción de modelos con ML [7] descritos en la revisión de literatura (véase Figura 21).

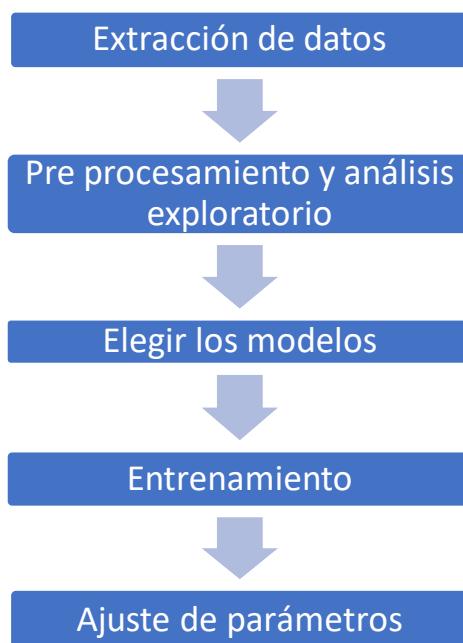


Figura 21. Etapas de la construcción del modelo ML (Imagen propia)

5.2.1. Extracción de datos

Para la extracción de datos se obtuvo los datos de ventas de la Industria Lojana de Especerías (ILE) (véase Anexo 8), el dataset pesa 800 MB, contiene las ventas de las fechas 01-03-2011 hasta 28-02-2018 de un total de 443 productos y cada uno contiene instancias diarias de ventas para cada producto (véase Tabla II).

TABLA II.
FRAGMENTO DEL DATASET VENTAS DE ILE

Fecha	Artículo	Venta
01/03/2011	1330-085	2,57
01/03/2011	1330-042	4,80
01/03/2011	1330-514	4,80
01/03/2011	1350-001	45,12
01/03/2011	1330-002	126,00
01/03/2011	1350-004	264,00
03/03/2011	1330-270	1,93
03/03/2011	1330-504	2,14
03/03/2011	1330-031	2,23
03/03/2011	1330-515	2,32
03/03/2011	1330-514	4,80
.....

5.2.2. Pre procesamiento y análisis exploratorio

En esta etapa se realizó un análisis exploratorio para seleccionar características y determinar las correlaciones entre las distintas características seleccionadas. Se balanceó los datos mediante varias operaciones como estabilización de varianza, estacionalidad, normalización, depuración y se separan en dos grupos: uno para entrenamiento y otro para evaluación del modelo (véase Resultados sección 2.1).

Los datos de ILE se trabajaron como una serie de tiempo que, es una secuencia de datos medidos en determinados momentos y ordenados cronológicamente [24]. Cuando se modelan series de tiempo, es conveniente aplicar algunas transformaciones que permitan tres cosas:

- **Estabilizar la varianza:** se aplicó la función *ln* a todos los datos.
- **Hacer estacionarios los datos:** para ello se debe lograr que la varianza, media y covarianza sean constantes. Hay varias técnicas para lograr esto, en este caso se aplicó diferencias.
- **Quitar la tendencia:** La tendencia muestra la conducta de los datos, para poder realizar predicciones se debe eliminar la tendencia en el entrenamiento para evitar conductas determinadas. Generalmente al hacer los datos estacionarios la tendencia se elimina.

Estas transformaciones se realizaron en el lenguaje de programación R, con las líneas de código:

```
library (ts)
plot(diff(log(ts(dataset))))
```

Donde:

- **ts**: Convierte la data a serie de tiempo.
- **log**: Aplica logaritmo natural a los datos del arreglo.
- **diff**: Esta función toma desde la segunda instancia y le resta el anterior, al finalizar esta función se reduce una instancia.
- **plot**: Nos devuelve la gráfica de los datos transformados

Para entrenar los modelos fue necesario normalizar las variables de entrada, el método empleado fue el escalado de variables (véase Ecuación 13), que normaliza los datos en los límites definidos, en este caso (0,1).

$$X_{std} = \frac{(X - X.\min)}{(X.\max - X.\min)} \quad (13)$$

En Python se logró mediante:

```
1. from sklearn.preprocessing import MinMaxScaler
2. scaler = MinMaxScaler(feature_range=(0, 1))
3. dataset = scaler.fit_transform(data)
```

Una vez que los datos están listos para aplicar el modelo, se procedió a decidir algún enfoque para seleccionar los conjuntos de entrenamiento y prueba. Se tomaron los primeros datos para el entrenamiento y los últimos para pruebas. La distribución fue la usual 80% y 20 %. Es decir, los primeros 2,044 datos para el entrenamiento y los restantes para prueba.

Finalmente se transformó el problema de series de tiempo a uno de regresión, para ello fue necesario ajustar la data agregando variables retardadas (lagged) empleando la ecuación 14, donde n es el tamaño del lag o cantidad de variables retardadas:

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_n y_{t-n} \quad (14)$$

Una función en Python para hacer esto es:

```
1. import numpy as np
2. import pandas as pd
3. def create_dataset(dataset, look_back=1):
4.     dataX, dataY = [], []
5.     for i in range(len(dataset)-look_back-1):
6.         a = dataset[i:(i+look_back), 0]
7.         dataX.append(a)
8.         dataY.append(dataset[i + look_back, 0])
9.     return np.array(dataX), np.array(dataY)
```

Esta función tiene como parámetro el número de lags (look_back) a considerar y regresa por un lado los predictores (variables retrasadas) y por otro la variable objetivo.

5.2.3. Elegir los modelos

En base al problema se seleccionó modelos basados en el resultado de la SLR. Para realizar la comparación; en los tres modelos se empleó el mismo dataset y la medida de error RMSE (raíz cuadrada del error cuadrático medio) la cual se expresa en la ecuación 15 (véase Resultados sección 2.2).

$$RMSE = \sqrt{MSE} = \sqrt{E[(y - \hat{y})^2]} \quad (15)$$

5.2.4. Entrenamiento y ajuste de parámetros

No se sabe a priori cual es la combinación adecuada de parámetros para un modelo DL y no se encontró algún método que optimice el cálculo de los mismos, por lo que se obtuvieron a través de prueba y error y se tomaron en cuenta los siguientes parámetros (véase Resultados sección 2.3):

- **Lag (n):** Ventana de retraso, es decir cuántas variables considerar para predecir las variables objetivo.
- **Dropout (p):** Técnica para evitar el overfitting que ocurre en las redes profundas. Esta técnica consiste en desechar aleatoriamente unidades y sus conexiones durante el entrenamiento. El parámetro $0 \leq p \leq 1$ se refiere al porcentaje de unidades desechadas. Se busca valores distintos de los extremos, si $p = 0$ no se desecha nada (no usamos dropout); y si $p = 1$ se está desechando todas las unidades, es decir, la capa no tendría unidades.

- **Stateful (True / False):** Se refiere al uso del último estado de cada muestra, es decir, dicta que tanta memoria tiene la red entre batches. Stateful es del tipo booleana, por tanto, si stateful = true indica que si se tiene memoria.
- **Número de bloques por capa:** Se refiere al número de bloques considerados en cada capa de la LSTM.
- **Tamaño del batch:** Se refiere al tamaño B del lote $\{i_1, i_2, \dots, i_B\}$ del gradiente estocástico, entre mayor sea el valor de B la red llegará al óptimo en menos iteraciones, pero el costo por cada iteración es más alto.
- **Número de épocas:** Se refiere al número de veces que los datos pasarán por el modelo.
- **Función de activación:** Se refiere a la función de activación empleada por las neuronas, las que se probó fueron: *relu*, *tanh*, *sigmoid*, *hard_sigmoid*, *linear*.
- **Número de capas:** Se refiere al apilamiento de módulos LSTM para agregar mayor profundidad al modelo.

5.2.5. Hardware y software

Se detalla el hardware y software empleado en el desarrollo de los experimentos.

5.2.5.1. Hardware

Dado que el conjunto de datos final no es demasiado grande no es necesario el uso de GPU. Las características del hardware utilizado son las siguientes:

- Memoria: 8GB
- Procesador: Intel Core i7 (2,2 GHz)
- Gráficos: Intel HD Graphics 6000
- Sistema Operativo: MacOS Mojave
- Disco sólido: 500 GB

5.2.5.2. Software

El software utilizado para hacer el modelo DL LSTM fue Keras, el cual se abordó en la revisión de literatura, se trabajó sobre Tensorflow con Python 3 y el entorno de trabajo Jupyter. Para comparar el modelo DL se hicieron dos modelos tradicionales una red neuronal tipo perceptrón de una capa, que se desarrolló en Keras y el modelo de regresión ARIMA para el cual se empleó la herramienta R en la versión 1.1.423.

R es un lenguaje y un entorno para la computación estadística y los gráficos, la documentación se encuentra disponible en <https://www.r-project.org>.

5.3. Objetivo 3

Escribir los resultados logrados en un artículo para su envío a una revista indexada e inicie su proceso de publicación.

Para cumplir con este objetivo se redactó un artículo de revisión y se envió al Simposio Iberoamericano en Programación Informática (SIIPRIN'2018) y Congreso Internacional de Tecnología Educativa y Gestión del Conocimiento (CITEGC'2018). La guía para la redacción del artículo se basó en “El proceso de escritura y publicación de un artículo científico” expuesta por [25]:

5.3.1. Seleccionar la revista

En este caso se eligió un congreso, por tanto, los resultados se publicaron como memoria de congreso. SIIPRIN'2018 realizó el llamado a trabajos en septiembre del 2018, este congreso aceptaba tres tipos de manuscritos: resultados de investigación, investigaciones en curso, estudios de caso y revisiones sistemáticas de literatura; se optó por desarrollar un artículo de revisión sistemática de literatura.

De la SLR ejecutada en este proyecto, se seleccionó los resultados relacionados al área financiera, en correlación al objeto modelo de demostración LSTM, así como estudios primarios relacionados a esta área. En el sitio Web del congreso <http://siiprin.esepoch.edu.ec/index.html>, se especificó la plantilla que se empleó para redactar el artículo.

5.3.2. Redacción e inicio del proceso de publicación

Revisada la redacción, fluidez, ortografía y referencias del manuscrito final (Anexo 4), se procedió a la redacción de la carta de presentación y al envío del artículo para su revisión a través de la plataforma EasyChair para SIIPRIN'2018 (véase Resultados sección 3).

6. Resultados

En este apartado, se exponen los resultados y evidencias obtenidos por cada objetivo, el objetivo 1 presenta los resultados de la SLR incluyendo las tablas de extracción de datos para cada estudio primario; en el objetivo 2 se muestra los resultados obtenidos por cada experimento realizado en la búsqueda de los parámetros adecuados para los modelos LSTM y red neuronal de una capa, igualmente se muestra el resultado del modelo ARIMA; los resultados del artículo enviado al SIIPRIN'2018 se presentan en el objetivo 3.

6.1. Objetivo 1:

Aplicar la técnica de revisión sistemática cualitativa de Bárbara Kitchenham para obtener los avances y conocimientos actuales de aplicaciones del Deep Learning en diferentes áreas de conocimiento

6.1.1. Identificación de la necesidad de una revisión.

Considerando las mejoras alcanzadas en el procesamiento de datos gracias al DL, se decide realizar una SLR con el fin de encontrar toda la documentación posible acerca de la aplicación del DL, que nos ayude a obtener aspectos relevantes que contribuyan con el objetivo planteado para esta revisión:

“Determinar los modelos y aplicaciones de Deep Learning que se usan en las diferentes áreas de conocimiento para resolver los diversos problemas de investigación”

6.1.2. Desarrollo de un protocolo de Revisión.

6.1.2.1. Especificación de las preguntas de investigación

En base al planteamiento anterior, en la Tabla III se muestra las preguntas de investigación que sirvieron para cumplir con este propósito.

TABLA III.
PREGUNTAS DE INVESTIGACIÓN

ID	Pregunta de Investigación
PI_1	¿Cuáles son las áreas de conocimiento en donde se aplica DL?
PI_2	¿Qué aplicación del DL existe en las diferentes áreas de conocimiento?
PI_3	¿Qué modelos o técnicas del DL se emplean para resolver los problemas de investigación?

6.1.2.2. Selección de las fuentes de búsqueda.

Siguiendo los lineamientos de Kitchenham [2], en la primera etapa se consideraron las fuentes de búsqueda indicadas en la Tabla IV, las cuales fueron seleccionadas basados en la accesibilidad y admisión de consultas avanzadas.

TABLA IV.
FUENTES DE BÚSQUEDA

Fuente de Búsqueda	Dirección Web
SCIENCECIRECT	https://www.sciencedirect.com/
SCOPUS	https://www.scopus.com/
IEEE XPLORER	http://ieeexplore.ieee.org/
GOOGLE SCHOLAR	https://scholar.google.com

6.1.2.3. Palabras claves y cadenas de búsqueda

Se seleccionaron a partir de una revisión preliminar de información en artículos, libros y ponencias. Estas son:

Deep Learning, Machine learning, Deep Neural Network, Aprendizaje Profundo, Algorithms, models, IA.

Una vez definidas las bases de datos e identificadas las palabras claves se realizó las consultas posibles utilizando los operadores lógicos **AND / OR**, generando las cadenas de búsqueda que se muestran en la Tabla V.

TABLA V.
CADENAS DE BÚSQUEDA

Fuente de Búsqueda	Identificación	Cadena de búsqueda
GOOGLE ACADÉMICO	CB01	"deep learning" OR "Deep neural network" "Aprendizaje profundo" "Machine learning" OR "IA" Algorithms OR models
IEEE	CB02	(("deep learning" OR "aprendizaje profundo") AND "Deep neural network" AND ("machine learning" OR "IA") AND (Algorithms AND models)) Filters Applied: 2012 - 2018
SCIENCEDIRECT	CB03	(("deep learning" OR "aprendizaje profundo") AND "Deep neural network" AND ("machine learning" OR "IA") AND (Algorithms AND models)) Year: 2012-2018 Title: "deep learning"
SCOPUS	CB04	TITLE-ABS-KEY (("deep learning" OR "aprendizaje profundo") AND "deep neural network" AND ("Machine Learning" OR "IA") AND ("Algorithms" AND "models")) AND PUBYEAR > 2011 AND (LIMIT-TO (LANGUAGE , "English") OR LIMIT-TO (LANGUAGE , "Spanish"))

6.1.2.4. Criterios de inclusión y exclusión.

Los criterios de inclusión y exclusión fueron clave para la selección de estudios primarios. Los criterios de inclusión se plantearon con el propósito de extraer documentos que aborden la aplicación del DL (véase Tabla VI).

TABLA VI.
CRITERIOS DE INCLUSIÓN

ID	Criterios de Inclusión
CI_1	Artículos científicos publicados en revistas o congresos
CI_2	Estudios publicados a partir del 2012.
CI_3	Estudios que en el título o resumen contenga las palabras claves.
CI_4	Estudios cuyo título tenga relación con el tema de investigación
CI_5	Estudios citados
CI_6	Estudios en español e inglés

Los criterios de exclusión sirvieron para descartar aquellos estudios que no fueron relevantes con el objetivo planteado (véase Tabla VII)

TABLA VII.
CRITERIOS DE EXCLUSIÓN

ID	Criterios de Exclusión
CE_1	Estudios que no sean de carácter experimental
CE_2	Trabajos mal estructurados y poco claros.
CE_3	Publicaciones informales que no siguen una metodología científica.
CE_4	Estudios que no contribuyan a las preguntas de investigación
CE_5	Estudios duplicados
CE_6	Todos los que no cumplen con los criterios de inclusión.

6.1.3. Selección de estudios primarios

TABLA VIII.
RESULTADOS DEL PROCESO DE SELECCIÓN DE ESTUDIOS PRIMARIOS

Fuente de búsqueda	Cadena de búsqueda	Estudios incluidos	Estudios excluidos	Estudios primarios
GOOGLE ACADÉMICO	CB01	51	38	13
IEEE	CB02	29	17	12
SCIENCEDIRECT	CB03	38	20	18
SCOPUS	CB04	72	51	21
TOTAL		190	126	64

Después de aplicar los criterios de inclusión expuestos en la Tabla VI se obtuvo como resultado 190 artículos. Seguidamente para seleccionar los estudios primarios se aplicó los criterios de exclusión visibles en la Tabla VII, que dieron como resultado 126 estudios excluidos, quedando un total de 64 estudios primarios analizados en esta revisión (véase Tabla VIII), en la Figura 22 se expresó el número de artículos seleccionados de acuerdo a la base de datos científica, donde se puede ver que la mayor parte de estudios primarios pertenecen a ScienceDirect y Scopus.

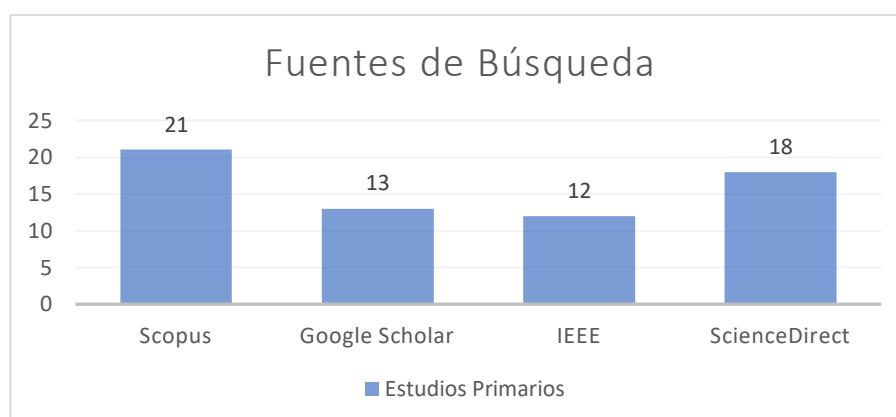


Figura 22. Estudios primarios seleccionados en cada base de datos científica

6.1.4. Evaluación de Calidad.

Se revisó los criterios de evaluación de calidad descritos en el estudio realizado por Kitchenham [8], de los cuales se consideraron los expuestos en la lista de verificación (véase Tabla IX) para la selección de los estudios primarios y dirección de la SLR.

TABLA IX.

LISTA DE VERIFICACIÓN DE EVALUACIÓN DE CALIDAD

#	Preguntas	Si	No
Para la selección de estudios primarios:			
1	Cumple con los criterios de inclusión	✓	
2	Se revisó el documento para verificar que no cumple con los criterios de exclusión	✓	
3	Los estudios seleccionados son experimentales	✓	
4	El autor o autores sustentan el problema de investigación	✓	
5	El autor o autores justifican el uso del DL en el problema de investigación.	✓	
6	El estudio realiza una comparación imparcial de varios modelos ML o estadísticos para validar la solución.	✓	
7	Los datos empleados en los experimentos se eligieron acorde al problema de investigación y pertenecen a fuentes reales.	✓	
Para la ejecución del Trabajo de Titulación:			
1	¿El documento tiene como objetivo identificar los modelos DL que se aplican en las diferentes áreas de conocimiento?	✓	
2	¿Está el documento sesgado hacia un algoritmo, modelo o técnica del DL?		✓
3	¿Los hallazgos abordan las preguntas de investigación originales?	✓	

6.1.5. Extracción de Datos.

Luego de haber ejecutado el protocolo de la SLR, se obtiene un total de 64 estudios primarios. Empleando la Tabla I (en la metodología) se extrajo la información relevante para este estudio, los resultados se muestran en el **Anexo 1** en las Tabla XXIII hasta la Tabla LXXXVI. Los resultados se ordenaron por el área de conocimiento y se identifican mediante el código EP (estudio primario), seguido del número correspondiente.

6.1.6. Síntesis de datos

En la síntesis de datos se presenta los resultados de la SLR de manera más concreta, contestando a las preguntas de investigación con el objetivo de que los resultados sean más legibles, en el **Anexo 2** se adiciona una tabla que integra la información de las tres preguntas de investigación.

6.1.6.1. Vista general de los estudios seleccionados

En la Figura 23 se sintetiza el proceso de la SLR para la selección de estudios primarios, indicando los resultados obtenidos en cada etapa, se explica que las búsquedas produjeron 18111 artículos de los cuales se obtuvo 631 tras aplicar los filtros de palabras claves, resumen y título, aplicando los criterios de inclusión y eliminando artículos repetidos quedaron un total de 190 estudios para revisión, aplicando los criterios de exclusión se redujeron a 64 estudios seleccionados.

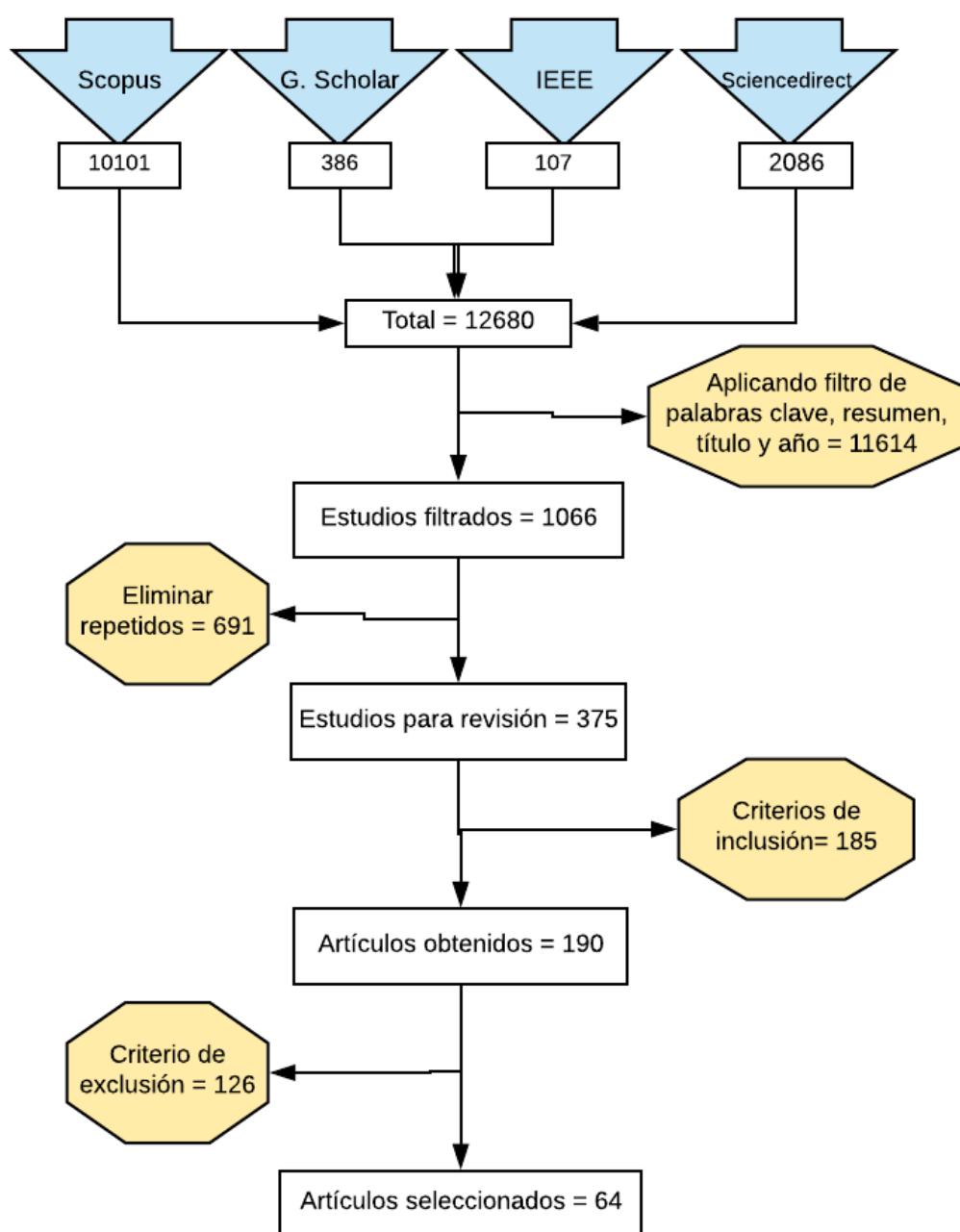


Figura 23. Síntesis del proceso de selección de estudios primarios.

En la Figura 24 se presenta la cantidad de estudios seleccionados por año de publicación, se observa que los estudios más antiguos encontrados fueron del 2012 y que existe un crecimiento de investigaciones con respecto al tema abordado notando que en el año 2017 obtuvo su mayor acogida, la reducción de resultados en el año 2018 puede corresponder a que las últimas consultas se realizaron a mediados de ese año.

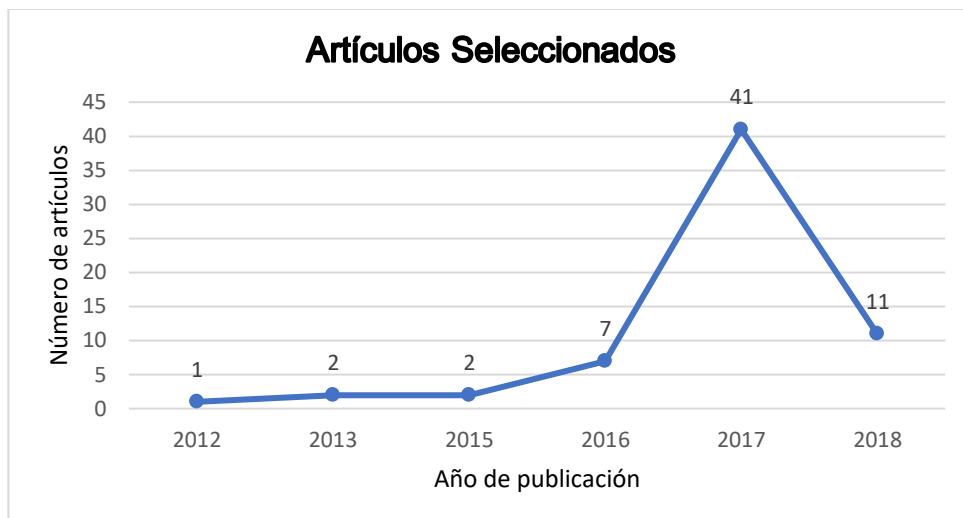


Figura 24. Resultado de artículos seleccionados por año de publicación

Todos los estudios seleccionados son de carácter experimental, la Tabla X presenta una visión general de los artículos seleccionados.

TABLA X.
TRABAJOS SELECCIONADOS POR CADA BASE DE DATOS CIENTÍFICA

#	Base de datos científica	Cadena de Búsqueda	No. resultados	Resultado de búsqueda: Estudio primario.
1	Google Académico	CB01	13	EP01, EP02, EP18, EP19, EP31, EP42, EP22, EP43, EP48, EP53, EP61, EP62, EP63
2	IEEE	CB02	12	EP15, EP16, EP17, EP20, EP21, EP25, EP29, EP30, EP33, EP39, EP59, EP60
3	ScienceDirect	CB03	18	EP04, EP06, EP07, EP08, EP09, EP10, EP11, EP12, EP13, EP27, EP26, EP34, EP45, EP46, EP52, EP54, EP55, EP64
4	Scopus	CB04	21	EP03, EP05, EP14, EP23, EP24, EP28, EP32, EP35, EP36, EP37, EP38, EP40, EP41, EP44, EP47, EP49, EP50, EP56, EP51, EP57, EP58

6.1.6.2. Síntesis PI_1: ¿Cuáles son las áreas de conocimiento en donde se aplica DL?

La Tabla XI presenta las áreas de conocimiento detectadas; se consideró dentro de ingeniería todo estudio relacionado con la robótica, visión artificial, etc. En salud se incluyó estudios que se relacionaron con detección de enfermedades, nutrición y otros artículos asociados a temas de salud humana. En finanzas, estudios cuyos problemas se relacionaron con temas financieros como: banca, bolsa de valores, impuestos, ventas, etc. En investigación se consideró estudios que emplearon modelos DL para resolver problemas como por ejemplo detección de CO₂. Finalmente, en industria se encontró estudios que se relacionan con el campo industrial.

La Figura 25 visualiza que el área de mayor aplicación del DL es ingeniería y el área de menor aplicación es industria, estos resultados pueden corresponder a que DL es un área relativamente nueva de estudio que tuvo mayor acogida en el campo de la visión artificial.

TABLA XI.
TRABAJOS SELECCIONADOS POR CADA ÁREA DE CONOCIMIENTO

#	Área	Estudios Primarios	# Artículos
1	Ingeniería	EP01 - EP25	25
2	Salud	EP26 - EP27	12
3	Finanzas	EP28 - EP32	20
4	Investigación	EP33 - EP44	5
5	Industria	EP45 - EP64	2

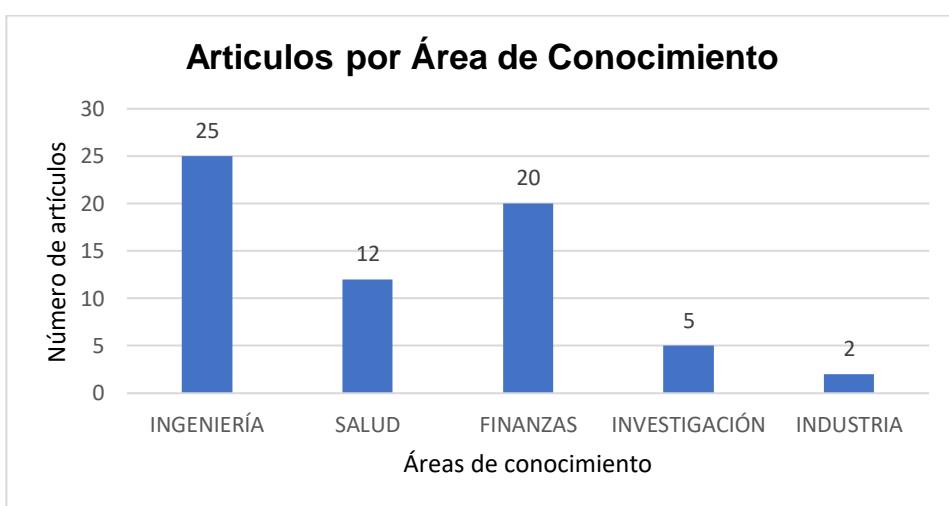


Figura 25. Resultado de artículos seleccionados por área de conocimiento

6.1.6.3. Síntesis PI_2: ¿Qué aplicación del DL existe en las diferentes áreas de conocimiento?

Todos los artículos se analizaron para descubrir: ¿Qué problema de investigación abordan los autores? Los resultados se muestran en la Tabla XII en la que se visualiza las aplicaciones encontradas por cada área de conocimiento, seguidas del identificador correspondiente al que pertenecen.

TABLA XII.

IDENTIFICACIÓN DE APLICACIONES DEL DL EN LAS DIFERENTES ÁREAS DE CONOCIMIENTO (PI_2)

Área	Aplicación		Estudios Primarios	
Ingeniería	Visión Artificial	Reconocimiento de objetos	EP01, EP02, EP03, EP07, EP08, EP09, EP10, EP12, EP13, EP14, EP16, EP18	
		Robótica	EP20	
		Emociones o expresiones faciales	EP04, EP21	
		Reconocimiento de personas	EP11, EP15	
	Procesamiento del lenguaje Natural		EP06	
	Clasificación de servicios de nube		EP17	
	Seguridad de la información	Detección de ciberataques	EP19	
		Detección de amenazas de seguridad	EP22	
		Detección de spam	EP05	
	Clasificación de actividad humana		EP23	
	Análisis de sentimientos		EP24	
	Detectar calidad de video		EP25	
Industria	Detectar calidad de un producto (lechuga)		EP26	
	Recomendación de hoteles		EP27	
Investigación	Detectar ecuaciones		EP28	
	Diagnóstico de rotor		EP29	
	Detectar grietas		EP30	
	Visión Artificial		EP31	
	Captura de CO2		EP32	
	Diagnóstico médico	Detección de tuberculosis	EP33	
Salud		Detección de cáncer de pulmón	EP35	
		Detectar cáncer de páncreas	EP39	
		Detectar la gravedad de osteoartritis de rodilla	EP41	
		Detectar cáncer de mama	EP42	
		Detectar displasia cervical	EP43	
		Detectar pacientes con epilepsia	EP44	
		Clasificación de grabaciones de EEG	EP34	
		Detectar la densidad nutricional de los purés	EP36	
		Identificación de los glóbulos blancos	EP37	
	Etiquetar genes en vías de cáncer		EP38	
	Predecir el comportamiento de la cinética molecular en proteínas y fármacos		EP40	

Finanzas	Análisis del mercado de valores	EP45, EP47, EP48, EP49, EP51, EP52, EP60, EP63, EP64
	Detectar riesgo financiero	EP46
	Pronóstico del precio del petróleo	EP50
	Predecir comportamiento del inversor	EP53
	Detección de evasores de impuestos	EP54
	Detección de fraudes de seguros	EP55
	Predicción del rendimiento corporativo	EP56
	Predicción de ventas	EP57, EP58, EP59, EP61
	Detección de riesgo hipotecario	EP62

La Figura 26 presenta los resultados para el área de ingeniería se encontró que el uso del DL más frecuente se relaciona con la visión artificial en aplicaciones como reconocimiento de objetos, personas, emociones o expresiones faciales.

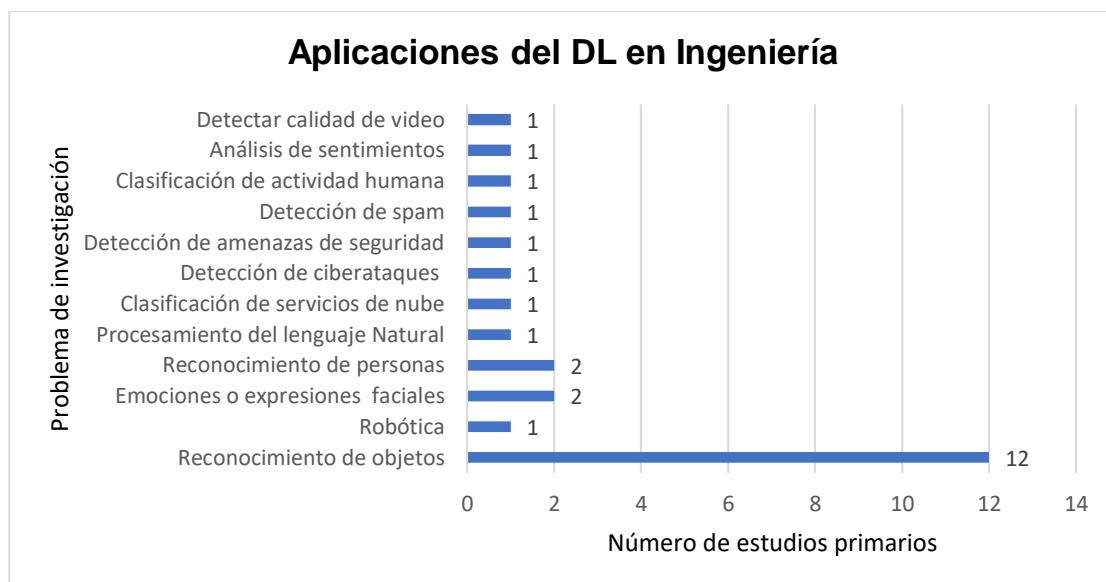


Figura 26. Resultado de aplicaciones del DL en el área de ingeniería

La Figura 27 expresa los resultados relacionados con la industria, donde DL se aplica para la detección de calidad de productos y para la recomendación de hoteles.

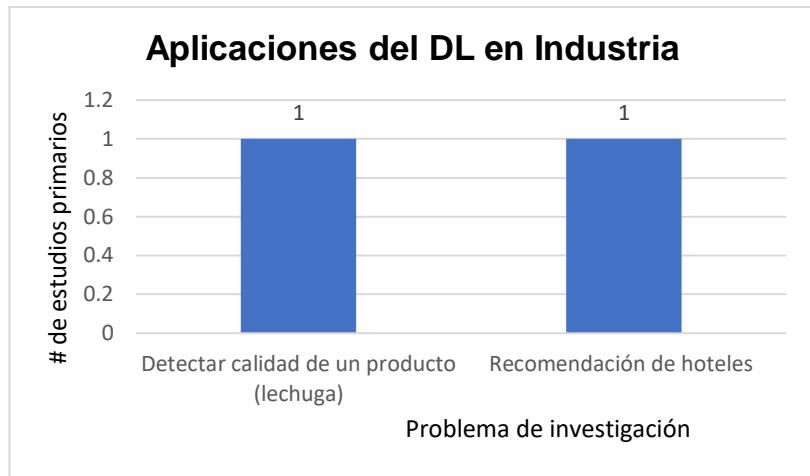


Figura 27. Resultado de aplicaciones del DL en el área de Industria

En la Figura 28 se despliegan los resultados para el área de investigación, estos estudios se colocaron aquí porque se centraron en el análisis de los datos y la creación de métodos para generar los datos; ejemplo el estudio EP28 que identifica ecuaciones de estado (EoS) empleadas en las simulaciones hidrodinámicas a partir de colisiones de iones pesados, o indagan con mayor rigor modelos del DL, como por ejemplo el estudio EP31 de visión artificial, pretendía entender la operación interna y el comportamiento de las CNN y diagnosticar problemas potenciales con el modelo.

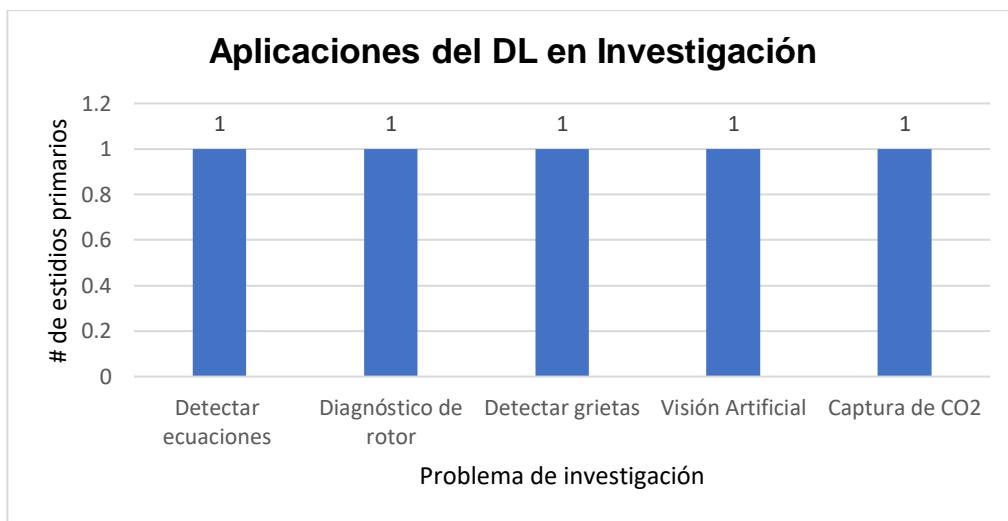


Figura 28. Resultado de aplicaciones del DL en el área de Investigación

La Figura 29 presenta los resultados para el área de salud, el uso del DL más frecuente se relaciona con el diagnóstico médico como detección de cáncer.

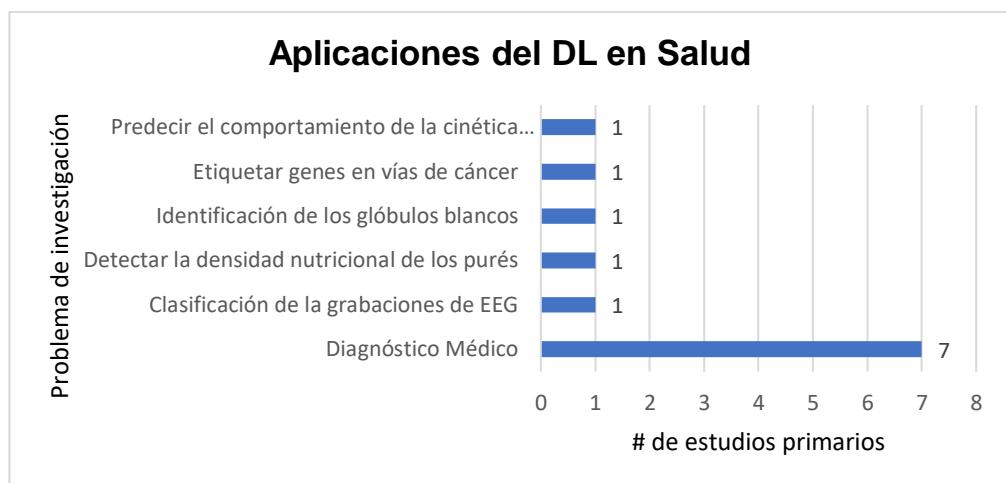


Figura 29. Resultado de aplicaciones del DL en el área de Salud

Finalmente, los resultados de finanzas se muestran en la Figura 30, donde el análisis del mercado de valores y pronóstico de ventas son las que tienen mayor atención de investigación en los últimos años.

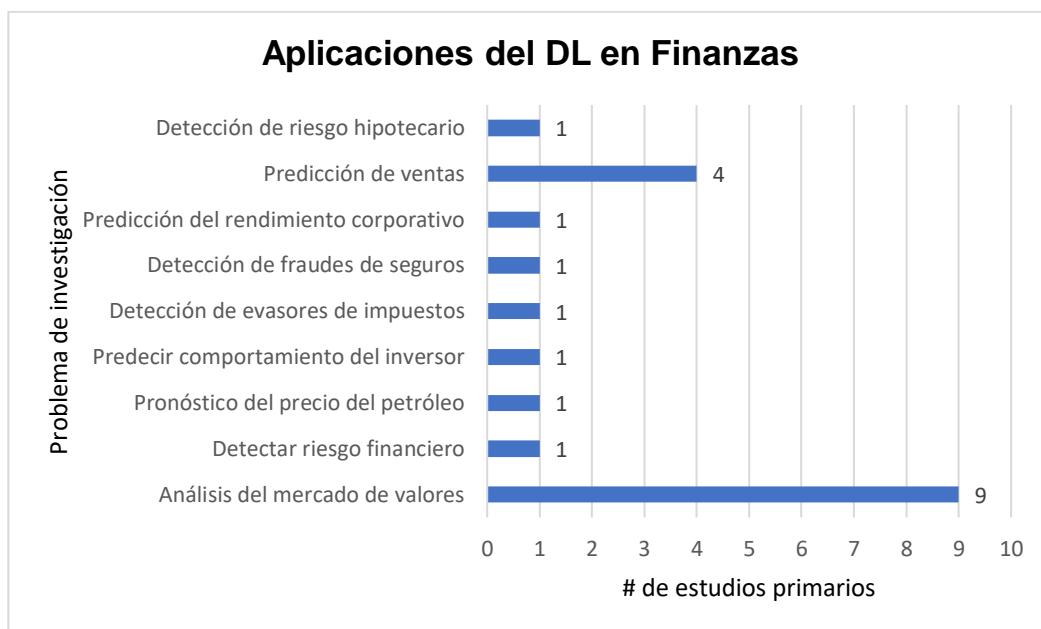


Figura 30. Resultado de aplicaciones del DL en el área de Finanzas

6.1.6.4. Síntesis PI_3 ¿Qué modelos o técnicas del DL se emplean para resolver los problemas de investigación?

En esta pregunta de investigación se enumeró diferentes modelos orientados a la clasificación y detección de patrones, en la Tabla XIII se presentan los modelos empleados en cada estudio analizado. Se observa que algunos estudios emplean dos

o más modelos, algunos autores combinan modelos para obtener mejores resultados, por ejemplo EP04, EP08, EP49, EP61 emplean autoencoders (AE) para reducir la dimensionalidad, o adicionan al modelo DL modelos de aprendizaje por refuerzo como es el caso de EP17 y EP54. Es importante resaltar que los estudios analizados que emplean modelos diferentes a un MLP profundo, utilizan a la salida del modelo una red neuronal completamente conectada como puede ser un perceptrón, pero en otros casos los autores emplean en la salida una red neuronal del tipo MLP.

TABLA XIII.
IDENTIFICACIÓN DE MODELOS / ALGORITMOS (PI_3)

Área	Estudios Primarios	MODELOS							
		MLP	CNN	AE	SAE	LSTM	RBM	DBN	Q-learning
Ingeniería	EP01	✓							
	EP02		✓						
	EP03		✓						
	EP04		✓	✓					
	EP05				✓			✓	
	EP06	✓				✓			
	EP07	✓							
	EP08	✓		✓					
	EP09	✓							
	EP10	✓							
	EP11	✓	✓			✓			
	EP12		✓						
	EP13		✓						
	EP14		✓						
	EP15		✓						
	EP16	✓	✓						
	EP17				✓				✓
	EP18			✓					
	EP19	✓							
	EP20	✓							
	EP21		✓						
	EP22				✓				
	EP23							✓	
	EP24						✓		
	EP25							✓	
Industria	EP26		✓						
	EP27		✓			✓			
Investigación	EP28		✓						
	EP29							✓	

	EP30	✓					
	EP31	✓					
	EP32					✓	
Salud	EP33	✓					
	EP34	✓					
	EP35	✓					
	EP36			✓			
	EP37	✓					
	EP38			✓			
	EP39	✓					
	EP40	✓					
	EP41	✓					
	EP42	✓					
	EP43	✓					
	EP44			✓			
Finanzas	EP45	✓					
	EP46	✓					
	EP47		✓				
	EP48	✓					
	EP49			✓	✓		
	EP50			✓			
	EP51	✓					
	EP52	✓					
	EP53		✓		✓		
	EP54	✓					✓
	EP55	✓					
	EP56				✓	✓	
	EP57				✓		
	EP58				✓		
Tecnología	EP59	✓					
	EP60				✓		
	EP61	✓		✓			
	EP62	✓					
	EP63		✓				
	EP64				✓		

En la Figura 31 se observa que el modelo de mayor aplicación en los estudios seleccionados es CNN (red neuronal convolucional), una razón puede ser que el DL tiene mayor aplicación en la visión artificial o procesamiento de imágenes y el modelo que más se ha explorado en este campo es CNN.

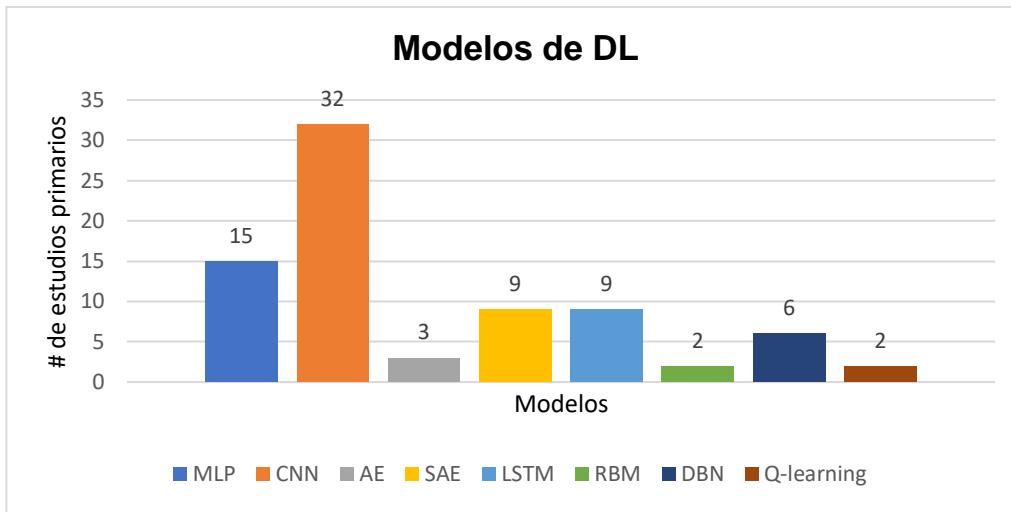


Figura 31. Resultado de los modelos DL aplicados en los estudios primarios

Los estudios seleccionados trabajaron sobre diferentes datasets, los cuales se pueden agrupar por: imágenes, texto, históricos y texto e históricos. En la Figura 32 se agrupa los modelo DL en base a los grupos de datos antes mencionados, se visualiza que para datasets de imágenes existe una tendencia en aplicar modelos CNN, en el caso de datos históricos no se observa un modelo dominante, esto puede deberse a que se incluyen dentro de históricos datos como: historiales de empresas, comportamiento de la banca, actividades humanas, ventas, historial de twitter, recopilado de datos como: CO2, comportamiento molecular, etc.; es decir datos recopilados en un transcurso de tiempo. Para dataset de texto se encontró mayor aplicación del modelo LSTM. Finalmente se encontró cuatro estudios que combinaban texto y datos históricos en los que se aplicaron modelos LSTM, CNN y DBN.

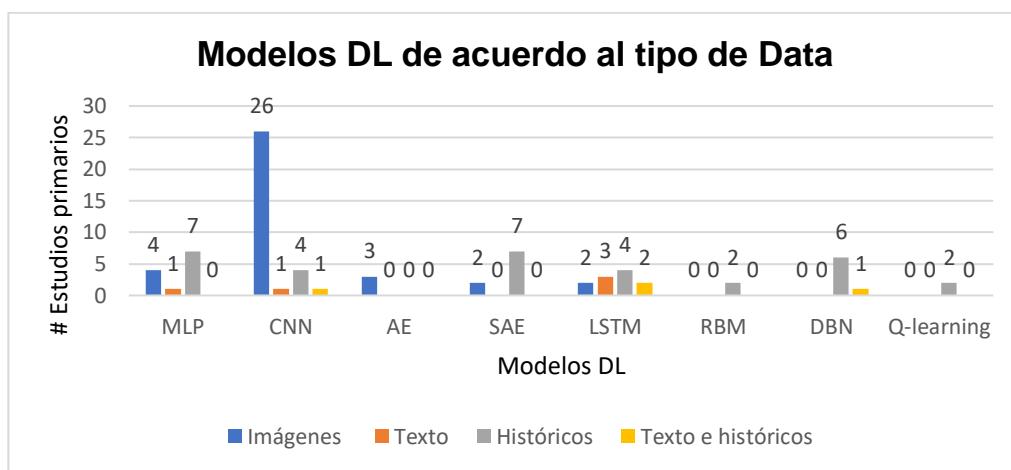


Figura 32. Resultado de los modelos DL de acuerdo al tipo de dataset en la que se aplican

6.1.7. Discusión de la SLR

En esta SLR se encontró que:

- DL se halla principalmente en estudios de ingeniería, donde se emplea con mayor frecuencia en la visión artificial para la extracción de características. Otra aplicación detectada fue la seguridad de la información, donde emplean DL para la detección de software malicioso, además de la detección de spam.
- En salud los estudios encontrados se encuentran relacionados con el diagnóstico médico, en los mismos se observó el uso de técnicas de visión artificial para la detección de anomalías en imágenes registradas durante las revisiones médicas, aquellos que no aplicaban técnicas de visión artificial analizaban historiales médicos para detectar patrones anómalos que permitan predecir la presencia de alguna enfermedad.
- En finanzas todos los datos fueron de tipo texto, históricos, o una mezcla de estos; no se encontró estudios relacionados a finanzas que aplicaran técnicas de visión artificial; las aplicaciones más comunes en esta área fueron el análisis de mercado y la predicción de ventas que en la mayoría de estudios emplean datasets históricos, pero se encontraron estudios en el análisis de mercado que mezclaron datos históricos con la información extraída de textos para mejorar la predicción de la tendencia del mercado de valores.
- Los estudios encontrados en industria no demuestran una tendencia de aplicación del DL, pero se considera que la aplicación del DL puede extenderse a problemas como predicción de stock y distribución de recursos.
- El estudio EP32 encontró que las características extraídas por una red CNN no son patrones aleatorios e imposibles de interpretar, hay que recordar que las redes neuronales colocan pesos a las neuronas por las que ingresa la información determinando cual es relevante y cual no, por tanto, al determinar que aprende un modelo del DL ayudaría a entender y mejorar el rendimiento de estos.
- A pesar de que, se encontró modelos del DL que se aplican con mayor frecuencia a un problema específico, sea el caso de redes CNN para el procesamiento de imágenes no se logró determinar una relación directa entre el problema de investigación y el modelo DL aplicado, se encontró que para visión artificial estudios

como EP01 y EP18 emplean modelos diferentes al CNN en el primer caso emplea un MLP y en el segundo un AE con resultados satisfactorios, por tanto el modelo se elige en base a los criterios del investigador.

- En otros estudios más complejos como EP04, EP05, EP06, EP08, EP11, EP16, EP17, EP27, EP49, EP53, EP54, EP56 y EP61 se combinan varias arquitecturas profundas para obtener mejores resultados, cada estudio resuelve un problema diferente, y los modelos combinados tienen propósitos definidos como es el caso de EP04, EP05, EP08, EP49 que utilizan autoencoders para reducir la dimensionalidad.
- Otra observación interesante son las diferentes soluciones para un mismo problema por ejemplo S54 y S55 aplican DL a la detección de fraudes, en ambos casos emplean un modelo del tipo perceptrón multicapa MLP, pero en el primero se agrega aprendizaje por refuerzo.
- La mayor parte de estudios analizados obtuvieron resultados favorables, pero no ocurrió lo mismo para S47 y S53 en el primer caso los resultados obtenidos con el modelo CNN, no igualan ni superan los resultados obtenidos por técnicas como Regresión logística y Máquinas de vectores de soporte y en el segundo, la red LSTM no mejora el rendimiento del análisis de sentimientos y la red CNN supera con muy poco a la regresión logística, lo cual no compensa el costo computacional.

6.2. Objetivo 2

Demostrar el uso de Deep Learning con el resultado de la revisión sistemática cualitativa de Bárbara Kitchenham.

6.2.1. Pre procesamiento y análisis exploratorio

El dataset de ventas de ILE tiene como características la fecha, el artículo y la venta (véase Tabla II), con lo cual la Serie de Tiempo más bien es un Panel de Datos, ya que tienen observaciones temporales para diferentes artículos. Como el objetivo es demostrar el uso del DL a través de un modelo de predicción de ventas, la data se ajusta como se muestra en la Tabla XIV, donde cada columna representa un artículo.

TABLA XIV.
FRAGMENTO DE LAS VENTAS AJUSTADA POR PRODUCTO

Fecha	1330-001	1330-002	1330-003
01/01/2012	2,62	0,00	0,00
02/01/2012	27,73	170000000000000000,00	0,00
03/01/2012	70,17	2979999999999990,00	1540000000000000,00
04/01/2012	2342100000000000,00	1196500000000000,00	419999999999990,00
05/01/2012	283,65	927000000000000,00	32,90
06/01/2012	3150799999999900,00	279649999999990,00	773500000000000,00
07/01/2012	257,04	221350000000000,00	164150000000000,00
08/01/2012	0,00	0,00	0,00
09/01/2012	2443000000000000,00	309549999999990,00	114,10
10/01/2012	550,89	1437000000000000,00	508999999999900,00
11/01/2012	590,77	103,40	58,95
.....

Al ajustar el dataset y ver cuantas instancias constaban por cada producto, se observó que eran muy pocas, alrededor de 800 por artículo, lo cual es insuficiente para poder entrenar un modelo. Por lo tanto, se decidió que, dado a la falta de observaciones por producto, el modelo predictivo se hizo para el total de la venta diaria, en el periodo 01-03-2011 hasta 28-02-2018 nos da un total de 2257 observaciones para entrenar el modelo (véase Tabla XV), que, si bien no es lo ambicionado para entrenar un modelo DL, se puede entrenar y comparar los resultados con otros modelos.

TABLA XV.
FRAGMENTO DEL DATASET AJUSTADO A LA VENTA TOTAL

Fecha	Venta_Total
01/03/2011	447,29
02/03/2011	0,00
03/03/2011	52,96
04/03/2011	230,80
05/03/2011	18,59
06/03/2011	0,00
07/03/2011	0,00
08/03/2011	0,00
09/03/2011	0,00
10/03/2011	13,25
11/03/2011	162,16
.....

La Figura 33 muestra las ventas de ILE, ignorando la fecha, solamente enumerando las instancias. Es importante tomar en cuenta que existen observaciones de cero o muy por debajo del promedio, esto puede crear ruido en las predicciones, y requieren de un análisis más exhaustivo, sin embargo, eso queda fuera del objetivo de este trabajo.

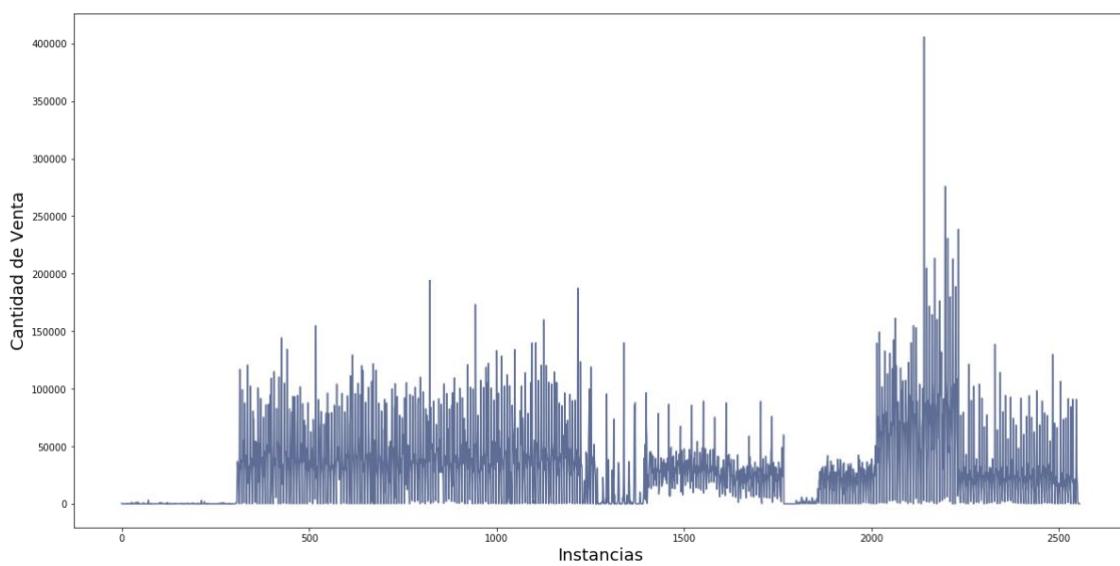


Figura 33. Serie de tiempo de las ventas de ILE

En la Figura 34 se muestra un histograma de los datos y puede verse una tendencia a la izquierda, lo cual se traduce en la presencia de gran número de datos por debajo de la media:

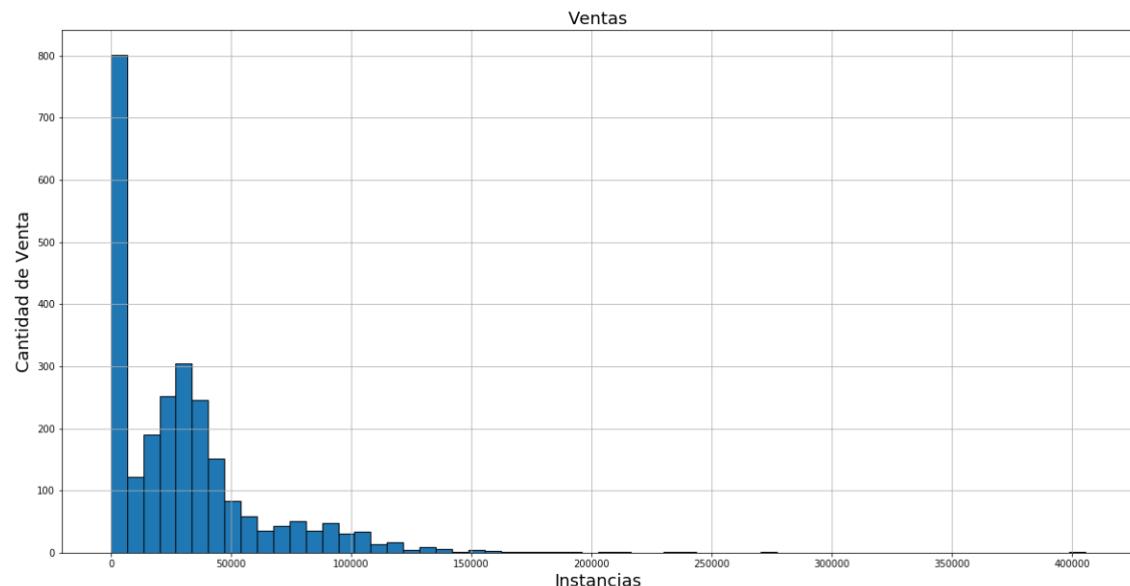


Figura 34. Histograma del total de ventas (2011-2018)

En la Figura 35 se muestran las estadísticas básicas del dataset ventas de ILE:

```
Total
Min. : 0
1st Qu.: 1068
Median : 24567
Mean   : 30091
3rd Qu.: 40545
Max.   : 405628
```

Figura 35. Captura de estadísticas obtenidas en R con el comando summary()

Para trabajar el problema como una serie de tiempo fue necesario transformar los datos a estacionarios. En la Figura 36 se observa la gráfica de la serie de tiempo transformada a estacionaria. Ahora la serie parece oscilar alrededor de una media y una varianza constante (zona más oscura). Sin embargo, para los valores pequeños, las transformaciones no sólo no eliminaron el ruido más bien se hizo más notorio.

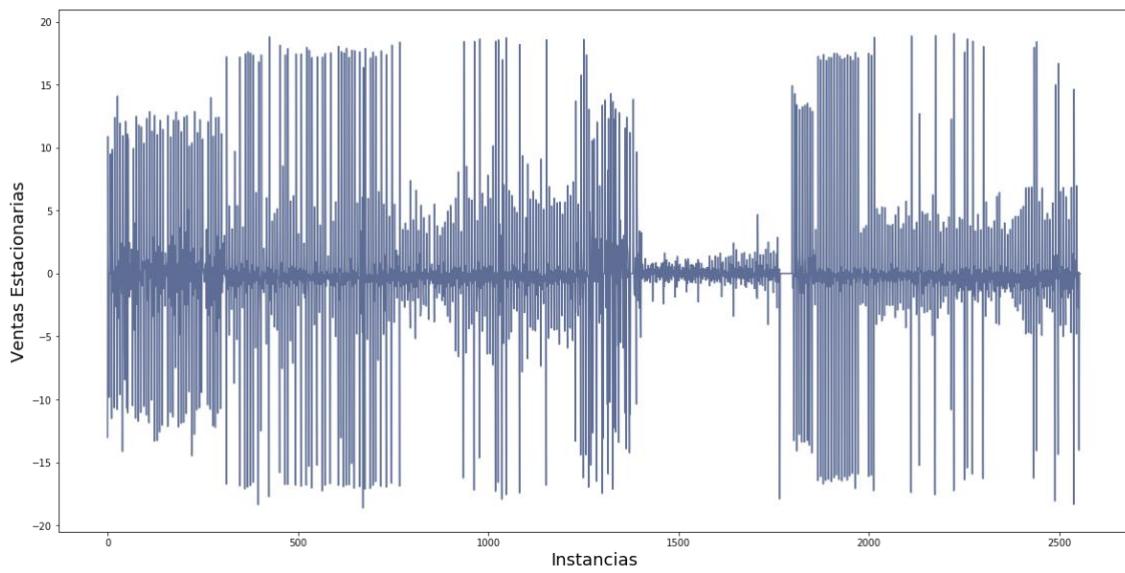


Figura 36. Serie de tiempo estacionaria de las ventas de ILE

Para tener un criterio más robusto de estacionalidad, se aplicó Test de Dickey-Fuller que valida si la serie es estacionaria. La línea de código en R y los resultados del test se observan en la Figura 37 donde se puede ver que la serie es estacionaria:

```
> adf.test(diff(log(ts(Ventas_ILE))), alternative="stationary", k=0)

Augmented Dickey-Fuller Test

data: diff(log(ts(Ventas_ILE)))
Dickey-Fuller = -81.266, Lag order = 0, p-value = 0.01
alternative hypothesis: stationary
```

Figura 37. Captura de la ejecución del test de Dickey-Fuller

Se volvió a graficar el histograma de los datos en la Figura 38 y se observa que los datos casi formaron una campana, lo cual demostró que ya no hay tendencia:

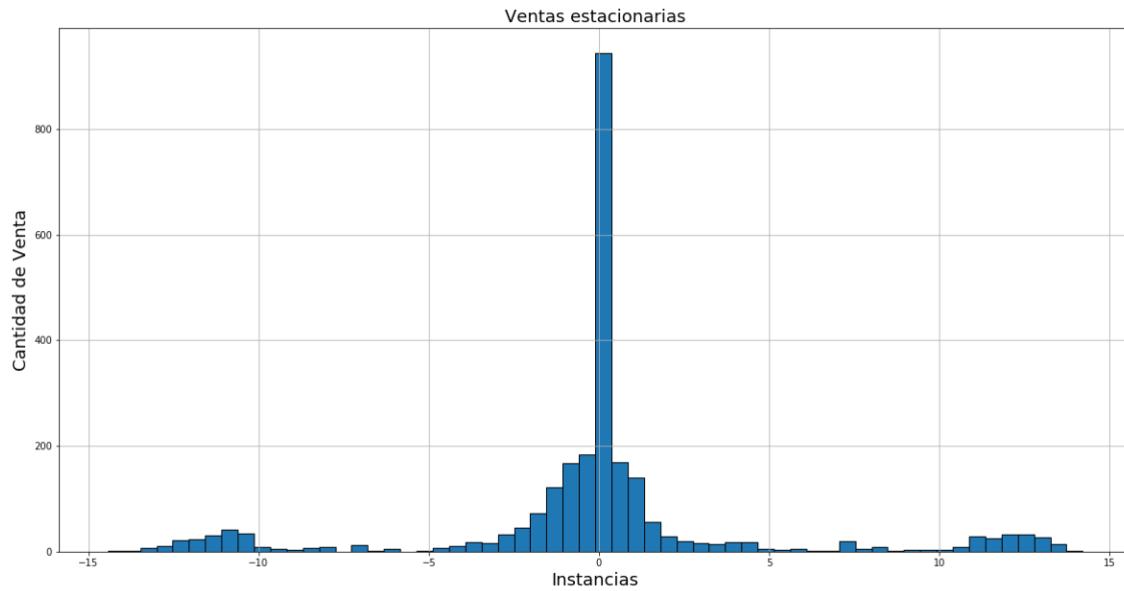


Figura 38. Histograma de ventas estacionarias

Aplicando la normalización de datos mediante el escalado de variables, en la Figura 39 muestra las observaciones escaladas.

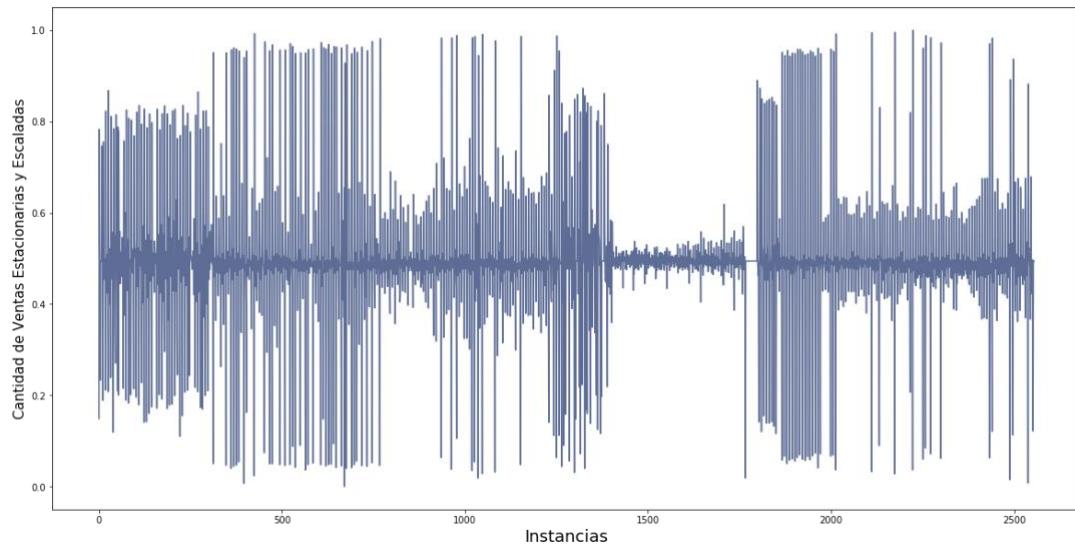


Figura 39. Serie de tiempo de las ventas de ILE escalada y estacionaria

6.2.2. Elegir los modelos

Es importante resaltar que el objetivo no es encontrar la mejor solución para la predicción de ventas de ILE, más bien se busca demostrar la potencia del DL frente a otras técnicas del ML, por lo que se consideraron tres modelos, dos tradicionales (ARIMA y red neuronal del tipo perceptrón de una capa) y el modelo DL LSTM.

6.2.3. Entrenamiento y ajuste de parámetros red LSTM

Como se mencionó en la metodología, se usó Keras para construir el modelo DL y como resultado de la SLR se seleccionó una red LSTM. Los principales módulos empleados para ejecución del modelo son:

```
3.   from keras.models import Sequential
4.   from keras.layers import Dense
5.   from keras.layers import LSTM
6.   from keras.layers import Dropout
7.   #MODELO
8.   model = Sequential()
9.
10.  model.add(LSTM(10, input_length = lag, input_dim = 1))
11.
12.  #OPTIMIZACIÓN
13.  model.compile(loss='mean_squared_error', optimizer='adam')
14.
15.  #ENTRENAMIENTO
16.  model.fit(trainX,
17.             trainY, nb_epoch=100,
18.             batch_size=1,
19.             verbose=1,
20.             shuffle=False)
```

El código completo se encuentra en el [Anexo 2](#).

6.2.3.1. Caso base

El tamaño del batch se mantuvo en 1 para evitar un sobreajuste, además debido al pequeño tamaño del dataset no afecta en la corrida del algoritmo.

El caso base se pensó tomando parámetros básicos (véase Figura 40), y a partir de este se irán variando los parámetros con cada experimento, pero se mantendrá como comparador en el primer renglón:

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
2	1	4	1	no	no	100	tanh	4,22761	3,29912	101	0:25:30

Figura 40. Captura de los resultados con los parámetros base

En la Figura 41, Figura 42 y Figura 43 muestran capturas del entrenamiento, la estructura y resultados de la red con estos parámetros

```
In [40]: model.fit(trainX, trainY, nb_epoch=100, batch_size=1, verbose=1, shuffle=False)
Epoch 1/100
2041/2041 [=====] - 2s 1ms/step - loss: 0.0244
Epoch 2/100
2041/2041 [=====] - 2s 1ms/step - loss: 0.0161
Epoch 3/100
2041/2041 [=====] - 2s 1ms/step - loss: 0.0136
Epoch 4/100
2041/2041 [=====] - 2s 1ms/step - loss: 0.0130
Epoch 5/100
2041/2041 [=====] - 3s 1ms/step - loss: 0.0129
Epoch 6/100
2041/2041 [=====] - 3s 1ms/step - loss: 0.0128
Epoch 7/100
2041/2041 [=====] - 3s 1ms/step - loss: 0.0128A: 1s - :
Epoch 8/100
2041/2041 [=====] - 3s 1ms/step - loss: 0.0128
Epoch 9/100
2041/2041 [=====] - 3s 1ms/step - loss: 0.0128
Epoch 10/100
2041/2041 [=====] - 3s 1ms/step - loss: 0.0127
```

Figura 41. Captura de las épocas de entrenamiento del caso base

El número de parámetros en cada capa es:

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(1, 4)	96
dense_1 (Dense)	(1, 1)	5
<hr/>		
Total params: 101		
Trainable params: 101		
Non-trainable params: 0		

Figura 42. Estructura de la red y parámetros del caso base

Evaluando modelo en entrenamiento y prueba:
 Train Score: 4.22761 RMSE
 Test Score: 3.29912 RMSE

Figura 43. Captura de los resultados del caso base

6.2.3.2. Dropout

Para agregar la técnica dropout al modelo se incluyó `model.add(Dropout(p))` al modelo; en la Figura 44 se observa los diferentes valores con los que se experimentó, empezando por colocar al dropout $p = 0,5$, al ver que el resultado no mejoró se varió el valor de p hasta lograr el óptimo con $p = 0,1$.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
2	1	4	1	no	no	100	tanh	4,22761	3,29912	101	0:25:30
Modificando dropout											
2	1	4	1	no	si, p=0,5	100	tanh	3,87864	3,45729	101	0:19:25
2	1	4	1	no	si, p=0,2	100	tanh	3,73345	3,30554	101	0:21:00
2	1	4	1	no	si, p=0,8	100	tanh	4,1299	3,81689	101	0:18:04
2	1	4	1	no	si, p=0,1	100	tanh	3,72496	3,20202	101	0:20:30

Figura 44. Captura de los resultados modificando el dropout

6.2.3.3. Tamaño del Lag

En la Figura 45 se presentan los resultados al variar el tamaño del lag, al aumentar $lag = 12$ el error se redujo y aumentó al sumar un punto ($lag = 13$), el mejor resultado se obtuvo al reducir el tamaño a $lag = 8$. También se puede observar que al modificar el lag el número de parámetros se mantiene en 101.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
2	1	4	1	no	si, p=0,1	100	tanh	3,72496	3,20202	101	0:20:30
Modificando lag											
3	1	4	1	no	si, p=0,1	100	tanh	3,66505	3,23155	101	0:28:45
6	1	4	1	no	si, p=0,1	100	tanh	2,79555	2,80478	101	0:30:03
5	1	4	1	no	si, p=0,1	100	tanh	3,45550	3,20180	101	0:32:05
12	1	4	1	no	si, p=0,1	100	tanh	2,74367	2,79802	101	0:38:00
13	1	4	1	no	si, p=0,1	100	tanh	3,50090	2,87960	101	0:40:00
10	1	4	1	no	si, p=0,1	100	tanh	2,90161	2,69407	101	0:38:46
8	1	4	1	no	si, p=0,1	100	tanh	2,73881	2,67092	101	0:36:06

Figura 45 Captura de los resultados modificando el lag

6.2.3.4. Número de Bloques

Se varió el número de bloques que hay en cada capa, en la Figura 46 se observa que el mejor desempeño se logra con $Bloques = 10$, nótese que también se altera el número de parámetros en cada experimento.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
8	1	4	1	no	si, p=0,1	100	tanh	2,73881	2,67092	101	0:36:06
Modificando Bloques											
8	1	2	1	no	si, p=0,1	100	tanh	5,55509	3,11815	35	0:30:00
8	1	5	1	no	si, p=0,1	100	tanh	3,87060	3,28180	146	0:31:25
8	1	10	1	no	si, p=0,1	100	tanh	2,69603	2,62997	491	0:32:18
8	1	6	1	no	si, p=0,1	100	tanh	2,75467	2,84563	199	0:29:56
8	1	8	1	no	si, p=0,1	100	tanh	2,62983	2,71030	329	0:30:35

Figura 46. Captura de los resultados modificando el número de bloques por capa.

6.2.3.5. Stateful

Al cambiar el parámetro stateful no se observó ningún cambio, por lo que se reseteo el estado de la red al terminar cada época empleando el siguiente método:

```
1. Stateful=True, resetea el estado en cada época
2. for i in range(100):
3.     model.fit(trainX, trainY, nb_epoch=1,
4.                batch_size=batch_size, verbose=2, shuffle=False)
5.     model.reset_states()
```

En la Figura 47 se puede observar que el error no mejoró y que aumentó el tiempo de ejecución.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
8	1	10	1	no	si, p=0,1	100	tanh	2,69603	2,62997	491	0:32:18
Modificando Stateful											
8	1	10	1	si	si, p=0,1	100	tanh	2,69664	2,63961	491	0:45:06

Figura 47. Captura de los resultados modificando el parámetro stateful

6.2.3.6. Número de Épocas

En la Figura 48 se observó que a mayor número de épocas el error se reduce, pero el tiempo de ejecución aumenta.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
8	1	10	1	no	si, p=0,1	100	tanh	2,69603	2,62997	491	0:32:18
Modificando número de épocas											
8	1	10	1	no	si, p=0,1	80	tanh	3,77974	3,80333	491	0:29:45
8	1	10	1	no	si, p=0,1	200	tanh	3,11539	2,61237	491	0:38:47

Figura 48. Captura de los resultados modificando el número de épocas

6.2.3.7. Función de activación

Se probó con varias funciones de activación, en la Figura 49 se observa que se obtienen mejores resultados con la función sigmoide, también observamos que el tiempo de ejecución aumentó siendo el más alto después del experimento que aplica la función lineal.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
8	1	10	1	no	si, p=0,1	200	tanh	3,11539	2,61237	491	0:38:47
Función de Activacion											
8	1	10	1	no	si, p=0,1	200	relu	4,16812	3,26667	491	0:38:50
8	1	10	1	no	si, p=0,1	200	sigmoid	2,19801	2,20418	491	0:45:32
8	1	10	1	no	si, p=0,1	200	hard_sigmoid	3,10938	2,81715	491	0:40:00
8	1	10	1	no	si, p=0,1	200	lineal	3,04838	2,83978	491	0:50:00

Figura 49. Captura de los resultados modificando la función de activación

6.2.3.8. Número de capas

En la Figura 50 se trabaja la profundidad de la red, los resultados muestran que agregarle más capas al modelo no reduce el error y además consume un tiempo considerablemente mayor que el modelo de una capa.

Lag	Batch	Bloques	Capas	Stateful	Dropout	Épocas	Función Activación	RMSE train	RMSE test	Parámetros	Tiempo
Parámetros Base											
8	1	10	1	no	si, p=0,1	200	sigmoid	2,19801	2,20418	491	0:45:32
Número de capas											
8	1	10	2	no	si, p=0,1	200	sigmoid	2,03534	2,38636	1401	1:17:09
10	1	10	3	no	si, p=0,1	100	sigmoid	3,04838	2,46667	491	1:20:07
10	1	10	4	no	si, p=0,1	100	sigmoid	3,10938	2,61715	491	1:25:08
10	1	10	5	no	si, p=0,1	200	sigmoid	2,90658	2,67963	491	1:04:06

Figura 50. Captura de los resultados modificando el número de capas

Terminado estos experimentos se obtuvo los parámetros de la Tabla XVI, los cuales se ajustan mejor al modelo para reducir el error y en la Figura 51 se muestra los resultados de la red LSTM con estos parámetros.

TABLA XVI.
VALORES USADOS EN CADA PARÁMETRO DE LA RED LSTM

Parámetro	Valor adecuado
Tamaño del Lag	8
Batch	1
Número de Bloques	10
Número de Capas	1
Stateful	False
Dropout	0,1
Número de épocas	200
Función de activación	sigmoide

```
Evaluando modelo en entrenamiento y prueba:
Train Score: 2.19801 RMSE
Test Score: 2.20418 RMSE
```

Figura 51. Resultados de la LSTM en Keras

6.2.4. Modelo ARIMA

El modelo ARIMA es un modelo autor regresivo integrado de promedio móvil, se aplica frecuentemente en series de tiempo, estadística y econometría [24][26]. En R hay una función *auto.arima()* que calcula los parámetros automáticamente y devuelve el posible mejor modelo y es la que se aplicó en este estudio.

Como ya se mencionó para emplear el modelo ARIMA se utilizó *auto.arima()* (véase Figura 52), este algoritmo calcula automáticamente los parámetros y nos devuelve el posible mejor modelo.

```
> modelo<-auto.arima(data_estacionaria)
> summary(modelo)
Series: data_estacionaria
ARIMA(1,0,2) with zero mean

Coefficients:
            ar1      ma1      ma2
           -0.7897  -0.0133  -0.8161
         s.e.   0.0231   0.0177   0.0152

sigma^2 estimated as 17.6:  log likelihood=-7291.44
AIC=14590.88  AICc=14590.9  BIC=14614.27

Training set error measures:
          ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set 0.01017815 4.192876 2.520297 -Inf    Inf 0.4841581 -0.002365047
```

Figura 52 Captura del modelo ARIMA (con *auto.arima* y en R)

6.2.5. Red neuronal superficial

Finalmente se diseñó una red neuronal del tipo perceptrón de una capa, para obtener los parámetros se siguió el mismo proceso que el modelo DL LSTM. En este caso los parámetros requeridos son: tamaño del lag, número de bloques, número de épocas y función de activación.

En la Figura 53 se observa todos los experimentos realizados en la búsqueda de los parámetros adecuados para el modelo, el error de esta red no se aleja del error del

modelo DL, los tiempos de ejecución son cortos y se obtuvo mejores resultados con la función de activación ReLU. Al emplear los parámetros del modelo DL el error aumentó.

Lag	Bloques	Épocas	Función Activación	RMSE en train	RMSE en test	Parámetros	hora
Parámetros Base							
2	4	100	tanh	4,22761	4,29912	41	0:14:10
Modificando lag							
3	4	100	tanh	3,66505	3,23155	41	0:15:00
6	4	100	tanh	3,66505	3,23155	41	0:15:03
5	4	100	tanh	3,66505	3,23155	41	0:15:07
12	4	100	tanh	2,94367	2,99802	41	0:22:00
13	4	100	tanh	3,72305	3,83455	41	0:15:04
10	4	100	tanh	2,90161	2,67407	41	0:15:06
8	4	100		2,73881	2,69092	41	0:14:09
Modificando Bloques							
10	2	100	tanh	3,55509	3,11815	35	0:09:34
10	5	100	tanh	3,66505	3,23155	51	0:12:02
10	10	100	tanh	2,69603	2,62997	101	0:15:54
10	6	100	tanh	3,66505	3,23155	61	0:13:04
10	8	100	tanh	2,62983	2,71030	81	0:13:56
Modificando número de épocas							
10	10	80	tanh	3,77974	3,60333	101	0:11:56
10	10	200	tanh	2,90003	2,61501	101	0:12:34
Función de Activación							
10	10	200	relu	3,18181	2,60783	101	0:10:23
10	10	200	sigmoid	4,19295	3,32873	101	0:12:34
10	10	200	hard_sigmoid	3,17539	2,91237	101	0:11:32
10	10	200	lineal	3,97313	2,82845	101	0:16:07
Parámetros iguales a la red LSTM							
8	10	200	sigmoid	3,26215	2,82495	101	0:09:13

Figura 53. Captura de los resultados modificando los diferentes parámetros de la red

En la Figura 54 se expone la estructura final de la red, en la Figura 55 se observan los resultados del modelo adecuado.

El número de parámetros en cada capa es:

Layer (type)	Output Shape	Param #
<hr/>		
Capa_oculta (Dense)	(None, 10)	90
<hr/>		
Capa_salida (Dense)	(None, 1)	11
<hr/>		
Total params: 101		
Trainable params: 101		
Non-trainable params: 0		

Figura 54. Captura de la estructura de la red

```
Evaluando modelo en entrenamiento y prueba:  
Train Score: 3.18181 RMSE  
Test Score: 2.60783 RMSE
```

Figura 55. Resultados del perceptrón unicapa en Keras

En la Tabla XVII se observa en orden en el desempeño de los modelos con respecto al error RMSE

TABLA XVII.
ERROR RMSE ALCANZADO POR LOS MODELOS DESARROLLADOS

#	Modelo	RMSE
1	Modelo DL	2,20418
2	Modelo perceptrón unicapa	2,60783
3	Modelo ARIMA	4,192876

Se observa que el error entre los modelos LSTM y de perceptrón unicapa es cercano, a diferencia del modelo ARIMA. Se considera que el desempeño del modelo DL es aceptable considerando que sólo se usaron 2257 instancias, este desempeño puede mejorar si se incrementa el tamaño del conjunto de datos, pues de esta manera la arquitectura tendrá oportunidad de modelar dependencias temporales mucho más complicadas, lo cual se esperaría por la habilidad de los modelos DL para modelar conceptos abstractos y jerárquicos. También se pudo analizar y mejorar el ruido en los datos generado por la presencia de valores iguales a cero o muy pequeños.

6.3. Objetivo 3

Escribir los resultados logrados en un artículo para su envío a una revista indexada e inicie su proceso de publicación.

El manuscrito titulado “Aplicaciones del Deep Learning en la Intermediación Financiera: Revisión Sistemática de Literatura” fue aceptado en el congreso SIIPRIN'2018 (en el Anexo 3 se adjunta la carta de aceptación) y se encuentra publicado en la plataforma digital de Knowledge E (<https://knowledgee.com/>) como memorias de congreso con el ISSN 2518-6841 (véase Figura 62).

Ibero-American Symposium on Computer Programming jointly held with the International Congress on Technology Education

The Ibero-American Symposium on Computer Programming (SIIPRIN) represents an Ibero-American meeting forum for researchers and professionals focused on the study and research of the different aspects involved in computer programming. Its goal is to foster the dissemination of works keeping in mind academic, scientific, and professional issues on this topic. SIIPRIN arises from the need to create and maintain a discussion forum among scientists, academics, and professionals interested in the study and research of computer programming in our society as well as in the academic arena. In this symposium, a meeting place was provided to present, disseminate, and discuss the results of ongoing or consolidated research conducted on this topic. As an implicit objective, it is intended to foster collaboration among the different stakeholders, where we expect to encourage the development of a scientific basis in this field. In this edition, the International Congress on Technology Education and Knowledge Management (CITEGC) was jointly held with the SIIPRIN. CITEGC is focused on the fields of new technologies applied to education and the development of knowledge management in organizations.

Conference date: 29–30 November 2018

Location: Riobamba, Ecuador

Editors: Gloria Arcos, Blanca Hidalgo, Vanessa Valverde, Alejandra Oñate, Lorena Aguirre, Raúl Rosero, Danilo Pástor, Gustavo Hidalgo, Eduardo Villa, Washington Luna, Patricio Moreno, Raúl Lozada, and Omar S. Gómez

Organizers: Grupo de Investigación en Ingeniería de Software (GrillSoft) and Grupo de investigación en Tecnologías de la Información para la Gestión del Conocimiento (TIGECON)

Sponsor: • Escuela Superior Politécnica de Chimborazo (ESPOCH)

Published: 27 December 2018

ISSN: 2518-6841

Figura 56. Captura de la memoria del congreso SIIPRIN'2018

Finalmente, este artículo se adjunta en el Anexo 4 y se encuentra disponible de manera gratuita bajo una licencia Creative Commons con el DOI 10.18502/keg.v3i9.3645. La defensa del artículo se justifica en el Anexo 5.

Adicionalmente, los resultados del modelo DL para predicción de ventas se presentaron en el II Ciclo de Conferencias CIS-UNL 2019 con la temática “Inteligencia Artificial” organizada por la Carrera de Ingeniería en Sistemas y Computación de la Universidad Nacional de Loja (véase Anexo 6).

7. Discusión

En este apartado se examina los resultados obtenidos en la sección 1 se discute los resultados de cada objetivo específico planteado en el presente Trabajo de Titulación “Detección de técnicas de aprendizaje profundo aplicadas en las diferentes áreas del conocimiento, empleando el método de revisión sistemática de literatura.”, trabajo propuesto considerando el acelerado desarrollo y aplicación del Deep Learning. En la sección 2 se describe la valoración técnica, económica y científica, alcanzada al finalizar el Trabajo de Titulación.

7.1. Desarrollo de la Propuesta Alternativa

Para poder desarrollar el presente Trabajo de Titulación se cumplió con los siguientes objetivos:

OBJETIVO ESPECÍFICO 1: Aplicar la técnica de revisión sistemática cualitativa de Bárbara Kitchenham para obtener los avances y conocimientos actuales de aplicaciones del Deep Learning en diferentes áreas de conocimiento.

Los resultados de este objetivo permitieron identificar las aplicaciones y modelos del DL, en las áreas de conocimiento de ingeniería, investigación, industria, finanzas y medicina, estos resultados son útiles para entender el alcance del DL, además de orientar en la selección de modelos DL para futuras investigaciones. Se decidió utilizar la SLR considerando que es una de las pocas que se orienta a la ingeniería y cuenta con un protocolo claramente establecido y estandarizado que garantiza la claridad y transparencia en el transcurso de la revisión.

El proceso empieza por determinar la necesidad de realizar la SLR que es identificar la aplicación de modelos DL, en base a ello se especificó tres preguntas de investigación que guiaron la recolección y el análisis de la información empleando el protocolo de revisión propuesto por Kitchenham [2], existen otros métodos para establecer correctamente el centro de la revisión como mentefactos conceptuales que no se emplearon en esta investigación. Dentro de la planificación de la SLR se establecieron cuatro motores de búsqueda considerando como criterio de selección la admisión de consultas avanzadas, es importante la correcta elaboración de las cadenas de búsqueda, para este trabajo se realizó una revisión previa para definir las palabras

claves que se concatenaron utilizando los operadores lógicos AND y OR para formar las cadenas de búsqueda, las cuales nos sirvieron para recopilar artículos que luego de pasar por los criterios de inclusión-exclusión se redujeron a 64 estudios primarios los cuales se analizaron en la búsqueda de los modelos de DL aplicados a diferentes problemas de investigación.

OBJETIVO ESPECÍFICO 2: Demostrar el uso de Deep Learning con el resultado de la revisión sistemática cualitativa de Bárbara Kitchenham

Este objetivo se planteó para demostrar el potencial de los modelos DL frente a los modelos convencionales de machine learning, para ello se empleó los resultados de la SLR donde se observó que para predicción de ventas el modelo de mayor uso son las redes LSTM. En los datos que corresponden a las ventas de ILE (Industria Lojana de Especerías), se encontraron valores ausentes y atípicos pero no se aplicaron técnicas de limpieza de datos diferentes a la normalización por escalado de variables para evaluar la eficacia de Deep Learning en datos con ruido, estudios en procesamiento de imágenes como [27][28] demostraron que los modelos profundos pueden trabajar en imágenes con ruido, esta decisión también se basa en que el objetivo no es encontrar la mejor solución para la predicción de ventas de los datos de ILE, más bien demostrar el alcance de los modelos Deep Learning. La predicción de ventas se trabajó como un problema de regresión para ello se empleó un método de regresión de variables, estudios parecidos [11][26][29] demostraron que este método da buenos resultados.

Para evaluar el desempeño del modelo LSTM se empleó el método de prueba y error para establecer los parámetros del modelo, y se comparó con el modelo ARIMA, uno de los más comunes en predicción de series temporales, y debido a que los modelos profundos basan su estructura en redes neuronales se comparó los resultados con una red perceptrón de una capa, que se considera una red neuronal no profunda; se utilizó como medida de comparación el error cuadrático medio. Los experimentos realizados miden el error en la predicción de ventas para los datos de prueba evaluando el comportamiento de los modelos en la predicción de nuevos datos. En los experimentos realizados se encontró que el error del modelo LSTM es de 2,20, seguido del 2,60 alcanzado por la red perceptrón de una capa y el error de 4,19 del modelo ARIMA, estos resultados demuestran que al analizar las series de datos con valores atípicos, los tres modelos son capaces de capturar la máxima información presente en los datos sin embargo, el modelo DL LSTM supera a los modelos convencionales, a pesar de ello

estos resultados no superan los alcanzados por [29] realizado con las ventas de productos de mercado recolectados durante 45 semanas, en donde se demostró la efectividad de las redes DL LSTM en la predicción de ventas en 16 de 66 productos con un error mínimo de 0,3, pero el error alcanzado en estos experimentos es menor que los resultados del estudio [26] realizado con 779569 instancias de productos de farmacia que demuestra que las redes LSTM funcionan con valores atípicos, en este estudio el error supera los 1246. Por tanto, las redes LSTM demuestran potencial en la previsión de ventas en conjuntos de datos limitados.

OBJETIVO ESPECÍFICO 3: Escribir los resultados logrados en un artículo para su envío a una revista indexada e inicie su proceso de publicación.

Con este objetivo se buscó transferir los resultados obtenidos sobre el uso de DL en el área financiera para exponer el crecimiento de DL en el área financiera, identificar que aplicaciones del área financiera son abordadas por los investigadores y los modelos DL comúnmente aplicados. Los resultados expuestos a través de un artículo de revisión sirven como referencia de la aplicación del DL en el área financiera para orientar futuras investigaciones en esta área.

El artículo denominado “Aplicaciones del Deep Learning en la Intermediación Financiera: Revisión Sistemática de Literatura” se presentó en el tercer Simposio Iberoamericano en programación Informática (SIIPRIN’2018), para ello el artículo se sometió al proceso de revisión por pares en el Anexo 3 se encuentra la carta de aceptación del artículo. Una vez que fue aceptado el artículo, los resultados fueron expuestos a los estudiantes y presentes en el congreso SIIPRIN’2018 y se encuentran disponibles para la comunidad científica como memoria de congreso en la plataforma digital de Knowledge E con el ISSN 2518-6841.

7.2. Valoración Técnica, Económica y Científica.

La valoración del Trabajo de Titulación se expresa describiendo los beneficios presentados en cuatro aspectos:

7.2.1. Valoración Técnica

- A través del gestor bibliográfico Mendeley se automatizó la recolección de artículos científicos, ya que permite organizar las referencias de manera sencilla desde la fuente.
- El método heurístico de prueba y error y el análisis exploratorio fueron la base para el desarrollo del modelo de predicción de ventas.
- Las herramientas Keras, Jupyter, Python y R facilitaron el desarrollo y ejecución del modelo de predicción de ventas.

7.2.2. Valoración Económica

Para poder llevar acabo los objetivos que se han planteado para este proyecto fue necesaria la inversión en recursos como recursos de hardware y software, servicios, imprevistos y talento humano. Los cuales son detallados a continuación:

7.2.2.1. Talento Humano

El Trabajo de Titulación involucra al tesista y la asesoría de un docente de la carrera cuyo costo es asumido por la Universidad Nacional de Loja. El tiempo empleado para el desarrollo del presente Trabajo de Titulación es de 480 horas. En la Tabla XVIII se da a conocer el costo que asumió el talento humano.

TABLA XVIII.
TALENTO HUMANO

ROL	Número de Horas	Valor por hora (\$)	Valor Total
Responsable	480.00	10,00	4800,00
Tutor	50.00	20,00	1000,00
Total (\$):			5.800,00

7.2.2.2. Recursos de hardware y software

Los recursos hardware y software empleados en este trabajo de titulación se muestran en la Tabla XIX, representan todos bienes que serán necesarios adquirir para poder realizar sin inconvenientes el desarrollo del presente proyecto.

TABLA XIX.
RECURSOS HARDWARE Y SOFTWARE

Recursos	Cantidad	Valor unitario	Valor Total
HARDWARE			
Portátil	1	1950,00	1950,00
Memoria Flash	2	10,00	20,00
Subtotal (\$):			1970,00
SOFTWARE			
Bases de datos científicas	1	00,00	00,00
Python	1	00,00	00,00
Jupyter	1	00,00	00,00
Keras	1	00,00	00,00
Subtotal (\$):			00,00
Total (\$):			1970,00

7.2.2.3. Servicios

En el transcurso del desarrollo del presente proyecto fue necesario adquirir los servicios de transporte e Internet (véase Tabla XX) para culminar con éxito las tareas que demanda el presente trabajo.

TABLA XX.
SERVICIOS

SERVICIO	Cantidad	Valor unitario	Valor Total
Transporte	200	0,30	60,00
Internet	400 h	0,75	300,00
Total (\$):			360,00

7.2.2.4. Materiales de oficina

Se empleó varios materiales de oficina los cuales se describen en la Tabla XXI

TABLA XXI.
MATERIALES DE OFICINA

Recursos	Cantidad	Valor unitario	Valor Total
Impresiones	312	0,05	15,60
Copias	312	0,02	6,24
Anillados	3	3,50	10,50
CDs	3	0,50	1,50
Empastado	1	28,00	28,00
	Total (\$):		61,84

7.2.2.5. Presupuesto final

Para imprevistos se tomó el 10% del valor total del presupuesto, los cuales se agregaron al valor total del proyecto y se muestran en la Tabla XXI.

TABLA XXII.
PRESUPUESTO TOTAL DEL PROYECTO

RECURSO	SUBTOTAL
T. Humano	5800,00
Recursos de hardware y software	1970,00
Servicios	360,00
Materiales de oficina	61,84
Subtotal (\$):	8191,84
Imprevistos 10%	819,84
Total (\$):	9011,68

7.2.3. Valoración Científica

- El aporte que presta a la realización de trabajos futuros, ya que esta revisión contiene una variedad de literatura y bibliográfica que es relevante a este tema y que permitirá agilizar la búsqueda de documentos que aporten conocimiento sobre el mismo.
- El manuscrito publicado en el SIIPRIN'2018 que está disponible en la plataforma digital de Knowledge E de manera gratuita aporta nuevos conocimientos a la comunidad científica.

8. Conclusiones

- En relación a los 64 estudios analizados, esta investigación identificó el uso de modelos Deep Learning en las áreas de: ingeniería, salud, industria, investigación y finanzas. Empleando el protocolo propuesto por Bárbara Kitchenham [2], se planteó tres preguntas de investigación orientadas a extraer: el área de aplicación, el problema de investigación y el modelo Deep Learning.
- En base a los experimentos realizados para la predicción de ventas, se encontró que el modelo LSTM tiene un error RMSE de 2,20 seguido por el error de la red de perceptrón de una capa de 2,60 y el modelo ARIMA de 4,19. Esto indica que los tres modelos son capaces de capturar la máxima información presente en los datos con valores atípicos, pero el modelo DL LSTM supera a los modelos convencionales, por tanto las redes LSTM demuestran potencial en la previsión de ventas en conjuntos de datos limitados.
- En cuanto a la redacción y publicación de resultados, se realizó la transferencia de conocimiento a una parte de la comunidad académica científica que asistieron a un congreso cuyo fin era mostrar los resultados en el campo del Deep Learning orientados al área financiera.
- Este estudio presenta algunas limitaciones, en la SLR no se analizó la relación del modelo Deep Learning y el campo de aplicación, es decir no se examinaron los criterios a la hora de seleccionar los modelos aplicados en los estudios seleccionados. El tamaño del dataset, ILE solo tiene registros desde el 2011. Datos con presencia de valores atípicos que generan ruido, no se realizó un proceso de limpieza de datos exhaustivo debido a que se deseaba evaluar el comportamiento del Deep Learning en datos con presencia de ruido. Los datos se analizaron en conjunto y no por producto, existe estudios que demuestran que la precisión del modelo varía dependiendo del producto, ya que los datos de cada producto tienen un comportamiento diferente. Finalmente, los experimentos se realizaron empleando la red LSTM como modelo profundo, pero existen otros estudios que aplican otros modelos como CNN, autoencoders e incluso modelos combinados.

9. Recomendaciones

- Como líneas futuras de investigación se recomienda analizar la relación entre la selección de modelos de Deep Learning y el campo de aplicación.
- Emplear técnicas como mapas conceptuales, mentefactos [30], para organizar la idea de la revisión sistemática de literatura y plantear las preguntas que orienten a la investigación.
- En los datos empleados en este proyecto se recomienda realizar un análisis de datos para reducir el ruido generado por valores atípicos, en [31] se presenta técnicas y herramientas para la transformación, normalización, integración, limpieza de ruido e imputación de valores perdidos que son las técnicas empleadas en el pre procesamiento de datos.
- Analizar los datos separados por producto ya que cada conjunto de datos tiene un comportamiento diferente, evaluar en cuales productos Deep Learning obtiene mejores resultados.
- Comparar los resultados obtenidos con otros modelos Deep Learning para explorar si existe una reducción de error.

10. Bibliografía

- [1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. 2016.
- [2] B. Kitchenham, “Procedures for performing systematic reviews,” *Br. J. Manag.*, vol. 14, no. 0, pp. 207–222, 2003.
- [3] S. J. A. Guirao Goris, “Utilidad y tipos de revisión de literatura,” *ENE Rev. enfermería*, vol. 9, no. 5, pp. 453–461, 2015.
- [4] C. Hart, *Doing a Literature Review*, vol. 20, no. 2. 2010.
- [5] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. MIT Press, 2012.
- [6] S. Ben-David and S. Shalev-Shwartz, *Understanding Machine Learning: From Theory to Algorithms*. 2014.
- [7] A. Moreno *et al.*, *Aprendizaje automático*. 1994.
- [8] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [9] C. Cortés Antona, “Herramientas Modernas En Redes Neuronales: La Librería Keras,” *Univ. Autónoma Madrid*, p. 60, 2017.
- [10] N. Buduma, *Deep Learning Designing next-generation artificial intelligence algorithms*. 2016.
- [11] M. F. Mora Alba, “Fundamentos de deep learning y una aplicacion para predecir la demanda de energia electrica en mexico,” Instituto Tecnológico Autónomo de Mexico, 2016.
- [12] I. Fuentes, “Inteligencia Artificial (AI) y sus aplicaciones – LQNSC,” 2018. [Online]. Available: <https://www.pfsgroup.es/loquenosoncuentas/inteligencia-artificial-ai-aplicaciones/>. [Accessed: 18-Feb-2019].
- [13] F. Gomez Quesada, M. Fernandez Graciani, M. Lopez Bonal, and M. Alonso Diaz, “Aprendizaje con Redes neuronales.” .
- [14] C. A. Ruiz, M. S. Basualdo, and D. J. Matich, *Redes Neuronales: Conceptos Básicos y Aplicaciones*. 2001.
- [15] F. Lara Rosano, “Fundamentos de redes neuronales artificiales.”
- [16] F. Moral Algaba, “Redes completamente convolucionales en la segmentación semántica de lesiones melanocíticas,” 2017.
- [17] A. Nacelle, “Redes neuronales artificiales.”
- [18] M. A. Nielsen, “Neural Networks and Deep Learning,” 2015. [Online]. Available: <http://neuralnetworksanddeeplearning.com/chap5.html>. [Accessed: 19-Mar-2019].
- [19] G. J. P. Restrepo Arteaga, “Aplicación del aprendizaje profundo (‘deep learning’) al procesamiento de señales digitales,” Universidad Autónoma de Occidente, 2015.
- [20] I. Vasilev, “Un Tutorial de Aprendizaje Profundo: De Perceptrones a Redes Profundas | Toptal.” [Online]. Available: <https://www.toptal.com/machine-learning/un-tutorial-de-aprendizaje-profundo-de-perceptrones-a-redes->

profundas/es. [Accessed: 19-Mar-2019].

- [21] M. T. Jones, "Arquitecturas de aprendizaje profundo." [Online]. Available: <https://www.ibm.com/developerworks/ssa/library/cc-machine-learning-deep-learning-architectures/index.html>. [Accessed: 22-Feb-2019].
- [22] S. Contreras and F. De la Rosa, "Aplicación de deep learning en robótica móvil para exploración y reconocimiento de objetos basados en imágenes," in *2016 IEEE 11th Colombian Computing Conference (CCC)*, 2016, pp. 1–8.
- [23] "Home - Keras Documentation." [Online]. Available: <https://keras.io/>. [Accessed: 07-Mar-2019].
- [24] G. Corres, L. I. Passoni, C. Zárate, and A. Esteban, "Estudio Comparativo de Modelos de Pronóstico de Ventas," *Iberoam. J. Ind. Eng.*, vol. 6, no. 11, pp. 113–134, 2014.
- [25] S. López Leyva, "El proceso de escritura y publicación de un artículo científico," *Rev. electrónica Educ.*, vol. 17, pp. 5–27, 2013.
- [26] R. Ferreira, M. Braga, and V. Alves, *Forecast in the Pharmaceutical Area – Statistic Models vs Deep Learning*, vol. 746. Springer International Publishing, 2018.
- [27] T. Shimobaba *et al.*, "Computational ghost imaging using deep learning," *Opt. Commun.*, vol. 413, no. October 2017, pp. 147–151, 2018.
- [28] Y. H. Chang *et al.*, "Deep learning based Nucleus Classification in pancreas histological images," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, no. January, pp. 672–675, 2017.
- [29] Q. Yu, K. Wang, J. O. Strandhagen, and Y. Wang, "Application of Long Short-Term Memory Neural Network to Sales Forecasting in Retail—A Case Study," pp. 11–17, 2018.
- [30] J. H. Pimienta Prieto and J. A. García Fraile, *Estrategias de enseñanza-aprendizaje : docencia universitaria basada en competencias*. Pearson Educación, 2012.
- [31] S. García, S. Ramírez Gallego, J. Luengo, and F. Herrera, "Big Data: Pre procesamiento y calidad de datos."
- [32] L. M. Pechuán, D. R. Pelayo, and J. Brieva, "Reconocimiento de dígitos escritos a mano mediante métodos de tratamiento de imagen y modelos de clasificación," *Res. Comput. Sci.*, vol. 93, no. May, pp. 83–94, 2015.
- [33] A. Arista-Jalife, G. Calderón-Auza, A. Fierro-Radilla, and M. Nakano, "Clasificación de Imágenes Urbanas Aéreas: Comparación entre Descriptores de Bajo Nivel y Aprendizaje Profundo," vol. 28, no. 3, pp. 209–224, 2017.
- [34] O. Gupta, D. Raviv, and R. Raskar, "Illumination invariants in deep video expression recognition," *Pattern Recognit.*, vol. 76, pp. 25–35, 2018.
- [35] Y. Li, X. Nie, and R. Huang, "Web spam classification method based on deep belief networks," *Expert Syst. Appl.*, vol. 96, pp. 261–270, 2018.
- [36] Y. Li *et al.*, "Incorporating knowledge into neural network for text representation," *Expert Syst. Appl.*, vol. 96, pp. 103–114, 2018.
- [37] M. Babaee, D. T. Dinh, and G. Rigoll, "A deep convolutional neural network for video sequence background subtraction," *Pattern Recognit.*, vol. 76, pp. 635–649, 2018.

- [38] F. Huang, C. Jin, Y. Zhang, K. Weng, T. Zhang, and W. Fan, "Sketch-based image retrieval with deep visual semantic descriptor," *Pattern Recognit.*, vol. 76, pp. 537–548, 2018.
- [39] L. Jiao, S. Zhang, L. Li, F. Liu, and W. Ma, "A modified convolutional neural network for face sketch synthesis," *Pattern Recognit.*, vol. 76, pp. 125–136, 2018.
- [40] X. Li, Q. Ding, and J. Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliab. Eng. Syst. Saf.*, vol. 172, no. June 2017, pp. 1–11, 2018.
- [41] J. C. Núñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, "Convolutional Neural Networks and Long Short-Term Memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognit.*, vol. 76, pp. 80–94, 2018.
- [42] H. Tang, B. Xiao, W. Li, and G. Wang, "Pixel convolutional neural network for multi-focus image fusion," *Inf. Sci. (Ny.)*, vol. 433–434, pp. 125–141, 2018.
- [43] Z. Yu, T. Li, G. Luo, H. Fujita, N. Yu, and Y. Pan, "Convolutional networks with cross-layer neurons for image recognition," *Inf. Sci. (Ny.)*, vol. 433–434, pp. 241–254, 2018.
- [44] S. Wang, Z. Zhou, W. Jin, and H. Qu, "Visual saliency detection for RGB-D images under a Bayesian framework," *IPSJ Trans. Comput. Vis. Appl.*, vol. 10, no. 1, p. 1, 2018.
- [45] H. Proen  a and J. C. Neves, "Deep-PRWIS: Periocular Recognition Without the Iris and Sclera Using Deep Learning Frameworks," *IEEE Trans. Inf. Forensics Secur.*, vol. 13, no. 4, pp. 888–896, 2018.
- [46] H. Zhang, L. Liu, Y. Long, L. Shao, and S. Member, "Unsupervised deep hashing with pseudo labels for scalable image retrieval," vol. 27, no. 4, pp. 1626–1638, 2018.
- [47] Y. Zhang, J. Yao, and H. Guan, "Intelligent Cloud Resource Management with Deep Reinforcement Learning," *IEEE Cloud Comput.*, vol. 4, no. 6, pp. 60–69, 2017.
- [48] A. Y. N. Q. V. Le , M. Ranzato, R. Monga , M. Devin , K. Chen , G. S. Corrado , J. Dean, "Building high-level features using large scale unsupervised learning," *2013 IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 8595–8598, 2013.
- [49] Enrique Javier Santiago and Jes  s S  nchez Allende, "Dise  o de un sistema multiagentes h  ibrido basado en aprendizaje profundo para la detecci  n y contenci  n de ciberataques," *Rev. Colomb. Tecnol. Av.*, vol. 2, 2016.
- [50] S. Contreras and F. De La Rosa, "Aplicaci  n de Deep Learning en Rob  tica M  vil para Exploraci  n y Reconocimiento de Objetos basados en Im  genes," *2016 IEEE 11th Colomb. Comput. Conf. CCC 2016 - Conf. Proc.*, 2016.
- [51] R. Gil, "Desarrollo de un sistema de reconocimiento de emociones faciales en tiempo real ´," no. Cvc, 2017.
- [52] L. Ferrado, "Filtrando eventos de seguridad en forma conservativa mediante deep learning," pp. 1–8.
- [53] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Futur. Gener. Comput. Syst.*, vol. 81, pp. 307–313, 2018.

- [54] S. Zhou, Q. Chen, and X. Wang, “Active deep learning method for semi-supervised sentiment classification,” *Neurocomputing*, vol. 120, pp. 536–546, 2013.
- [55] M. T. Vega *et al.*, “Deep Learning for Quality Assessment in Live Video Streaming,” *IEEE Signal Process. Lett.*, vol. 24, no. 6, pp. 736–740, 2017.
- [56] D. Pietro Cavallo, M. Cefola, B. Pace, A. F. Logrieco, and G. Attolico, “Non-destructive automatic quality evaluation of fresh-cut iceberg lettuce through packaging material,” *J. Food Eng.*, vol. 223, pp. 46–52, 2018.
- [57] Y. Ma, Z. Xiang, Q. Du, and W. Fan, “Effects of user-provided photos on hotel review helpfulness: An analytical approach with deep leaning,” *Int. J. Hosp. Manag.*, vol. 71, no. April 2017, pp. 120–131, 2018.
- [58] L. G. Pang, K. Zhou, N. Su, H. Petersen, H. Stöcker, and X. N. Wang, “An equation-of-state-meter of quantum chromodynamics transition from deep learning,” *Nat. Commun.*, vol. 9, no. 1, 2018.
- [59] H. Oh, J. H. Jung, B. C. Jeon, and B. D. Youn, “Scalable and Unsupervised Feature Engineering Using Vibration-Imaging and Deep Learning for Rotor System Diagnosis,” *IEEE Trans. Ind. Electron.*, vol. 65, no. 4, pp. 3539–3549, 2018.
- [60] F.-C. Chen and M. R. Jahanshahi, “NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naïve Bayes Data Fusion,” *IEEE Trans. Ind. Electron.*, vol. 65, no. 5, pp. 4392–4400, 2018.
- [61] M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks,” *Comput. Vision–ECCV 2014*, vol. 8689, pp. 818–833, 2014.
- [62] F. Li, J. Zhang, C. Shang, D. Huang, E. Oko, and M. Wang, “Modelling of a post-combustion CO₂capture process using deep belief network,” *Appl. Therm. Eng.*, vol. 130, pp. 997–1003, 2018.
- [63] Y. Cao *et al.*, “Improving Tuberculosis Diagnostics Using Deep Learning and Mobile Health Technologies among Resource-Poor and Marginalized Communities,” *2016 IEEE First Int. Conf. Connect. Heal. Appl. Syst. Eng. Technol.*, no. 1, pp. 274–281, 2016.
- [64] Z. Jiao, X. Gao, Y. Wang, J. Li, and H. Xu, “Deep Convolutional Neural Networks for mental load classification based on EEG data,” *Pattern Recognit.*, vol. 76, pp. 582–595, 2018.
- [65] Y. Xie, J. Zhang, Y. Xia, M. Fulham, and Y. Zhang, “Fusing texture, shape and deep model-learned information at decision level for automated classification of lung nodules on chest CT,” *Inf. Fusion*, vol. 42, no. October 2016, pp. 102–110, 2018.
- [66] K. J. Pfisterer, R. Amelard, A. G. Chung, and A. Wong, “A new take on measuring relative nutritional density: The feasibility of using a deep neural network to assess commercially-prepared puréed food concentrations,” *J. Food Eng.*, vol. 223, pp. 220–235, 2018.
- [67] A. I. Shahin, Y. Guo, K. M. Amin, and A. A. Sharawi, “White blood cells identification system based on convolutional deep neural learning networks,” *Comput. Methods Programs Biomed.*, vol. 0, pp. 1–12, 2017.
- [68] R. Guan *et al.*, “Multi-label Deep Learning for Gene Function Annotation in Cancer Pathways,” *Sci. Rep.*, vol. 8, no. 1, pp. 1–9, 2018.

- [69] A. Mardt, L. Pasquali, H. Wu, and F. Noé, “VAMPnets for deep learning of molecular kinetics,” *Nat. Commun.*, vol. 9, no. 1, pp. 1–11, 2018.
- [70] A. Tiulpin, J. Thevenot, E. Rahtu, P. Lehenkari, and S. Saarakkala, “Automatic knee osteoarthritis diagnosis from plain radiographs: A deep learning-based approach,” *Sci. Rep.*, vol. 8, no. 1, pp. 1–10, 2018.
- [71] D. Romo-Bucheli, A. Janowczyk, H. Gilmore, E. Romero, and A. Madabhushi, “A deep learning based strategy for identifying and associating mitotic activity with gene expression derived risk categories in estrogen receptor positive breast cancers,” *Cytom. Part A*, vol. 91, no. 6, pp. 566–573, 2017.
- [72] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, “Multimodal deep learning for cervical dysplasia diagnosis,” vol. 1, pp. 115–123, 2016.
- [73] S. Gasparini *et al.*, “Information Theoretic-Based Interpretation of a Deep Neural Network Approach in Diagnosing Psychogenic Non-Epileptic Seizures,” *Entropy*, vol. 20, no. 2, p. 43, 2018.
- [74] H. Gunduz, Y. Yaslan, and Z. Cataltepe, “Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations,” *Knowledge-Based Syst.*, vol. 137, pp. 138–148, 2017.
- [75] S. Rönnqvist and P. Sarlin, “Bank distress in the news: Describing events through deep learning,” *Neurocomputing*, vol. 264, pp. 57–70, Nov. 2017.
- [76] A. Dingli and K. S. Fournier, “Financial Time Series Forecasting – A Deep Learning Approach,” *Int. J. Mach. Learn. Comput.*, vol. 7, no. 5, pp. 118–122, 2017.
- [77] R. Singh and S. Srivastava, “Stock prediction using deep learning,” *Multimed. Tools Appl.*, vol. 76, no. 18, pp. 18569–18584, Sep. 2017.
- [78] W. Bao, J. Yue, and Y. Rao, “A deep learning framework for financial time series using stacked autoencoders and long-short term memory,” *PLoS One*, vol. 12, no. 7, p. e0180944, Jul. 2017.
- [79] Y. Zhao, J. Li, and L. Yu, “A deep learning ensemble approach for crude oil price forecasting,” *Energy Econ.*, vol. 66, pp. 9–16, 2017.
- [80] S. Zhou *et al.*, “Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies,” *Neurocomputing*, vol. 83, no. 6, p. 43, Nov. 2018.
- [81] C. Krauss, X. A. Do, and N. Huck, “Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500,” *Eur. J. Oper. Res.*, vol. 259, no. 2, pp. 689–702, 2017.
- [82] S. Sohangir, D. Wang, A. Pomeranets, and T. M. Khoshgoftaar, “Big Data: Deep Learning for financial sentiment analysis,” *J. Big Data*, vol. 5, no. 1, 2018.
- [83] N. D. Goumagias, D. Hristu-Varsakelis, and Y. M. Assael, “Using deep Q-learning to understand the tax evasion behavior of risk-averse firms,” *Expert Syst. Appl.*, vol. 101, pp. 258–270, 2018.
- [84] Y. Wang and W. Xu, “Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud,” *Decis. Support Syst.*, vol. 105, pp. 87–95, 2018.
- [85] J. Lee, D. Jang, and S. Park, “Deep learning-based corporate performance prediction model considering technical capability,” *Sustain.*, vol. 9, no. 6, pp. 1–

12, 2017.

- [86] Y. Kaneko, "A Deep Learning Approach for the Prediction of Retail Store Sales," *Icdmw*, 2016.
- [87] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, "Deep learning for stock prediction using numerical and textual information," *2016 IEEE/ACIS 15th Int. Conf. Comput. Inf. Sci. ICIS 2016 - Proc.*, 2016.
- [88] O. Chang, I. Naranjo, C. Guerron, D. Criollo, J. Guerron, and G. Mosquera, "A Deep Learning Algorithm to Forecast Sales of Pharmaceutical Products," no. August, 2017.
- [89] J. Sirignano, A. Sadhwani, and K. Giesecke, "Deep Learning for Mortgage Risk," *Ssrn*, pp. 1–83, 2016.
- [90] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Deep Learning for Event-Driven Stock Prediction," no. *Ijcai*, pp. 2327–2333, 2015.
- [91] M. Kraus and S. Feuerriegel, "Decision support from financial disclosures with deep neural networks and transfer learning," *Decis. Support Syst.*, vol. 104, pp. 38–48, 2017.

11. Anexos

Anexo 1: Extracción de datos	88
Área de Conocimiento: INGENIERÍA	88
Área de Conocimiento: INDUSTRIA.....	100
Área de Conocimiento: INVESTIGACIÓN	101
Área de Conocimiento: SALUD	104
Área de Conocimiento: FINANZAS	110
Anexo 2: Tabla integral de las preguntas de investigación.....	120
Anexo 3: Código completo del modelo desarrollado en Keras	122
Anexo 4: Carta de aceptación del artículo	125
Anexo 5: Artículo indexado en Knowledge E	126
Anexo 6: Certificado de Participación en el SIIPRIN'2018.....	140
Anexo 7: Certificado de Participación en el II Ciclo de Conferencias CIS-UNL 2019.....	141
Anexo 8: Declaración de Confidencialidad	142
Anexo 9: Licencia Creative Commons	143

Anexo 1: Extracción de datos

Área de Conocimiento: INGENIERÍA

TABLA XXIII.
RESULTADOS DEL ARTÍCULO EP01

#	Descripción	Detalle	
1	Información bibliográfica	Título	Reconocimiento de dígitos escritos a mano mediante métodos de tratamiento de imagen y modelos de clasificación
		Autor	Luis Miralles Pechuán, Dafne Rosso Pelayo, Jorge Brieva
		Referencia	[32]
		Año	2015
2	Aplicación	Visión Artificial: Reconocimiento y clasificación de dígitos escritos a mano.	
3	Modelos	Redes Neuronales multicapa retroalimentadas (MLP).	
4	Funciones	Se emplearon varias funciones de activación Tanh y ReLu además se probó con capas Dropout, pero los mejores resultados se obtuvieron con la función de activación Tanh y sin capa dropout.	
5	Conclusiones Relevantes	Reconocimiento de dígitos escritos a mano a partir del procesamiento de un conjunto de pixeles que representan a los dígitos, la metodología comprende reducción de la dimensionalidad, extracción de características, selección de variables y la evaluación de varios modelos. La métrica de evaluación fue accuracy, entre los modelos evaluados, aplicando Random Forest sin reducción de la imagen se obtiene una exactitud de 0.7373, mientras que los algoritmos del DL lograron alcanzar un 0.9998 y un 1 de exactitud, empleando todas las características extraídas de la imagen.	

TABLA XXIV.
RESULTADOS DEL ARTÍCULO EP02

#	Descripción	Detalle	
1	Información bibliográfica	Título	Clasificación de Imágenes Urbanas Aéreas: Comparación entre Descriptores de Bajo Nivel y Deep learning
		Autor	Antonio Arista-Jalife, Gustavo Calderón-Auza, Atoany Fierro-Radilla, Mariko Nakano
		Referencia	[33]
		Año	2017
2	Aplicación	Visión Artificial: Reconocimiento y clasificación de imágenes aéreas encontradas en zonas urbanas, de tal manera que pueda ser utilizado en cualquier parte del mundo.	
3	Modelos	Redes Neuronales Convolucionales (CNN).	
4	Funciones	Función de activación sigmoidal.	
5	Conclusiones Relevantes	Se compara algoritmos descriptivos uno es la máquina de soporte vectorial (SVM) y una CNN para DL. En el primer algoritmo a comparar se emplean 3 descriptores diferentes. Para la CNN las imágenes no recibieron ningún preprocesamiento en la entrada. Las métricas de evaluación fueron exactitud y sensibilidad, los resultados muestran que la CNN supera a los otros métodos de comparación alcanzando una exactitud de 0.87, precisión de 0.87 y especificidad de 0.87, mientras que los otros métodos no alcanzan al 0.80.	

TABLA XXV.
RESULTADOS DEL ARTÍCULO EP03

#	Descripción	Detalle	
1	Información bibliográfica	Título	Computational ghost imaging using deep learning
		Autor	Tomoyoshi Shimobaba, Yutaka Endo, Takashi Nishitsuji, Takayuki Takahashi, Yuki Nagahama, Satoki Hasegawa, Marie Sano, Ryuji Hirayama, Takashi Kakue, Atsushi Shiraki, Tomoyoshi Ito
		Referencia	[27]
		Año	2017
2	Aplicación	Visión Artificial: Reconocer imágenes de un solo píxel. Mejorar la calidad de la imagen mediante la predicción de menos ruido de las nuevas imágenes en imágenes de fantasmas computacionales (CGI)	
3	Modelos	Redes Neuronales Convolucional (CNN).	
4	Funciones	Función de activación ReLu y en la última capa emplean sigmoide, además emplearon la técnica de dropout para el entrenamiento	
5	Conclusiones Relevantes	El método propuesto por los autores CGI puede crear imágenes de objetos incluso en entornos ruidosos. La red propuesta se llama U-Net, se usó originalmente para la segmentación de imágenes, pero se puede usar para la restauración. La red consta de: una ruta de construcción y una ruta expansiva. Estas rutas incluyen convolución, agrupamiento máximo y capas de muestreo. La mejora de la calidad de imagen lograda por el método propuesto se evaluó en términos del índice de similitud estructural (SSIM). Las imágenes reconstruidas según el método propuesto se comparan con las obtenidas por CGI diferencial y eliminación de ruido bilateral. Al probar con ocho imágenes que no se incluyeron en el conjunto de datos de entrenamiento, el SSIM promedio del método propuesto fue superior a 0,3 en comparación con solo alrededor de 0,2 para los métodos de CGI diferencial y de eliminación de ruido bilateral.	

TABLA XXVI.
RESULTADOS DEL ARTÍCULO EP04

#	Descripción	Detalle	
1	Información bibliográfica	Título	Illumination invariants in deep video expression recognition
		Autor	Otkrist Gupta, Dan Raviv, Ramesh Raskar
		Referencia	[34]
		Año	2017
2	Aplicación	Visión Artificial: Reconocer y clasificar expresiones faciales. Modelar y parametrizar rostros humanos para la clasificación y comprensión de expresiones humanas en vídeo	
3	Modelos	Redes Neuronales Convolucional (CNN) y Autoencoder.	
4	Funciones	Función de activación ReLu y softmax en la salida para la predicción	
5	Conclusiones Relevantes	Los autores emplean autoencoder para extraer características significativas. La red neuronal de predicción, consiste en una combinación de varias capas convolucionales seguidas por múltiples capas conectadas por completo que terminan en una capa de regresión logística softmax para la predicción. La topología del autoencoder está inspirada en ImageNet, se compone de capas convolucionales que reducen gradualmente la dimensionalidad de los datos hasta que alcanza una capa completamente conectada.	

TABLA XXVII.
RESULTADOS DEL ARTÍCULO EP05

#	Descripción	Detalle	
1	Información bibliográfica	Título	Web spam classification method based on deep belief networks
		Autor	Yuancheng Li, Xiangqian Nie, Rong Huang
		Referencia	[35]
		Año	2017
2	Aplicación	Detección de Spam: Aplica DL para clasificar páginas Web que son spam	
3	Modelos	Redes Creencia Profunda (DBN) y Denoising Autoencoder (SAE).	
4	Funciones	Función de activación Sigmoide, SMOTE técnica inteligente de sobre muestreo que se utiliza en la clasificación de conjuntos de datos desequilibrados.	
5	Conclusiones Relevantes	Los autores emplean el algoritmo SMOTE para equilibrar las proporciones de spams Web y sitios Web legítimo, se emplea el algoritmo SAE para reducir la dimensión en el conjunto de muestra. El clasificador es DBN en el inicio el proceso de inicialización de parámetros de red, cada RBM obtiene los parámetros inicializados y se forma el marco inicial de DBN. Se capacita al DBN para optimizar los parámetros de las capas de red. Las métricas de evaluación consideradas en este estudio son: precisión, recall, F1-Measure y se comparó el modelo con los algoritmos SVM y RF. El método propuesto obtuvo los mejores resultados.	

TABLA XXVIII.
RESULTADOS DEL ARTÍCULO EP06

#	Descripción	Detalle	
1	Información bibliográfica	Título	Incorporating knowledge into neural network for text representation
		Autor	Yiming Li, Baogang Wei, Yonghuai Liu, Liang Yao, Hui Chen, Jifang Yu, Wenhao Zhu
		Referencia	[36]
		Año	2017
2	Aplicación	Procesamiento del Lenguaje Natural: Emplea DL para obtener características. Representar numéricamente los documentos de texto no estructurados para que puedan computarse matemáticamente.	
3	Modelos	Redes neuronales convolucionales (CNN), red de memoria a largo plazo (LSTM)	
4	Funciones	Softmax en la última capa.	
5	Conclusiones Relevantes	Emplean Text Concept Vector (TCV) para la representación de texto, donde extraen la información de nivel de concepto del texto, reconocen las entidades en el texto a través del algoritmo Backward Maximum Matching. Para determinar el sentido de la entidad, el texto sin procesar se transforma en texto conceptual que se compone de un conjunto de conceptos. La identificación del texto y sus conceptos se asignan por separado a un vector único. Implementan una red neuronal convolucional (CNN) para entrenar estos dos tipos de vectores obteniendo características de nivel de concepto del texto original y aplicándolas a tareas de procesamiento de lenguaje natural (NLP), como la clasificación de texto. El método propuesto extrae conceptos de un texto en bruto y los transforma en texto conceptualizado. Al implementar la red neuronal CNN se extrae la información de nivel conceptual además de considerar el orden de los términos.	

TABLA XXIX.
RESULTADOS DEL ARTÍCULO EP07

#	Descripción	Detalle	
1	Información bibliográfica	Título	A deep convolutional neural network for video sequence background subtraction
		Autor	Mohammadreza Babaee, Duc Tung Dinh, Gerhard Rigoll
		Referencia	[37]
		Año	2017
2	Aplicación	Visión artificial: Extraer y clasificar características para eliminar fondo de videos	
3	Modelos	Redes neuronales convolucionales (CNN)	
4	Funciones	Función de activación Sigmoide y ReLu. Empleando sigmoide en la última capa para integrar la red.	
5	Conclusiones Relevantes	La metodología empieza con la implementación de un algoritmo de generación de modelos de fondo para utilizar una imagen de fondo y extraer el fondo, el algoritmo genera pares de imágenes que corresponden al fondo y los fotogramas de video se extraen y se introducen en la CNN para restar el fondo creando por el algoritmo anterior. Después de volver a ensamblar los parches en la salida de la red profunda, se post-procesa segmentando el cuadro de video. El método propuesto se combina con el algoritmo SuBSENSE, para mejorar las actualizaciones del modelo de fondo.	

TABLA XXX.
RESULTADOS DEL ARTÍCULO EP08

#	Descripción	Detalle	
1	Información bibliográfica	Título	Sketch-based image retrieval with deep visual semantic descriptor
		Autor	Fei Huang, Cheng Jin, Yuejie Zhang, Kangnian Weng, Tao Zhang, Weiguo Fan
		Referencia	[38]
		Año	2017
2	Aplicación	Visión artificial: Extraer y clasificar características. Recuperación de imágenes con miles de millones de imágenes disponibles tanto en línea como fuera de línea.	
3	Modelos	Redes neuronales convolucionales (CNN), Autoencoder (AE)	
4	Funciones	Softmax en la última capa. Función de activación Sigmoide y ReLu.	
5	Conclusiones Relevantes	Para la transformación similar a la de un boceto los contornos gruesos son detectados por bosques estructurado, en la representación de características emplean: un AE bimodal es decir que incluye codificador y decodificador para reducir la dimensionalidad, las salidas del AE se conectan a una CNN para generar características profundas y semánticas. Este modelo aplica DL en un descriptor semántico visual profundo que codifica características de bajo y alto nivel en bocetos transformados a partir de imágenes naturales originales.	

TABLA XXXI.
RESULTADOS DEL ARTÍCULO EP09

#	Descripción	Detalle	
1	Información bibliográfica	Título	A modified convolutional neural network for face sketch synthesis
		Autor	Licheng Jiao, Sibo Zhang, Lingling Li, Fang Liu, Wenping Ma
		Referencia	[39]
		Año	2017
2	Aplicación	Visión artificial: Aplica DL para extraer las características de las fotos a través de la detección y clasificación de patrones para elaborar bocetos (síntesis). Construir un mapeo de fotos a boceto.	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLu	
5	Conclusiones Relevantes	Proponen una red neuronal convolucional de cuatro capas modificada, llamada Face Sketch Synthesis Network (FSSN), la primera capa se usa para extraer las características de las fotos, se conecta una capa de pooling para aliviar la desalineación, una capa no lineal, llamada capa convolucional de perceptrón multicapa (mlpconv) que transforma los mapas de características de fotografía a boceto, se utiliza una capa convolucional para sintetizar el boceto final. En la fase de entrenamiento, todos los filtros y parámetros se aprenden de los pares de entrenamiento por el algoritmo de retro propagación. En la fase de prueba, las fotos se transmiten capa por capa hasta que se obtienen los resultados finales.	

TABLA XXXII.
RESULTADOS DEL ARTÍCULO EP10

#	Descripción	Detalle	
1	Información bibliográfica	Título	Remaining useful life estimation in prognostics using deep convolution neural networks
		Autor	Xiang Li, Qian Ding, Jian-Qiao Sun
		Referencia	[40]
		Año	2017
2	Aplicación	Visión artificial: Estimación de la vida útil remanente (RUL). Mantenimiento y pronóstico de vida útil de unidades de motor de aeronáutica.	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación Tanh. Algoritmo de propagación inversa (BP), actualiza los parámetros del modelo propuesto para minimizar el error de entrenamiento.	
5	Conclusiones Relevantes	El método DL propuesto consta de dos subestructuras: redes neuronales de convolución múltiple y capa de regresión totalmente conectada. La técnica de dropout se utiliza en la capa aplanada, para aliviar el sobreajuste. Se adjunta una neurona al final de la red propuesta para la estimación RUL. Todas las capas usan tanh como funciones de activación, y el inicializador normal de Xavier se emplea para las inicializaciones de peso. Para mejorar el rendimiento del pronóstico, se aplica un proceso de ajuste emplean el BP. El algoritmo de Adam se emplea para la optimización.	

TABLA XXXIII.
RESULTADOS DEL ARTÍCULO EP11

#	Descripción	Detalle	
1	Información bibliográfica	Título	Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition.
		Autor	Juan C. Núñez, Raúl Cabido, Juan J. Pantrigo, Antonio S. Montemayor, José F. Vélez
		Referencia	[41]
		Año	2017
2	Aplicación	Visión artificial: Detectar y capturar patrones. Reconocer la actividad que esté realizando un sujeto en una imagen.	
3	Modelos	Redes neuronales convolucionales (CNN), red de memoria a largo plazo (LSTM)	
4	Funciones	Función de activación ReLu.	
5	Conclusiones Relevantes	La arquitectura combina una red neuronal convencional (CNN) seguida de una red recurrente de larga memoria a corto plazo (LSTM). Las CNN se centrarán en la detección de patrones espaciales de la posición de las articulaciones esqueléticas en el espacio tridimensional. La red recurrente LSTM se utiliza para capturar patrones espaciotemporales relacionados con la evolución temporal de las coordenadas tridimensionales de las articulaciones esqueléticas. Primero pre-entrenan la CNN independientemente de la LSTM. Emplean un perceptrón multicapa (MLP) para conectarlo a la CNN. Con la CNN pre-entrenado, conectan la salida al LSTM y finalizamos la etapa de capacitación.	

TABLA XXXIV.
RESULTADOS DEL ARTÍCULO EP12

#	Descripción	Detalle	
1	Información bibliográfica	Título	Pixel convolutional neural network for multi-focus image fusion
		Autor	Han Tang, Bin Xiao, Weisheng Li, Guoyin Wang
		Referencia	[42]
		Año	2017
2	Aplicación	Visión artificial: Clasificar cada píxel en las imágenes de origen en tres clases: centrado, desenfocado y desconocido. Superar la capacidad de obtener la medida de enfoque hecha a mano.	
3	Modelos	Redes neuronales convolucionales (CNN)	
4	Funciones	No específica	
5	Conclusiones Relevantes	La CNN supervisada necesita muchas imágenes de entrenamiento etiquetadas, crearon un conjunto de imágenes que incluyen imágenes enfocadas y desenfocadas, correctamente etiquetadas empleando el dataset Cifar-10 desenfocando con desviación estándar y filtros Gaussianos. Emplean una red neuronal convolucional de pixel (p-CNN) cuya estructura es: entrada la imagen (parche) de tamaño 32×32 , y la salida la puntuación de las tres clases (enfocada, desenfocada y desconocida). Pruebas: emplean 290,000 imágenes generadas por las 10,000 imágenes de prueba en Cifar-10. Obtienen el mapa de decisión final que marca las imágenes desenfocadas y enfocadas, se aplica un proceso de eliminación en el cuál si la zona desenfocada (puntos negros) es más pequeña que las que la rodean se reemplaza con una zona enfocada e igualmente a la inversa con las zonas enfocadas (puntos blancos).	

TABLA XXXV.
RESULTADOS DEL ARTÍCULO EP13

#	Descripción	Detalle	
1	Información bibliográfica	Título	Convolutional networks with cross-layer neurons for image recognition
		Autor	Zeng Yu, Tianrui Li, Guangchun Luo, Hamido Fujita, Ning Yu, Yi Pan
		Referencia	[43]
		Año	2017
2	Aplicación	Visión artificial: Extraer y clasificar imágenes. Diseñar una red DL que integre todas las características aprendidas. Resolver el problema de gradiente de fuga cuando la red se vuelve muy profunda. Mejorar el reconocimiento de imágenes utilizando toda la información generada por los extractores de características	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLu	
5	Conclusiones Relevantes	En este trabajo no se realiza un pre-procesamiento de la imagen. El modelo propuesto: el bloque de capas cruzadas consiste en una red principal, una capa cruzada y una capa cruzada de neuronas. La red principal adopta dos capas de convolución consecutivas (3×3) para construir su estructura de construcción. Después de cada capa de convolución, normalizamos la salida con normalización de lote (BN) y usamos ReLU como la función de activación no lineal. La capa cruzada es un enlace que entrega la información desde una capa de nivel inferior a una capa de nivel superior. Pruebas, luego de realizar las pruebas, se ajustan los parámetros de acuerdo con el conjunto de validación. Construyen un nuevo modelo neuronal profundo denominado redes de neuronas de capa cruzada (CLNN) para resolver parcialmente el problema del gradiente de fuga, también acelerar el proceso de clasificación con neuronas de capa cruzada.	

TABLA XXXVI.
RESULTADOS DEL ARTÍCULO EP14

#	Descripción	Detalle	
1	Información bibliográfica	Título	Visual saliency detection for RGB-D images under a Bayesian framework
		Autor	Songtao Wang, Zhen Zhou, Wei Jin and Hanbing Qu
		Referencia	[44]
		Año	2018
2	Aplicación	Visión artificial: Extraer y clasificar características. Detección de proximidad, resaltar regiones u objetos visualmente destacados en una imagen.	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLu	
5	Conclusiones Relevantes	Entrena una CNN para imágenes de profundidad enseñando a la red a reproducir la representación semántica de nivel medio aprendida de imágenes RGB para las cuales hay imágenes emparejadas. Luego, las características profundas del RGB y las imágenes de profundidad son extraídas por una CNN. Los experimentos verifican que las características profundas de las imágenes de profundidad pueden servir como un complemento útil para las características profundas de las imágenes RGB dentro de un marco bayesiano.	

TABLA XXXVII.
RESULTADOS DEL ARTÍCULO EP15

#	Descripción	Detalle	
1	Información bibliográfica	Título	DEEP-PRWIS: Periocular recognition without the iris and sclera using deep learning frameworks
		Autor	Hugo Proença , y João C. Neves
		Referencia	[45]
		Año	2017
2	Aplicación	Visión artificial: Clasificar imágenes. Reconocimiento encubierto de humanos es decir al aire libre y no cooperativo.	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLu, emplean dropout para el entrenamiento y softmax en la capa final.	
5	Conclusiones Relevantes	Los datos se obtuvieron intercambiando imágenes de Regiones Ocular / Periocular: Usando segmentación, obtienen dos máscaras binarias Bi y Bj que discriminan entre el ocular (iris y esclerótica) y los componentes, además, se utilizaron otras dos transformaciones que preservan la etiqueta para fines de aumento de datos. Como modelo se emplea una arquitectura CNN basada en Alexnet: datos de entrada imágenes RGB $150 \times 200 \times 3$, pasan por 6 capas de convolución, 3 capas de pooling y dropout. Todas las capas convolucionales son adyacentes a las funciones de activación de la Unidad Lineal Rectificada (ReLU)	

TABLA XXXVIII.
RESULTADOS DEL ARTÍCULO EP16

#	Descripción	Detalle	
1	Información bibliográfica	Título	Unsupervised deep hashing with pseudo labels for scalable image retrieval
		Autor	Haofeng Zhang , Li Liu, Yang Long, and Ling Shao
		Referencia	[46]
		Año	2017
2	Aplicación	Visión artificial: Extracción de características. Recuperación de información visual	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLU, función tanh en la capa final	
5	Conclusiones Relevantes	El modelo de hashing profundo se inicializa con los parámetros pre entrenados de VGG-16, en este punto el modelo está entrenado para el conjunto de datos de ImageNet cuyas características visuales no se aplican para aprender códigos de hash, por lo que se entrena una red CNN con cinco capas convolucionales cada una seguida implementa la función de activación ReLU y las capas de pooling; seguidas de dos capas completamente conectadas la primera con la función de activación ReLU y la última capa se sustituye por el código binario y con la función de activación Tanh para restringir los códigos aprendidos dentro del rango (-1,1). Los códigos son asignados empleando un algoritmo de pseudoetiquetas.	

TABLA XXXIX.
RESULTADOS DEL ARTÍCULO EP17

#	Descripción	Detalle	
1	Información bibliográfica	Título	Intelligent cloud resource management with deep reinforcement learning
		Autor	Yu Zhang, Jianguo Yao, and Haibing Guan
		Referencia	[47]
		Año	2017
2	Aplicación	Clasificación de servicios de nube: Toma de decisiones a través de la extracción de características y clasificación. Optimización de la configuración del servicio en la nube y la asignación adaptativa de los recursos	
3	Modelos	Autoencoder apilado (SAE), aprendizaje por refuerzo Q-learning.	
4	Funciones	No especifica.	
5	Conclusiones Relevantes	La red neuronal profunda está preajustada a través del autoencoder apilado (SAE), seguido del uso de experiencias de aprendizaje de refuerzo para la optimización. El controlador, el monitor y el asignador se coordinan entre sí para asignar recursos de forma inteligente, respetando dos restricciones: los requisitos de QoS deben cumplirse y la cantidad de uso de los recursos debe ser menor que el total cantidad de recursos disponibles en el sistema.	

TABLA XL.
RESULTADOS DEL ARTÍCULO EP18

#	Descripción	Detalle	
1	Información bibliográfica	Título	Building high-level features using large scale unsupervised learning
		Autor	Quoc V. Le, Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S. Corrado, Je Dean, Andrew Y. Ng
		Referencia	[48]
		Año	2012
2	Aplicación	Visión artificial: Clasificar caras con presencia de distractores. Construir detectores de funciones específicas de clase de alto nivel a partir de imágenes sin etiqueta	
3	Modelos	Autoencoder.	
4	Funciones	Función de activación Síguide.	
5	Conclusiones Relevantes	En la metodología aplicada en este estudio ejecutan los siguientes pasos: - Obtener dataset: Marcos de muestreo de 10 millones de videos de YouTube: cada ejemplo es una imagen en color con 200x200 píxeles. -Aplicar modelo: Arquitectura empleada autoencoder profundo se construye replicando tres veces la misma etapa compuesta de filtrado local, agrupación local y normalización de contraste local. La salida de una etapa es la entrada a la siguiente y el modelo general se puede interpretar como una red de nueve capas. -Pruebas se aplica al dataset de pruebas, para medir el rendimiento de cada neurona al clasificar las caras contra los distractores. Para cada neurona, encontramos sus valores de activación máximos y mínimos. Este estudio muestra que es posible entrenar a las neuronas para que sean selectivas para los conceptos de alto nivel utilizando datos totalmente no etiquetados.	

TABLA XLI.
RESULTADOS DEL ARTÍCULO EP19

#	Descripción	Detalle	
1	Información bibliográfica	Título	Diseño de un sistema multiagentes híbrido basado en Deep learning para la detección y contención de ciberataques
		Autor	Enrique Javier Santiago, Jesús Sánchez Allende
		Referencia	[49]
		Año	2016
2	Aplicación	Detección de ciberataques: Clasificar el tráfico monitorizado como normal, ilegal o sospechoso. Detección y contención de ciberataques.	
3	Modelos	Redes Neuronales multicapa retroalimentadas (MLP).	
4	Funciones	Función de activación Sigmoidal y tanh.	
5	Conclusiones Relevantes	La solución se divide en 3 capas: capa de monitorización, compuesta por un grupo de agentes reactivos compuestos por los siguientes módulos: lectura del tráfico de las interfaces de red, pre-procesador de los diferentes flujos de tráfico, analizador de tráfico, forense, de entrada/salida, defensa, un módulo de comunicaciones. Capa de análisis: Los agentes inteligentes de esta capa están compuestos por los módulos: de comunicaciones a través del cual interactúan con los agentes reactivos, de aprendizaje adaptativo compuesto por dos redes neuronales multicapa, de clasificación del tráfico y de registro de actividades. Y una capa de supervisión: está compuesta por la instancia de uno o más agentes reactivos que tienen como función recibir flujos de información referentes al registro de actividades de cada agente activo de todas las capas de la solución. La solución de ciberseguridad propuesta es considerada escalable, distribuida, tolerante a fallos y con la capacidad de adaptarse a los cambios de técnicas y estrategias usadas por los ciberagresores.	

TABLA XLII.
RESULTADOS DEL ARTÍCULO EP20

#	Descripción	Detalle	
1	Información bibliográfica	Título	Aplicación de deep learning en robótica móvil para exploración y reconocimiento de objetos basados en imágenes
		Autor	Stevenson Contreras, Fernando de la Rosa
		Referencia	[50]
		Año	2016
2	Aplicación	Robótica: Reconocer y aprender estructuras físicas que permitan a un robot navegar de manera segura en su espacio. Simular un comportamiento inteligente más natural.	
3	Modelos	Redes Neuronales multicapa retroalimentadas (MLP).	
4	Funciones	Función de activación Sigmoidal y tanh.	
5	Conclusiones Relevantes	Plantean un sistema compuesto por dos redes neuronales profundas (DNN) controlado principalmente por el módulo NeuralDriver, el cual integra los componentes ImageProcessor y RobotSystem que se utilizan para poder obtener la información de la cámara y el robot respectivamente. Para entrenar las redes emplean el módulo Training que contiene librerías para realizar todo el proceso de aprendizaje y obtener todos los datos de la red neuronal. La información resultante se reporta a TargetManager, encargado de persistir la información de la red neuronal entrenada y/o generar una acción para el modo Online. En modo Online (tiempo real) se obtiene la información de los sensores (cámara y láser) para reportar la información obtenida y dar las órdenes de acción al robot. Resultando dos redes neuronales: una de reconocimiento de imágenes con un alto nivel de confianza y otra de reconocimiento de situaciones de proximidad a obstáculos con un nivel de confianza aceptable.	

TABLA XLIII.
RESULTADOS DEL ARTÍCULO EP21

#	Descripción	Detalle	
1	Información bibliográfica	Título	Desarrollo de un sistema de reconocimiento de emociones faciales en tiempo real
		Autor	Héctor Ribes Gil
		Referencia	[51]
		Año	2017
2	Aplicación	Visión artificial: Clasificar emociones en una imagen, en tiempo real. Reconocer emociones faciales a partir de una fuente visual, implica reconocimiento facial, detectar puntos de interés y análisis sobre un conjunto de imágenes etiquetadas	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLu en las capas de convolución, Sigmoid para capas ocultas, dropout para entrenar la red y softmax en la capa de salida del modelo.	
5	Conclusiones Relevantes	Todas las etiquetas de los datos se han interpretado para poder unirlas y tener las etiquetas del proyecto, catalogadas por números. Detector de rostros metodologías FaceDetector, haar cascade. Detectar puntos de interés se utilizó KeyPoints. Detector de emociones, para clasificar las emociones se emplean una SVM, una NN con retro propagación, una CNN cuya entrada es la imagen del rostro detectada, normalizada de 0 a 1 y tratada con PCA	

TABLA XLIV.
RESULTADOS DEL ARTÍCULO EP22

#	Descripción	Detalle	
1	Información bibliográfica	Título	Filtrando eventos de seguridad en forma conservativa mediante deep learning
		Autor	Leandro Ferrado and Matías Cuenca-Acuna
		Referencia	[52]
		Año	2016
2	Aplicación	Detección de amenazas de seguridad: Filtrar un flujo de eventos para detectar amenazas de seguridad.	
3	Modelos	Autoencoders apilados (SAE).	
4	Funciones	Función de activación ReLu. Softmax en la capa de salida del modelo.	
5	Conclusiones Relevantes	Emplean un SAE entrenado de forma no supervisada, con la SAE pre entrenado se sigue el entrenamiento supervisado. Para la evaluación del modelo, se busca que el modelo propuesto cumpla con: descartar eventos que suponen comportamiento normal. Retener eventos que no se suponen como normal.	

TABLA XLV.
RESULTADOS DEL ARTÍCULO EP23

#	Descripción	Detalle	
1	Información bibliográfica	Título	A robust human activity recognition system using smartphone sensors and deep learning
		Autor	Mohammed Mehedi Hassan, Md. Zia Uddin, Amr Mohamed, Ahmad Almogren
		Referencia	[53]
		Año	2017
2	Aplicación	Clasificación de actividad humana: Clasificación de las características obtenidas, para reconocer actividad humana basada en los datos de los sensores de teléfonos inteligentes.	
3	Modelos	Redes Creencia Profunda (DBN).	
4	Funciones	Función de activación Síguoide.	
5	Conclusiones Relevantes	Los autores afirman que es altamente factible usar teléfonos inteligentes para el reconocimiento de actividades, por el alto uso que tienen en la vida diaria. Durante la detección se recoge los datos que ingresan al sistema HAR, la extracción de características aplica eliminación de ruido, análisis estadístico PCA para generar características robustas. Las características robustas se combinan con la técnica DBN para entrenamiento y reconocimiento de actividades.	

TABLA XLVI.
RESULTADOS DEL ARTÍCULO EP24

#	Descripción	Detalle	
1	Información bibliográfica	Título	Active deep learning method for semi-supervised sentiment classification
		Autor	Shusen Zhou, Qingcai Chen, Xiaolong Wang
		Referencia	[54]
		Año	2013
2	Aplicación	Análisis de sentimientos: Determinar sentimientos en textos para determinar que textos son relevantes y obtener resúmenes.	
3	Modelos	Máquina restringida de Boltzmann (RBM)	
4	Funciones	No específica	
5	Conclusiones Relevantes	Los dataset empleados en este estudio incluyen datos de revisión de películas, revisiones de libros, DVD, electrónica, electrodomésticos de cocina; cada conjunto de datos contiene 1000 comentarios positivos y 1000 comentarios negativos. En cuanto al modelo activo de red neuronal profunda propuesto, su arquitectura es similar a los DBN, pero con una función de pérdida diferente introducida en la fase de aprendizaje supervisado. El modelo emplea RBM en el pre-entrenamiento se aplica entrenamiento semi-supervisado por tanto captura datos etiquetados y sin etiquetar. En el entrenamiento se extrae una pequeña cantidad de datos no etiquetados para realizar un etiquetado manual y se entrena con los datos etiquetados y sin etiquetar.	

TABLA XLVII.
RESULTADOS DEL ARTÍCULO EP25

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep learning for quality assessment in live video streaming
		Autor	María Torres Vega, Decebal Constantin Mocanu, Jeroen Famaey, Stavros Stavrou y Antonio Liotta
		Referencia	[55]
		Año	2017
2	Aplicación	Detectar calidad de video: Clasificar videos dañados y buenos. Automatizar y optimizar el proceso de evaluación de calidad de video.	
3	Modelos	Redes Creencia Profunda (DBN)	
4	Funciones	No específica	
5	Conclusiones Relevantes	Los datos se obtienen de secuencias de video originales. Cada muestra se compone de 8 características extraídas de la secuencia de video original. Con duración de entre 2 y 10 segundos, según el proveedor de video. El entrenamiento se realiza en tiempo real, mientras se lleva a cabo una sesión de transmisión, el cliente realiza la extracción en tiempo real de las ocho características en la transmisión, el cliente promedia los resultados. Estos ocho valores sirven como entrada para el DBN. El modelo emite los valores estimados correspondientes para la versión dañada. El método propuesto es adaptable debido al aprendizaje no supervisado, no requiere datos de entrenamiento etiquetados. Por lo tanto, es mucho más rápido y más fácil de adaptar, a medida que se agregan nuevos videos.	

Área de Conocimiento: INDUSTRIA

TABLA XLVIII.
RESULTADOS DEL ARTÍCULO EP26

#	Descripción	Detalle	
1	Información bibliográfica	Título	Non-Destructive automatic quality evaluation of fresh-cut iceberg lettuce through packaging material
		Autor	Dario Pietro Cavallo, Maria Cefola, Bernardo Pace, Antonio Francesco Logrieco, Giovanni Attolico.
		Referencia	[56]
		Año	2017
2	Aplicación	Detectar calidad de un producto: Emplean visión artificial para clasificar las lechugas de acuerdo al nivel de calidad de la lechuga envasada recién cortada.	
3	Modelos	Redes neuronales convolucionales (CNN).	
4	Funciones	Función de activación ReLu. Softmax en la capa de salida del modelo. Además de emplear la técnica dropout para el entrenamiento.	
5	Conclusiones Relevantes	Los artículos se clasificaron según una escala de cinco niveles de calidad (Muy bueno, bueno, aceptable, pobre, muy pobre). Para cada bolsa, se adquirieron dos imágenes: la primera era de toda la bolsa cerrada con la lechuga recién cortada dentro; el segundo fue del producto solo, después de retirar la bolsa. La parte relevante de la lechuga recién cortada empaquetada fue seleccionada usando una CNN entrenada para separar 3 clases diferentes (paquete, producto, artefacto). Todos los píxeles etiquetados como paquete o artefacto fueron eliminados. Esta arquitectura no necesitó intervención humana para seleccionar características que describan el color y la textura o para ajustar los parámetros de la red. La CNN logró con éxito seleccionar la región adecuada para ser procesada para clasificar el nivel de calidad	

TABLA XLIX.
RESULTADOS DEL ARTÍCULO EP27

#	Descripción	Detalle	
1	Información bibliográfica	Título	Effects of user-provided photos on hotel review helpfulness: an analytical approach with deep learning
		Autor	Yufeng Maa, Zheng Xiangb, Qianzhou Duc, Weiguo Fand
		Referencia	[57]
		Año	2017
2	Aplicación	Recomendación de hoteles: Comprender los efectos de las fotos provistas por el usuario. Predecir en qué medida las fotos proporcionadas por el usuario influyen en las revisiones de hoteles en línea.	
3	Modelos	Redes neuronales convolucionales (CNN), red de memoria a largo plazo (LSTM).	
4	Funciones	Función de activación Sigmoid, Softmax en la capa de salida.	
5	Conclusiones Relevantes	<p>Se procesan dos tipos de información: contenido textual e imágenes: Para el contenido textual, se aplicaron procedimientos PNL, como tokenización y derivación. Para procesar las imágenes, se empleó CNN profunda para extraer las características de imagen. Despues se fusionaron a través del módulo LSTM. A diferencia de la capacidad del DL para el modelado de secuencias, los clasificadores como árbol de decisión (DT), máquina de vectores de soporte (SVM), regresión logística (RL) solo aceptan características de entrada de longitud fija. Por tanto, los autores consideran que, con la creciente cantidad de contenidos generados por usuarios en Internet, los modelos del DL tendrán relevancia en muchos escenarios de aplicación y un amplio rango de potencial para ayudar a entender muchos de los problemas de investigación emergentes y de larga data en hospitalidad y turismo.</p>	

Área de Conocimiento: INVESTIGACIÓN

TABLA L.
RESULTADOS DEL ARTÍCULO EP28

#	Descripción	Detalle	
1	Información bibliográfica	Título	An equation – of – state - meter of quantum chromodynamics transition from deep learning
		Autor	Long-Gang Pang, Kai Zhou, Nan Su, Hannah Petersen, Horst Stöcker y Xin-Nian Wang
		Referencia	[58]
		Año	2018
2	Aplicación	Detectar ecuaciones: Detectar patrones y clasificar información para identificar la ecuación de estado (EoS) empleada en las simulaciones hidrodinámicas relativistas de colisiones de iones pesados.	
3	Modelos	Redes Neuronales Convolucionales CNN	
4	Funciones	Función de activación ReLu.	
5	Conclusiones Relevantes	<p>Los autores construyen una CNN y la entrena con pT, ϕ de pions cargados generados a partir del programa hidrodinámico relativista CLVisc 30,31 con dos EoS diferentes como entrada: crossover32 y first order33. Luego, se capacita a la CNN con supervisión para identificar las EoS diferentes. Concluyen que la CNN profunda proporciona un decodificador poderoso y eficiente desde el cual la información EoS se puede extraer directamente. El método de análisis de diferencia de predicción se emplea para extraer las características más relevantes para la tarea de clasificación, esto se puede aplicar en estudios fenomenológicos y experimentales.</p>	

TABLA LI.
RESULTADOS DEL ARTÍCULO EP29

#	Descripción	Detalle	
1	Información bibliográfica	Título	Scalable and unsupervised feature engineering using vibration-imaging and deep learning for rotor system diagnosis
		Autor	Hyunseok Oh , Joon Ha Jung, Byung Chul Jeon, and Byeng Dong Youn
		Referencia	[59]
		Año	2017
2	Aplicación	Diagnóstico de rotor: Emplea visión artificial para la extracción de características de alto nivel. Diagnóstico del sistema de rotor que incorpora datos de un banco de pruebas a pequeña escala y de sistemas de campo reales.	
3	Modelos	Redes Creencia Profunda (DBN).	
4	Funciones	No especifica.	
5	Conclusiones Relevantes	El método empleado por los autores se divide en tres etapas: Generación de imágenes de vibración, las señales de vibración se recogen de los sensores en los sistemas del rotor, el método de imagen por vibración empleado detecta fallas en los sistemas de rotor, se procesan para mejorar la calidad de las imágenes. Extracción de características no supervisadas, se extraen características de alto nivel empleando redes de creencia profunda (DBN). Diseño del clasificador de fallas, se ajusta al combinarlo con un perceptrón multicapa (MLP).	

TABLA LII.
RESULTADOS DEL ARTÍCULO EP30

#	Descripción	Detalle	
1	Información bibliográfica	Título	NB-CNN: Deep learning-based crack detection using convolutional neural network and naive bayes data fusion
		Autor	Fu-Chen Chen and Mohammad R. Jahanshahi
		Referencia	[60]
		Año	2017
2	Aplicación	Detectar grietas: Emplea visión artificial para clasificar imágenes con presencia o no de fisuras o grietas.	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ELU (unidad lineal exponencial). Se emplea dropout para el entrenamiento. Softmax en la capa de salida.	
5	Conclusiones Relevantes	Se captura el video, para la detección de parche de grietas emplean CNN, cada cuadro de video se escanea con parches de tamaño 120×120 píxel, la CNN propuesta clasifica cada parche como un parche de crack o noncrack. Para determinar si un tubo es una fisura o no, se emplean Naïve Bayes. La arquitectura de la CNN propuesta consiste en: La entrada es datos tridimensionales: un parche de imagen de 120×120 de canales R, G y B. Se aplica normalización a la imagen. Los datos pasan por diferentes capas en cada capa se aplica la convolución, normalización por bloques, se emplea la función de activación ELU y se aplica poling en la capa final se emplea softmax. Los autores concluyen que las CNN necesitan muchos datos de entrenamiento para hacer que la capacitación converja y evitar el sobreajuste.	

TABLA LIII.
RESULTADOS DEL ARTÍCULO EP31

#	Descripción	Detalle	
1	Información bibliográfica	Título	Visualizing and understanding convolutional networks
		Autor	Matthew D. Zeiler, Rob Fergus
		Referencia	[61]
		Año	2013
2	Aplicación	Visión artificial: Entender la operación interna y el comportamiento de las CNN y diagnosticar problemas potenciales con el modelo.	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ReLu. Emplean Softmax en la última capa	
5	Conclusiones Relevantes	<p>Para examinar las CNN, adjuntan una red deconvolucional (DCNN) a cada una de las capas convolucionales, proporcionando una ruta continua a los píxeles de la imagen. (1) la imagen de entrada al CNN se envía con características calculadas a través de las capas. Para examinar una activación dada de una CNN, todas las demás activaciones en la capa deben ser cero y pasan los mapas de características como entrada a la capa DCNN adjunta. Luego, sucesivamente (i) desconectar, (ii) rectificar y (iii) reconstruir la actividad en la capa inferior que dio lugar a la activación elegida. Esto se repite hasta que se alcanza el espacio de entrada de píxeles.</p> <p>Este estudio revela que las características no son patrones aleatorios e imposibles de interpretar. Por el contrario, muestran muchas propiedades intuitivamente deseables.</p>	

TABLA LIV.
RESULTADOS DEL ARTÍCULO EP32

#	Descripción	Detalle	
1	Información bibliográfica	Título	Modelling of a post-combustion CO ₂ capture process using deep belief network
		Autor	Fei Li, Jie Zhang, Chao Shang, Dexian Huang, Eni Oko, Meihong Wang
		Referencia	[62]
		Año	2017
2	Aplicación	Captura de CO₂: Extraer y clasificar características. Aplicar DL al modelado de procesos de captura de CO ₂ posterior a la combustión.	
3	Modelos	Redes Creencia Profunda (DBN)	
4	Funciones	No especifica.	
5	Conclusiones Relevantes	<p>Datos: de los datos que se obtienen en la captura de CO₂ para este estudio se emplean: nivel de captura de CO₂ y la tasa de producción de CO₂, para cada variable se debe escalar a media cero y varianza unitaria. Modelo: DBN, todos los datos de entrada se usan para el proceso de capacitación no supervisado para extraer características. Se construyen dos modelos DBN para las predicciones de calidad de la tasa de producción de CO₂ y el nivel de captura de CO₂. La validación cruzada se usa para seleccionar la arquitectura de red, cada DBN tiene 26 nodos de entrada, 20 nodos ocultos en la primera capa oculta, 17 nodos ocultos en la segunda capa oculta y 1 nodo de capa de salida.</p>	

Área de Conocimiento: SALUD

TABLA LV.
RESULTADOS DEL ARTÍCULO EP33

#	Descripción	Detalle	
1	Información bibliográfica	Título	Improving tuberculosis diagnostics using deep learning and mobile health technologies among resource-poor and marginalized communities
		Autor	Yu Cao, Chang Liu, Benyuan Liu, Maria J. Brunette, Ning Zhang, Tong Sun, Peifeng Zhang, Jesús Peinado, Epifanio Sánchez Garavito, Leonid Lecca García, Walter H. Curioso
		Referencia	[63]
2	Aplicación	Diagnóstico médico: Emplea visión artificial para detectar presencia de tuberculosis mediante clasificación de imágenes. Reducir el tiempo de diagnóstico de la enfermedad para zonas de escasos recursos	
		3 Modelos	Red neuronal convolucional (CNN).
		4 Funciones	No especifica
		5 Conclusiones Relevantes	Recolectaron 5.000 imágenes de radiografía de tórax tomadas de pacientes con TB del mundo real con descripciones detalladas de detección de TB, la segunda fuente es ImageCLEF 2004-2013, con 400,000 imágenes médicas, anotaciones diagnósticas, temas de búsqueda y juicios de relevancia. Para cada imagen se extrae regiones de la imagen usando búsqueda selectiva. Se generan las características globales y locales de las imágenes con características de Gabor, características de patrones binarios locales. Para el modelo CNN se emplean entrenamiento supervisado; se ajusta la función de CNN para detección usando un conjunto de datos más pequeño. Según los autores este estudio mejora en precisión a los sistemas de detección existentes, y es portable pero el costo computacional que conlleva entrenar a la red neuronal es un inconveniente para la aplicación de este sistema en zonas de escasos recursos

TABLA LVI.
RESULTADOS DEL ARTÍCULO EP34

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep convolutional neural networks for mental load classification based on EEG data
		Autor	Zhicheng Jiao, Xinbo Gao, Ying Wang, Jie Li, Haojun Xu
		Referencia	[64]
		Año	2017
2	Aplicación	Clasificación de la carga mental: Emplean visión artificial para clasificar la carga mental a través de datos de la electroencefalografía (EEG), EEG es una medida utilizada para actividades cerebrales. Variando en la complejidad de las tareas cognitivas, la carga mental da como resultado diferentes grabaciones de EEG.	
		3 Modelos	Red neuronal convolucional (CNN).
		4 Funciones	Función de activación ReLu. Sofmax en la última capa.
		5 Conclusiones Relevantes	Las pruebas de EEG se han dividido en ventanas de 0,5 s para formar una instancia de 7 cuadros. Y la señal en cada cuadro se ha analizado con la transformada rápida de Fourier (FFT) también emplean el método de proyección polar denominado proyección equidistante azimutal (AEP) para realizar la transformación. Los autores generan dos arquitecturas CNN que difieren en la forma de los datos de entrada y otra arquitectura que fusiona los dos tipos de CNN anteriores. Para validar el modelo propuesto los autores lo comparan con otras técnicas del machine learning como regresión logística, arboles aleatorios, modelos DL como DBN, LSTM y redes neuronales completamente conectadas. El rendimiento de las CNNs fusionadas tiene el menor error registrado durante las pruebas. Los autores además concluyen que la información proporcionada a los modelos CNN se debe extraer de manera más eficiente.

TABLA LVII.
RESULTADOS DEL ARTÍCULO EP35

#	Descripción	Detalle	
1	Información bibliográfica	Título	Fusing texture, shape and deep model-learned information at decision level for automated classification of lung nodules on chest CT
		Autor	Xie Yutong, Zhang Jianpeng, Xia Yong, Michael Fulham, Zhang Yanninga
		Referencia	[65]
		Año	2017
2	Aplicación	Detección temprana de cáncer de pulmón: Utilizan visión artificial para clasificar nódulos pulmonares malignos y benignos.	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ReLu.	
5	Conclusiones Relevantes	<p>Emplean un descriptor de textura basado en la matriz de co-ocurrencia de nivel gris (GLCM), un descriptor de forma de Fourier para caracterizar la heterogeneidad de nódulos y una red neuronal convolucional (CNN) para aprender automáticamente la representación característica de nódulos en un corte-slice base. Entrena una red neuronal de propagación hacia atrás AdaBoosted (BPNN) usando cada tipo de característica y fusiona las decisiones tomadas por tres clasificadores para diferenciar los nódulos.</p> <p>Los autores recomiendan que para estudios futuros se considera descubrir métodos del DL más efectivos para caracterizar los nódulos pulmonares.</p>	

TABLA LVIII.
RESULTADOS DEL ARTÍCULO EP36

#	Descripción	Detalle	
1	Información bibliográfica	Título	A new take on measuring relative nutritional density: The feasibility of using a deep neural network to assess commercially-prepared pureed food concentrations
		Autor	Kaylen J. Pfisterer, Robert Amelard, Audrey G. Chung, Alexander Wong
		Referencia	[66]
		Año	2017
2	Aplicación	Detectar la densidad nutricional de los purés: DL y técnicas de visión artificial para clasificar diluciones hechas en los productos puros en donde extraen información óptica como color, textura, etc. Con ello predecir automáticamente la densidad de nutrientes de los productos comercialmente preparados. Evaluar cuantitativa y objetivamente la densidad nutricional de los purés.	
3	Modelos	Autoencoder apilados. (SAE)	
4	Funciones	Función de activación Sigmoide. Softmax en la última capa.	
5	Conclusiones Relevantes	<p>Los datos contienen imágenes de 13 sabores de puré en tres categorías de alimentos: fruta, carne y vegetales. Para preparar los datos de entrenamiento, cada imagen se normalizó en blanco, todas las imágenes fueron etiquetadas y deconstruidas en 100 subimágenes de 200 píxeles para cada muestra. La arquitectura de la red, consiste en dos capas de autoencoders apilados pre tratados y una capa final de activación de softmax.</p> <p>El sistema propuesto para la formación de imágenes y el análisis de la densidad de nutrientes de las máquinas es prometedor como una herramienta para asegurar la calidad de los nutrientes.</p>	

TABLA LIX.
RESULTADOS DEL ARTÍCULO EP37

#	Descripción	Detalle	
1	Información bibliográfica	Título	White blood cells identification system based on convolutional deep neural learning networks
		Autor	A.I. Shahin, Yanhui Guo, K.M. Amin, Amr A. Sharawi.
		Referencia	[67]
		Año	2017
2	Aplicación	Identificación de los glóbulos blancos: Emplear DL y técnicas de visión artificial para identificar las diferentes clases de glóbulos blancos para lograr la experticia de los patólogos en la identificación de los glóbulos blancos.	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ReLu. Softmax en la última capa	
5	Conclusiones Relevantes	Las imágenes son adaptadas a la capa de entrada de la CNN pre entrenada, la cual tiene una arquitectura de WBCsNet, consiste en una capa de entrada de imagen, tres capas convolucionales principales, dos capas de agrupamiento, cuatro unidades lineales rectificadas de activación (ReLU), dos capas totalmente conectadas y una capa de clasificación (Softmax). Los pesos aprendidos se adaptan a la forma, textura y características complejas de las imágenes utilizadas, luego se pasa a través de la red pre-capacitada con sus pesos previos y filtros aprendidos. Para clasificar nuevas imágenes se emplea la red con los pesos aprendidos y la máquina de vectores de soporte (SVM).	

TABLA LX.
RESULTADOS DEL ARTÍCULO EP38

#	Descripción	Detalle	
1	Información bibliográfica	Título	Multi-label deep learning for gene function annotation in cancer pathways
		Autor	Renchu Guan, Xu Wang, Mary Qu Yang, Yu Zhang, Fengfeng Zhou, Chen Yang, Yanchun Liang
		Referencia	[68]
		Año	2018
2	Aplicación	Etiquetar genes en vías de cáncer: Organizar información biológica, representarla y hacerla accesible. Explorar los efectos, si los hubiera, de múltiples funciones génicas en las vías del cáncer en KEGG.	
3	Modelos	Autoencoder de eliminación de ruido apilado.	
4	Funciones	Función de activación Sísmoide.	
5	Conclusiones Relevantes	1. Generar una matriz de características: extraen todas las identificaciones de genes de un archivo KGML dado, se busca todos los artículos que cubren funciones genéticas específicas, quedan artículos que poseen tanto la identificación de PubMed como la de PMC en un archivo de texto que contiene la lista de todos los archivos PDF descargables a través del servicio FTP de PMC. 2. Después de descargar todos los archivos pdf 18930, extraen el contenido de texto requerido. La matriz resultante representa genes con frecuencias de término. 3. Con la matriz de características se aplica el método basado en DL SdaML que consta de dos módulos: (a) autoencoders de eliminación de ruido apilados (SdA), capturan las dependencias entre dimensiones en alta distribución dimensional y (b) la retro propagación para aprendizaje de etiqueta múltiple (BP-MLL) se centra en encontrar la etiqueta de vía adecuada para cada gen. Este modelo aprovecha la reducción efectiva de la dimensión y la clasificación multicategoría a causa de los autoencoders de eliminación de ruido apilados.	

TABLA LXI.
RESULTADOS DEL ARTÍCULO EP39

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep learning-based nucleus classification in pancreas histological images
		Autor	Young Hwan Chang, Guillaume Thibault, Owen Madin, Vahid Azimi, Cole Meyers, Brett Johnson, Jason Link, Adam Margolin, Joe W. Gray
		Referencia	[28]
		Año	2017
2	Aplicación	Detectar cáncer de páncreas: Clasificar núcleos de células cancerígenas para calcular de manera precisa la pureza del tumor de páncreas.	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ReLu, emplean dropout para el entrenamiento.	
5	Conclusiones Relevantes	Construyen el dataset al recolectar imágenes celulares, toman la imagen de diapositiva completa, segmentada y emplean filtros gabor cada píxel se mejora mediante las características elegidas y luego, al agrupar los píxeles vecinos, que tienen características similares, se pueden segmentar núcleos individuales diferenciando entre diferentes tejidos y células o núcleos. Para la clasificación emplean una CNN para enfrentar el problema de falta de datos y sobreajuste utilizan esquemas de aumento de datos para el entrenamiento y generan imágenes mediante diferentes operaciones aleatorias, como rotaciones artificiales, turnos, volteos, etc., agregando ruido.	

TABLA LXII.
RESULTADOS DEL ARTÍCULO EP40

#	Descripción	Detalle	
1	Información bibliográfica	Título	VAMPnets for deep learning of molecular kinetics
		Autor	Andreas Mardt, Luca Pasquali, Hao Wu y Frank Noé
		Referencia	[69]
		Año	2018
2	Aplicación	Predecir el comportamiento de la cinética molecular en proteínas y fármacos: Predecir comportamiento mediante detección de patrones. Aprender la cinética molecular a partir de datos de simulación de una gama de sistemas modelo para equilibrar procesos biomoleculares, como la unión de proteínas y fármacos	
3	Modelos	Red neuronal profunda DNN de tipo MLP	
4	Funciones	Función de activación ReLu, se emplea dropout para el entrenamiento. Softmax en la capa de salida.	
5	Conclusiones Relevantes	VAMPnets arquitectura de red neuronal contienen dos lóbulos de red que transforman las configuraciones moleculares encontradas en un retraso de tiempo τ a lo largo de las trayectorias de simulación, combina las tareas de Featurization, reducción de dimensión, discretización y modelado cinético de grano grueso en un único marco de aprendizaje de extremo a extremo. La arquitectura de la red está definida por dos parámetros: la profundidad d y la cantidad de nodos de salida n-out aplicando dropout en las primeras dos capas y Softmax en la capa de salida. Todas las capas ocultas emplean unidades lineales rectificadas (ReLU).	

TABLA LXIII.
RESULTADOS DEL ARTÍCULO EP41

#	Descripción	Detalle	
1	Información bibliográfica	Título	Automatic knee osteoarthritis diagnosis from plain radiographs: A deep learning-based approach
		Autor	Aleksei Tiulpin, Jérôme Thevenot, Esa Rahtu, Petri Lehenkari y Simo Saarakkala
		Referencia	[70]
		Año	2018
2	Aplicación	Detectar la gravedad de osteoartritis de rodilla: Emplean técnicas de visión artificial para extraer características y clasificar datos para calificar automáticamente la gravedad de OA (osteoartritis de rodilla)	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ReLu, emplean dropout para el entrenamiento.	
5	Conclusiones Relevantes	Los datos se extraen de dos conjuntos de datos públicos: MOST y OAI. La arquitectura de la red se basa en la arquitectura CNN, consta de dos ramas, cada una de ellas con convolución, agrupamiento máximo, unidad lineal rectificada (ReLU), normalización de lotes, capas globales de agrupación y concatenación. Para los autores, el modelo propuesto posee la capacidad de evaluar sistemáticamente el estado de la rodilla del paciente y controlarlo para detectar otras afecciones	

TABLA LXIV.
RESULTADOS DEL ARTÍCULO EP42

#	Descripción	Detalle	
1	Información bibliográfica	Título	A deep learning-based strategy for identifying and associating mitotic activity with gene expression derived risk categories in estrogen receptor positive breast cancers
		Autor	David Romo-Bucheli, Andrew Janowczyk, Hannah Gilmore, Eduardo Romero, Anant Madabhushi
		Referencia	[71]
		Año	2017
2	Aplicación	Detectar cáncer de mama: Emplea DL y visión artificial para clasificar si un núcleo es mitótico o no, distinguiendo entre los tumores de mama más y menos agresivos, para la identificación precisa de los casos de cáncer que se beneficiarán de quimioterapia adyuvante adicional frente a aquellos que indican únicamente terapia hormonal.	
3	Modelos	Red neuronal convolucional (CNN).	
4	Funciones	Función de activación ReLu.	
5	Conclusiones Relevantes	Para obtener datos, emplean imágenes teñidas con Hematoxilina y Eosina. En la imagen resultante resalta la información de cromatina y un umbral adaptativo, la imagen de relación azul se convierte a mapa binario utilizando el histograma de nivel de gris. Se aplica umbral óptimo para maximizar la separabilidad de dos clústeres de píxeles resultantes. Los componentes conectados finalmente se seleccionan como núcleos candidatos. El centroide de cada núcleo candidato corresponde al centro del parche de color RGB asociado. Se emplean un modelo CNN, el objetivo determinar si es o no un núcleo mitótico. La arquitectura CNN, tiene una capa de normalización de lotes, función de activación (ReLU) y un operador de grupo máximo. Con el modelo entrenado al ingresar las nuevas imágenes el modelo extrae las características y se emplea una SVM (máquina de soporte vectorial) para evaluar si es o no un núcleo mitótico. Para los autores, la información mitótica extraída mediante DL alcanza un nivel de precisión suficiente para tratar la variabilidad del tejido del paciente y preservar suficiente información pronóstico para clasificarlos	

TABLA LXV.
RESULTADOS DEL ARTÍCULO EP43

#	Descripción	Detalle
1	Información bibliográfica	Título Multimodal deep learning for cervical dysplasia diagnosis
		Autor Tao Xu, Han Zhang, Xiaolei Huang, Shaoting Zhang y Dimitris N. Metaxas
		Referencia [72]
		Año 2016
2	Aplicación	Detectar displasia cervical: Predecir el diagnóstico de displasia cervical utilizando información multimodal recopilada durante una visita de selección del paciente. (emplean técnicas de visión artificial)
3	Modelos	Red neuronal convolucional (CNN).
4	Funciones	Función de activación ReLu. Softmax en la capa de salida.
5	Conclusiones Relevantes	<p>Los datos se extrajeron en cada visita del paciente el Cervigram y los resultados clínicos de las pruebas de Papanicolaou y del VPH. Siguen el esquema de aprendizaje de transferencia, toman el modelo que está pre entrenado en la tarea de clasificación de ImageNet y reemplazan la capa de salida por una nueva capa de enlace bidireccional softmax. Los parámetros se ajustan en el conjunto de datos Cervigram. Para detectar una región cervical de interés emplean la técnica ROI para cada Cervigram, cada región de ROI se alimenta a la red que genera el vector de características correspondiente desde su última capa completamente conectada.</p> <p>Los autores además de la imagen proporcionada por el Cervigram procesan datos como el test de Papanicolaou, el test de virus de papiloma humano (HPV), la apreciación médica y unifican la red de procesamiento de imágenes de CNN y la red de aprendizaje conjunto en un marco de extremo a extremo</p>

TABLA LXVI.
RESULTADOS DEL ARTÍCULO EP44

#	Descripción	Detalle
1	Información bibliográfica	Título Information theoretic-based interpretation of a deep neural network approach in diagnosing psychogenic non-epileptic seizures
		Autor Sara Gasparini, Maurizio Campolo, Cosimo Ieracitano, Nadia Mammone, Edoardo Ferlazzo, Chiara Sueri, Giovanbattista Gaspare Tripodi, Umberto Aguglia y Francesco Carlo Morabito
		Referencia [73]
		Año 2018
2	Aplicación	Detectar pacientes con epilepsia: Clasificar a pacientes afectados por ataques no epilépticos psicogénos (PNES) y sujetos de control sanos (CNT) basados en grabaciones de EEG.
3	Modelos	Autoencoder apilado (SAE).
4	Funciones	Función softmax.
5	Conclusiones Relevantes	<p>El diagnóstico de PNES se basó en los siguientes criterios: (a) eventos clínicos típicos registrados por video-EEG, provocados por maniobras de sugestión; (b) EEG que no muestra actividad ictal concomitante, ni ralentización posictal.</p> <p>El modelo en la primera etapa de representación consiste en transformar los vectores de tiempo de EEG en un mapa bidimensional de tiempo-frecuencia. A continuación, se extraen algunas características de "ingeniería" de los mapas. Tales características forman la entrada a dos etapas del autoencoder apilado (SAE) con aprendizaje no supervisado, y para la clasificación se entrena una red con retro propagación con linealidad softmax.</p> <p>Este enfoque propuesto, puede ser útil para la identificación temprana de personas que sufren de ataques no epilépticos psicogénos.</p> <p>Los sistemas basados en DL son capaces de extraer características de mayor nivel directamente de los datos disponibles, de tal forma que también se reduce el ruido y se rechaza la información no relevante.</p> <p>Limitantes: (a) Es difícil explicar el comportamiento y logros de DL. (b) las aplicaciones clínicas basadas en EEG, tienen datos limitados.</p>

Área de Conocimiento: FINANZAS

TABLA LXVII.
RESULTADOS DEL ARTÍCULO EP45

#	Descripción	Detalle	
1	Información bibliográfica	Título	Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations
		Autor	Gunduz H, Yaslan Y, Cataltepe Z
		Referencia	[74]
		Año	2017
2	Aplicación	Análisis del mercado de valores: Predicción de precios en el mercado bursátil.	
3	Modelos	Red Neural Convolutacional (CNN).	
4	Funciones	Función de activación ReLu. Softmax en la capa de salida. Dropout para el entrenamiento de la red	
5	Conclusiones Relevantes	Arquitectura de Red Neural Convolutacional (CNN) con un conjunto de características que se extraen utilizando diferentes indicadores, precios e información temporal. Las correlaciones entre las instancias de las características se utilizan para ordenar las características antes de que se presenten como entradas a la CNN. Los clasificadores se entrenaron y su rendimiento se midió en términos de la métrica F-Measure promediada (MA). Las etiquetas de clase se asignaron utilizando cada precio de cierre de cada acción y los movimientos en los precios de las acciones se indicaron mediante estas etiquetas. Se predijeron los movimientos por hora de 100 acciones en Borsa Estambul. El resultado experimental se obtiene en 4 años de entrenamiento y 1 año de prueba.	

TABLA LXVIII.
RESULTADOS DEL ARTÍCULO EP46

#	Descripción	Detalle	
1	Información bibliográfica	Título	Bank distress in the news: Describing events through deep learning
		Autor	Rönnqvist S, Sarlin P
		Referencia	[75]
		Año	2017
2	Aplicación	Detectar riesgo financiero: Emplean técnicas de PLN para detectar discusiones relevantes en el texto y extraer descripciones del lenguaje natural de los eventos, aplicado al estudio del riesgo financiero basado en noticias. Aplica DL para predecir eventos a partir del texto, extraer características y clasificación	
3	Modelos	Redes Neuronales Profundas del tipo MLP	
4	Funciones	Función de activación no lineal (ReLU, sigmoide, tanh no se especifica cual emplearon los autores).	
5	Conclusiones Relevantes	El método aplicado por los autores en este estudio consiste en: 1. Etiquetado de texto por datos de evento, se basa en la conexión de dos tipos de datos, texto y datos de eventos, por entidades y cronología. Un conjunto de datos de eventos contiene información sobre las fechas y los nombres de las entidades involucradas, en relación con el tipo específico de evento que se modelará. 2. Modelado: La red neuronal profunda que se aplica en este estudio está formada en dos pasos: mediante el aprendizaje de vectores oracionales como pre entrenamiento, aprendizaje supervisado frente a la señal de evento. El método y el análisis ejemplifican cómo el texto puede ofrecer una fuente de información para análisis de riesgos financieros y sistémicos. Entre las conclusiones más relevantes de los autores resalto: los datos de texto ofrecen una ruta posible para eludir los problemas de acceso privilegiado, publicación rezagada. Se espera que el método sea aplicable para describir eventos más allá del dominio financiero. DL es útil para manejar las complejidades de nuevos tipos de datos, además ofrecen la flexibilidad necesaria al explorar nuevos campos de análisis.	

TABLA LXIX.
RESULTADOS DEL ARTÍCULO EP47

#	Descripción	Detalle	
1	Información bibliográfica	Título	Financial time series forecasting - A deep learning approach
		Autor	Dingli A, Fournier K
		Referencia	[76]
		Año	2017
2	Aplicación	Análisis del mercado de valores: Pronosticar la dirección del precio de acciones de mercado del próximo período con respecto al precio actual.	
3	Modelos	Red Neural Convolucional (CNN).	
4	Funciones	Función de activación ReLu. Softmax en la capa de salida.	
5	Conclusiones Relevantes	La metodología empleada por los autores consiste en: 1. Datos: los datos empleados fueron: precios históricos e indicadores técnicos, intercambios de divisas, índices mundiales, productos básicos 3. Equilibrio de conjuntos de datos, división de datos en entrenamiento y prueba. Datos de 2003 a 2013 (11 años) para entrenamiento y de 2014 a 2016 (3 años) para prueba. 4. Modelo de pronóstico se emplea redes neuronales convolucionales (CNN). Los resultados obtenidos con el método propuesto, no igualan ni superan los resultados obtenidos por las técnicas líderes de la industria, tales como Regresión logística y Máquinas de vectores de soporte.	

TABLA LXX.
RESULTADOS DEL ARTÍCULO EP48

#	Descripción	Detalle	
1	Información bibliográfica	Título	Stock prediction using deep learning
		Autor	Singh R, Srivastava S
		Referencia	[77]
		Año	2017
2	Aplicación	Análisis del mercado de valores: Predecir el stock en el mercado de valores (valor de acciones de mercado).	
3	Modelos	Redes Neuronales Profundas del tipo MLP	
4	Funciones	Función de activación hyperbolic tangent	
5	Conclusiones Relevantes	En este estudio se propone un modelo DL hecho con una red neuronal profunda de propagación directa con cuatro capas ocultas. El método empleado por los autores consiste en la obtención de datos (dataset multimedia de Google NASDAQ), se emplea PCA para reducir la dimensionalidad de los datos, entran la DNN. Se compara el desempeño del modelo propuesto con una red neuronal recurrente (RNN) y la red neuronal función de base radial (RBFNN), los resultados experimentales confirman que el modelo propuesto (DNN) proporciona un método prometedor para la predicción de tendencia de stock. Los autores comentan que para estudios futuros se pudiera investigar la efectividad de la orientación profunda en la gestión de cartera y las estrategias de negociación.	

TABLA LXXI.
RESULTADOS DEL ARTÍCULO EP49

#	Descripción	Detalle	
1	Información bibliográfica	Título	A deep learning framework for financial time series using stacked autoencoders and long-short term memory
		Autor	Bao W, Yue J, Rao Y
		Referencia	[78]
		Año	2017
2	Aplicación	Análisis del mercado de valores: Predicciones en el mercado financiero. Prever el precio de las acciones.	
3	Modelos	autoencoders apilados (SAE) y red de memoria a largo plazo (LSTM)	
4	Funciones	Función de activación Sigmóide.	
5	Conclusiones Relevantes	<p>El marco DL comprende tres etapas. En primer lugar, transformaciones wavelet (WT), descompone la serie temporal del precio de las acciones para eliminar el ruido. Segundo, Se aplica SAE para generar funciones de alto nivel en la predicción del precio de las acciones. En tercer lugar, las características de eliminación de ruido de alto nivel se introducen en la LSTM para prever el precio de cierre del día siguiente.</p> <p>Se seleccionaron seis índices de mercado y sus futuros índices correspondientes para examinar el rendimiento del modelo propuesto. Para los autores el modelo propuesto tiene un rendimiento predictivo satisfactorio, pero se optimizaría con un esquema de selección de hiperparámetros más avanzado. Este trabajo es el primer intento de introducir el método de SAE para extraer características diarias invariantes profundas de series de tiempo financieras.</p>	

TABLA LXXII.
RESULTADOS DEL ARTÍCULO EP50

#	Descripción	Detalle	
1	Información bibliográfica	Título	A deep learning ensemble approach for crude oil price forecasting
		Autor	Iti Chaturvedia, Erik Cambria, Roy E. Welsch, Francisco Herrera
		Referencia	[79]
		Año	2017
2	Aplicación	Pronóstico del precio del petróleo.	
3	Modelos	Autoencoder apilado de eliminación de ruido (SDAE).	
4	Funciones	Función de activación Sigmóide.	
5	Conclusiones Relevantes	<p>Se generan múltiples conjuntos de entrenamiento k, réplicas de las muestras de entrenamiento, mediante bootstrapping.</p> <p>Se entrena para cada conjunto k un modelo SDAE. Para pruebas se ingresa el conjunto de pruebas por cada SDAE entrenado, se toma el valor medio de las predicciones k como resultado final.</p> <p>El modelo propuesto fue denominado Enfoque de conjunto DL- bagging (SDAE-B), combina SDAE y el bagging método de conjunto.</p> <p>El enfoque propuesto se puede utilizar como una herramienta de pronóstico prometedora para el precio del petróleo crudo.</p> <p>Es bien sabido que factores irregulares como el clima extremo, los riesgos políticos y el factor psicológico también tienen un gran impacto en la volatilidad del precio del petróleo, sin embargo, es bastante difícil cuantificar el efecto de ellos. Para trabajos futuros se puede generar una mejor precisión predictiva cuantificando estos factores y utilizar la información de ellos.</p>	

TABLA LXXIII.
RESULTADOS DEL ARTÍCULO EP51

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies
		Autor	Eunsuk Chong, Chulwoo Han, Frank C. Park
		Referencia	[80]
		Año	2017
2	Aplicación	Análisis del mercado de valores: Reconocer patrones, extraer características para la predicción y análisis de los mercados bursátiles.	
3	Modelos	Red neuronal profunda DNN tipo perceptrón multicapa (MLP).	
4	Funciones	Función de activación Sigmoide para la capa de salida, y ReLu en las capas ocultas.	
5	Conclusiones Relevantes	<p>El objetivo de este estudio es construir la función de predicción empleando una DNN, el rendimiento se compara con un modelo autorregresivo univariado (AR) y con una red neuronal artificial (ANN).</p> <p>La DNN utiliza un modelo de red de tres capas para la función de activación emplean ReLU (unidad lineal definida). Entrenamiento: datos de entrada: conjuntos de funciones, RawData, PCA380, AE400 y RBM400, el 20% del conjunto de entrenamiento se usa como un conjunto de validación para evitar el sobreajuste. Los DNN funcionan mejor que un modelo autorregresivo lineal en el conjunto de entrenamiento, pero la ventaja desaparece principalmente en el conjunto de prueba. El modelo de predicción se aplica al análisis de estructura de mercado basado en covarianza y este modelo mejora la estimación de covarianza de manera efectiva. Estos resultados pueden ayudar a mejorar el rendimiento de existencias y la predicción de covarianza.</p>	

TABLA LXXIV.
RESULTADOS DEL ARTÍCULO EP52

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500
		Autor	Christopher Krauss, Xuan Anh Do, Nicolas Huck
		Referencia	[81]
		Año	2016
2	Aplicación	Análisis del mercado de valores: Arbitraje estadístico para obtener los mismos resultados con técnicas de machine learning y las técnicas convencionales.	
3	Modelos	Red neuronal profunda tipo MLP	
4	Funciones	Función de activación maxout y softmax en la capa de salida	
5	Conclusiones Relevantes	<p>Consideran que existe una brecha entre las finanzas académicas que explican las anomalías del mercado y la industria financiera, que son propensos a implementar métodos de caja negra debido a la rentabilidad. Esto lo apoyan con The Journal of Finance, donde al realizar búsquedas de implementación de redes neuronales los resultados son limitados, pero para reducir esta brecha ellos proponen una técnica de arbitraje estadístico de manera que se pueda conseguir los mismos resultados con técnicas de machine learning y las técnicas convencionales, los modelos comparados son una DNN, un árbol con aumento de gradiente (GBT) y bosques aleatorios (RAF) concluyeron que los GBT superan en rendimiento a las DNN pero estas últimas tienen ventaja en la selección de parámetros. Evalúan el desempeño de la estrategia en términos de distribución del retorno, valor en riesgo, características de riesgo-rendimiento y exposición a fuentes comunes de riesgo sistemático. En segundo lugar, evalúan los rendimientos a lo largo del tiempo. En tercer lugar, evalúan las variables, dividen la cartera por sectores y realizan un análisis de sensibilidad para mostrar la solidez de la selección de hiperparámetros. Dividen los datos completos en períodos de estudio, que consisten en sesiones de entrenamiento y negociación. Para cada período de estudio, generan el espacio de funciones necesario para hacer predicciones.</p>	

TABLA LXXV.
RESULTADOS DEL ARTÍCULO EP53

#	Descripción	Detalle	
1	Información bibliográfica	Título	Big Data: Deep learning for financial sentiment analysis
		Autor	Sahar Sohangir1, Dingding Wang, Anna Pomeranets and Taghi M. Khoshgoftaar
		Referencia	[82]
		Año	2018
2	Aplicación	Predecir comportamiento del inversor: Mejorar el rendimiento del análisis de sentimiento para prever el comportamiento del inversor.	
3	Modelos	Red de memoria a largo plazo (LSTM), Red neuronal convolucional (CNN)	
4	Funciones	Función de activación ReLU	
5	Conclusiones Relevantes	<p>En el conjunto de datos de StockTwits: mensajes publicados desde (01-06 del 2015). Examinan las publicaciones para ver si existe alguna relación entre el precio de las acciones futuras y el sentimiento de los usuarios. En el filtrado de características se emplean técnicas como Chi-cuadrado, ANOVA e información mutua. Para la ganancia de información, emplean selección de funciones de información mutua. El modelo se entrena varios modelos para comparar sus resultados primero l modelo doc2vec para ver si puede aumentar la precisión de la predicción del sentimiento para los escritores del mercado de valores y los modelos LSTM, CNN. En este trabajo intentan ver si los modelos del DL podrían mejorar la precisión del análisis de opinión de los mensajes de StockTwits.</p> <ul style="list-style-type: none"> - Doc2vec- no sirve para predecir sentimientos - LSTM en comparación con doc2vec aumentó la precisión, pero está por debajo de los requisitos - CNN supera la regresión logística. 	

TABLA LXXVI.
RESULTADOS DEL ARTÍCULO EP54

#	Descripción	Detalle	
1	Información bibliográfica	Título	Using deep Q-learning to understand the tax evasion behavior of risk-averse firms
		Autor	Nikolaos D. Goumarias, Dimitrios Hristu-Varsakelis, Yannis M. Assael
		Referencia	[83]
		Año	2018
2	Aplicación	Detección de evasión de impuestos: Describir y predecir el comportamiento de los evasores de impuestos en contribuyentes adversos al riesgo	
3	Modelos	Red de Q-learning profunda (Deep Q-learning DQN), MLP	
4	Funciones	Función de activación ReLU	
5	Conclusiones Relevantes	<p>La red es un perceptrón multicapa de 4 capas (MLP) cuya entrada es el estado actual. Las primeras tres capas constan de 256 neuronas, seguidas de dos capas lineales paralelas de 101 y 2 neuronas, para calcular Q, respectivamente. La red hace uso de la función de transformación de la función de activación ReLU entre las capas. Finalmente, la DQN produce dos elementos de acción. Para mejorar la escalabilidad luego de la experimentación numérica, optan por utilizar Q-learning independiente para aprender dos Q-funciones diferentes. La red neuronal fue entrenada para "almacenar" los ingresos óptimos a largo plazo de la empresa, dado el estado inicial y la decisión. DQN se utilizó para "aprender" de manera eficiente las decisiones óptimas de la empresa a través de simulaciones de la evolución del estado de la empresa. Demostraron que pueden calcular la política óptima de la empresa y los ingresos tributarios correspondientes para el gobierno en el modelo "completo" que incluye tanto la aversión al riesgo, como la amnistía fiscal.</p>	

TABLA LXXVII.

RESULTADOS DEL ARTÍCULO EP55

#	Descripción	Detalle	
1	Información bibliográfica	Título	Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud
		Autor	Yibo Wang, Wei Xu
		Referencia	[84]
		Año	2017
2	Aplicación	Detección de fraudes de seguros: Identificar correctamente los factores de riesgo y reducir las pérdidas causadas por los reclamos fraudulentos.	
3	Modelos	Red neuronal profunda tipo MLP	
4	Funciones	Función de activación ReLU y dropout en el entrenamiento.	
5	Conclusiones Relevantes	Este estudio explota el conocimiento de los expertos para evitar el sobreajuste en los métodos de minería de datos sobre los datos existentes. Emplea métodos de minería de datos para evitar el sesgo; utilizan un algoritmo conocido como latent dirichlet allocation (LDA) y el modelo de red neuronal profunda compuesta por 7 capas ocultas cada una con la función de activación ReLU y cuya entrada son las características categóricas, numéricas y tópicas para el entrenamiento. Después del proceso iterativo de optimización de hiperparámetros, el modelo DNN genera resultados de detección y determina si un reclamo es fraudulento. La combinación de LDA y DNN permite que el modelo tenga un mejor rendimiento cuando se caracterizan comportamientos fraudulentos.	

TABLA LXXVIII.

RESULTADOS DEL ARTÍCULO EP56

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep learning-based corporate performance prediction model considering technical capability
		Autor	Joonhyuck Lee, Dongsik Jang y Sangsung Park
		Referencia	[85]
		Año	2017
2	Aplicación	Predicción del rendimiento corporativo: Utilizando los indicadores financieros y técnicos de una empresa como predictores.	
3	Modelos	Máquinas restringidas de Boltzmann (RBM), Redes Creencia Profunda (DBN)	
4	Funciones	No especifica	
5	Conclusiones Relevantes	Emplean máquinas restringidas de Boltzmann (RBM) como el componente principal de una deep belief network (DBN). Después de la capacitación previa con el RBM, el modelo propuesto se ajusta con un algoritmo de retropropagación. Pre-entrenamiento: Se emplea una RBM como entradas dos datasets (datos recientes y datos antiguos). Entrenamiento: la DBN emplea los parámetros pre-entrenados por la RBM para inicializar los valores de FNN (retroalimentación hacia adelante), para refinar el entrenamiento se emplean algoritmos de retro propagación.	

TABLA LXXIX.
RESULTADOS DEL ARTÍCULO EP57

#	Descripción	Detalle	
1	Información bibliográfica	Título	Forecast in the Pharmaceutical Area – Statistic Models vs Deep Learning
		Autor	Raquel Ferreira, Martinho Braga y Víctor Alves
		Referencia	[26]
		Año	2018
2	Aplicación	Predicción de ventas.	
3	Modelos	Red de memoria a largo plazo (LSTM)	
4	Funciones	Función de activación ReLU y Dropout para el entrenamiento.	
5	Conclusiones Relevantes	<p>La arquitectura del modelo se conforma de: la capa de entrada en la librería Keras no es necesario definir la capa de entrada porque se crea automáticamente cuando se definen las capas intermedias. Las siguientes capas son dos pares de capas de LSTM y Dropout.</p> <p>Y a la salida dos capas de Multi Perceptrón (capas densas) con una capa de Dropout en el centro. Las capas de LSTM aparecen en primer lugar ya que son los que tienen capacidad de memoria (tanto antigua como reciente).</p>	

TABLA LXXX.
RESULTADOS DEL ARTÍCULO EP58

#	Descripción	Detalle	
1	Información bibliográfica	Título	Application of Long Short-Term Memory Neural Network to Sales Forecasting in Retail—A Case Study
		Autor	Quan Yu, Kesheng Wang, Jan Ola Strandhagen y Yi Wang
		Referencia	[29]
		Año	2018
2	Aplicación	Predicción de ventas.	
3	Modelos	Red de memoria a largo plazo (LSTM)	
4	Funciones	Función de activación Sigmoide y tanh	
5	Conclusiones Relevantes	<p>Los datos de ventas son de 66 productos en 45 semanas en 2015. El pronóstico se realiza a nivel semanal, las ventas de cuatro semanas consecutivas se utilizan para pronosticar la venta de la quinta semana. La red LSTM está configurada con cuatro capas. La data se divide en entrenamiento y prueba, los datos de las primeras 30 semanas se toman como datos de entrenamiento, mientras que los últimos se utilizan como datos de prueba.</p>	

TABLA LXXXI.
RESULTADOS DEL ARTÍCULO EP59

#	Descripción	Detalle	
1	Información bibliográfica	Título	A Deep Learning Approach for the Prediction of Retail Store Sales
		Autor	Yuta Kaneko, Katsutoshi Yada
		Referencia	[86]
		Año	2016
2	Aplicación	Predicción de ventas.	
3	Modelos	Red neuronal profunda del tipo MLP	
4	Funciones	Función de activación ReLU, Dropout para el entrenamiento.	
5	Conclusiones Relevantes	<p>El modelo consiste en una red multiperceptron de tres capas con 100 neuronas en cada capa y con una función de activación ReLU y aplicando el algoritmo de Dropout para evitar el sobreajuste.</p> <p>Al poner en práctica el enfoque del DL, es necesario ajustar los parámetros mediante la realización de experimentos de prueba y error. De acuerdo con las propiedades de los datos para construir un modelo preciso.</p>	

TABLA LXXXII.
RESULTADOS DEL ARTÍCULO EP60

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep Learning for stock prediction using numerical and textual information
		Autor	Ryo Akita, Akira Yoshihara, Takashi Matsubara, Kuniaki Uehara
		Referencia	[87]
		Año	2016
2	Aplicación	Análisis del mercado de valores: Predecir el valor de acciones en el mercado de valores.	
3	Modelos	Red de memoria a largo plazo (LSTM)	
4	Funciones	Función de activación Sigmóide.	
5	Conclusiones Relevantes	Este enfoque predice los precios de cierre de acciones de 10 empresas por regresión a partir de información textual (empleando algoritmos de vectores oracionales) y numérica utilizando LSTM para memorizar los pasos de tiempo anteriores. La data contempla múltiples empresas para aprender las correlaciones entre ellas. Por ejemplo, un evento como "Nissan recuerda..." podría hacer que el precio de las acciones de Nissan disminuya, mientras que el precio de las acciones como Toyota aumente al mismo tiempo. Los autores deciden que el número de empresas de predicción sea de 10 debido a la limitación de tiempo de cálculo. El vector de entrada a la red LSTM se obtiene de la combinación del grupo de artículos y de los precios de acciones. Los dos vectores pueden no estar concatenados directamente porque hay una gran diferencia entre el número de sus dimensiones, para resolver este problema, se escala el tamaño de estos vectores ampliando o reduciendo las dimensiones de los vectores a la mitad de las dimensiones del vector de entrada. Utilizan una red neuronal como el método de escalado.	

TABLA LXXXIII.
RESULTADOS DEL ARTÍCULO EP61

#	Descripción	Detalle	
1	Información bibliográfica	Título	A Deep Learning Algorithm to Forecast Sales of Pharmaceutical Products
		Autor	Oscar Chang, Ivan Naranjo, Christian Guerron, Dennys Criollo, Jefferson Guerron, Galo Mosquera
		Referencia	[88]
		Año	2017
2	Aplicación	Predicción de ventas.	
3	Modelos	Autoencoder apilado (SAE), Perceptrón multicapa (MLP).	
4	Funciones	Función de activación ReLU	
5	Conclusiones Relevantes	Este modelo se compone de dos etapas autoencoder y precursor. El autoencoder que tiene 19 neuronas de entrada, 11 ocultas y 19 de salida, para el entrenamiento se establece una ventana que se moviliza por todos los datos un n número de veces para establecer los parámetros del autoencoder, cuyo objetivo es reproducir las entradas lo más exacto posible en la salida. El precursor es una red neural de tres capas cuyas entradas son las abstracciones generadas por el autoencoder entrenado. La salida está entrenada para predecir, el valor de venta de la semana siguiente a la posición de la ventana. El precursor nunca ve los patrones de datos reales capturados por la ventana, sino sólo la abstracción resultante generada por el Autoencoder, por lo que se requieren abstracciones de buena calidad.	

TABLA LXXXIV.
RESULTADOS DEL ARTÍCULO EP62

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep Learning for Mortgage Risk
		Autor	Justin A. Sirignano, Apaar Sadhwani, Kay Giesecke
		Referencia	[89]
		Año	2018
2	Aplicación	Detección de riesgo hipotecario.	
3	Modelos	Red neuronal profunda tipo MLP	
4	Funciones	Función de activación Sísmoide.	
5	Conclusiones Relevantes	El modelo aplicado es una arquitectura tipo perceptrón multicapa. Formulan como datos el tiempo discreto para los períodos por meses, la enumeración de los posibles estados de la hipoteca (actual, 30 días de morosidad, etc.), el estado de la n-ésima hipoteca en el momento t después de la originación. Una hipoteca hará la transición entre los distintos estados a lo largo de su historia. de por vida	

TABLA LXXXV.
RESULTADOS DEL ARTÍCULO EP63

#	Descripción	Detalle	
1	Información bibliográfica	Título	Deep Learning for Event-Driven Stock Prediction
		Autor	Xiao Ding, Yue Zhang, Ting Liu, Junwen Duan
		Referencia	[90]
		Año	2015
2	Aplicación	Análisis del mercado de valores: Predecir el valor de acciones en el mercado de valores.	
3	Modelos	Red neuronal convolucional (CNN)	
4	Funciones	Función de activación Sísmoide y tanh	
5	Conclusiones Relevantes	<p>Los autores modelan eventos a largo plazo (eventos del mes pasado), eventos a medio plazo (eventos de la semana pasada) y eventos a corto plazo (eventos en el último día) del cambio de precio de la acción. El modelo de predicción aprende el efecto de estos tres diferentes períodos de tiempo en las cotizaciones bursátiles basados en el en el marco de una CNN cuya entrada es una secuencia de incrustaciones de eventos, donde los eventos se organizan en orden cronológico, para las noticias a largo plazo (izquierda) y a medio plazo, la operación de convolución estrecha se utiliza para combinar eventos vecinos. Puede verse como una extracción de características basada en una ventana corrediza, que puede capturar información local a través de combinaciones de vectores en una ventana.</p> <p>Utilizan una capa de pooling máxima encima de la capa convolucional, que fuerza a la red para conservar sólo las características locales más útiles producidas por la capa convolucional. La operación de convolución se aplica al evento a largo plazo y a medio plazo, porque la unidad de tiempo es un día.</p>	

TABLA LXXXVI.
RESULTADOS DEL ARTÍCULO EP64

#	Descripción	Detalle	
1	Información bibliográfica	Título	Decision support from financial disclosures with deep neural networks and transfer learning
		Autor	Mathias Kraus, Stefan Feuerriegel
		Referencia	[91]
		Año	2017
2	Aplicación	Análisis del mercado de valores: Predecir el valor de acciones en el mercado de valores.	
3	Modelos	Red de memoria a largo plazo (LSTM)	
4	Funciones	Función de activación Síntomaide.	
5	Conclusiones Relevantes	La metodología para predecir los movimientos de las cotizaciones bursátiles sobre la base de la información financiera, comparan el aprendizaje automático ingenuo usando bolsas de palabras contra el modelo DL LSTM aplicados a la clasificación de la dirección del movimiento del precio de la acción (al alza o a la baja) a una regresión, donde predicen la magnitud del cambio. En ambos casos, estudian las variaciones de precios en términos de rendimiento nominal como anormal. Este último corrige para confundir los movimientos del mercado y asíla efecto del comunicado de prensa en sí.	

Anexo 2: Tabla integral de las preguntas de investigación

TABLA LXXXVII. RESULTADOS INTEGRADOS DE PI_2 Y PI_3

Área	Aplicación		Modelos	Resultado de búsqueda
Ingeniería	Visión Artificial	Reconocimiento de objetos	MLP	EP01, EP16
			CNN	EP02, EP03, EP07, EP08, EP09, EP10, EP12, EP13, EP14, EP16
			AE	EP08, EP18
		Robótica	MLP	EP20
		Emociones o expresiones faciales	CNN	EP04, EP21
			AE	EP04
		Reconocimiento de personas	MLP	EP11
	Procesamiento del lenguaje Natural		CNN	EP11, EP15
			LSTM	EP11
			CNN	EP06
	Clasificación de servicios de nube		LSTM	EP06
			SAE	EP17
			Q-learning	EP17
	Seguridad de la información	Detección de ciberataques	MLP	EP19
		Detección de amenazas de seguridad	SAE	EP22
		Detección de Spam	DBN	EP05
	Clasificación de actividad humana		SAE	EP05
	Análisis de sentimientos		DBN	EP23
	Detectar calidad de video		DBN	EP24
	Detectar calidad de video		DBN	EP25
Industria	Detectar calidad de un producto (lechuga)		CNN	EP26
	Recomendación de hoteles		CNN	EP27
			LSTM	EP27
Investigación	Detectar ecuaciones		CNN	EP28
	Diagnóstico de rotor		DBN	EP29
	Detectar grietas		CNN	EP30
	Captura de CO2		DBN	EP32
	Visión Artificial		CNN	EP31

Salud	Diagnóstico médico	Detección de tuberculosis	CNN	EP33
		Detección de cáncer de pulmón	CNN	EP35
		Detectar cáncer de páncreas	CNN	EP39
		Detectar la gravedad de osteoartritis de rodilla	CNN	EP41
		Detectar cáncer de mama	CNN	EP42
		Detectar displasia cervical	CNN	EP43
		Detectar pacientes con epilepsia	SAE	EP44
		Clasificación de grabaciones de EEG	CNN	EP34
		Detectar la densidad nutricional de los purés	SAE	EP36
		Identificación de los glóbulos blancos	CNN	EP37
		Etiquetar genes en vías de cáncer	SAE	EP38
		Predecir el comportamiento de la cinética molecular en proteínas y fármacos	MLP	EP40
Finanzas	Análisis del mercado de valores		CNN	EP45, EP47, EP63
			MLP	EP48, EP51, EP52
			SAE	EP49
			LSTM	EP49, EP60, EP64
		Detectar riesgo financiero	MLP	EP46
		Pronóstico del precio del petróleo	SAE	EP50
		Predecir comportamiento del inversor	LSTM	EP53
			CNN	EP53
		Detección de evasores de impuestos	MLP	EP54
			Q-learning	EP54
		Detección de fraudes de seguros	MLP	EP55
		Predicción del rendimiento corporativo	RBM	EP56
			DBN	EP56
		Predicción de ventas	LSTM	EP57, EP58
			MLP	EP59, EP61
			SAE	EP61
		Detección de riesgo hipotecario	MLP	EP62

Anexo 3: Código completo del modelo desarrollado en Keras

```
6. #IMPORTAR PAQUETES
7. import numpy as np
8. import matplotlib.pyplot as plt
9. import pandas as pd
10. import math
11. from keras.models import Sequential
12. from keras.layers import Dense
13. from keras.layers import LSTM
14. from keras.layers import Dropout
15. from sklearn.preprocessing import MinMaxScaler
16. from sklearn.metrics import mean_squared_error
17. import os
18.
19. # Fijamos la semilla para tener reproducibilidad
20. np.random.seed(103596)
21.
22. #Llama la data sin transformar
23. data = pd.read_csv('dataVentas.csv')
24.
25. #Llama a la serie de tiempo convertida en estacionaria
26. #Esta operación se realizó en r con diff(log(ts(data)))
27. data = pd.read_csv('data_estacionaria.csv')
28.
29. #escala la serie en (0,1)
30. scaler = MinMaxScaler(feature_range=(0, 1))
31. #dataset = scaler.fit_transform(dataDif)
32. dataset = scaler.fit_transform(data)
33.
34. #Divide el dataset en entrenamiento y prueba (80-20):
35. train_size = int(len(dataset) * 0.8)
36. train, test = dataset[0:train_size,:],
37.     dataset[train_size:len(dataset),:]
38.
39.
40. # APLICAR LAG
41. # hay que convertir el vector de valores de la serie
42. # de tiempo en un problema de regresión para poder aplicar
43. # algoritmos de predicción
44. # Devuelve (X = variables retrasadas, Y= variable objetivo)
45. # loo_back = Determina el tamaño del lag o # de variables
46. # retrasadas
47. def create_dataset(dataset, look_back=1):
48.     dataX, dataY = [], []
49.     for i in range(len(dataset)-look_back-1):
50.         a = dataset[i:(i+look_back), 0]
51.         dataX.append(a)
52.         dataY.append(dataset[i + look_back, 0])
53.     return np.array(dataX), np.array(dataY)
54.
55. look_back = 10
56.
57. trainX, trainY = create_dataset(train, look_back)
58. testX, testY = create_dataset(test, look_back)
59.
60.
```

```

61. # Actualmente los datos están en la forma de
62. [ samples , features ]
63. # Reformular el input para que sea
64. [ samples , time steps , features ]
65. trainX = np.reshape(trainX,(trainX.shape[0],trainX.shape[1],1))
66.
67. testX = np.reshape(testX, (testX.shape[0],testX.shape[1],1))
68.
69. #MODELO
70. # Creación y entrenamiento de la red LSTM
71. # Keras calcula la capa de entrada automáticamente
72. #La salida es de un valor (predicción)
73. #Los parámetros de la red son:
74. #- Tamaño del lag (look_back)
75. #- Número de bloques
76. #- Profundidad
77. #- Dropout
78. # - Tamaño del batch (batch_size = 1)
79. # - Número de épocas (nb_epoch)
80. # - Función de activación por defecto tanh
81. #Arquitectura LSTM
82. batch_size = 1
83.
84. model = Sequential()
85.
86. #look_back=input_length, input_dim=1
87. model.add(LSTM(10,
88.                 batch_input_shape=(batch_size, look_back, 1),
89.                 stateful=False,
90.                 return_sequences=False,
91.                 activation = 'sigmoid'))
92. #model.add(TimeDistributed(LSTM(10,
93.                             batch_input_shape=(batch_size, look_back, 1),
94.                             stateful=False,
95.                             return_sequences=True)))
96. model.add(Dropout(0.1))
97. #model.add(LSTM(10, return_sequences=True, stateful=True))
98. #model.add(Dropout(0.1))
99. #model.add(LSTM(10, return_sequences=True))
100. #model.add(LSTM(10, return_sequences=True))
101. #model.add(LSTM(10, return_sequences=True))
102. #output Size=1 (1 timestep)
103. #model.add(Flatten())
104. model.add(Dense(1))
105.
106. #OPTIMIZACIÓN
107. model.compile(loss='mean_squared_error', optimizer='adam')
108.
109. #Captura el tiempo antes de entrenar el modelo
110. tiempo_inicial = time()
111.
112. #ENTRENAMIENTO
113. model.fit(trainX,
114.             trainY, nb_epoch=200,
115.             batch_size=1,
116.             verbose=1,
117.             shuffle=False)
118.
119. #Stateful=True, resetea el estado en cada época

```

```

120. #for i in range(100):
121. #    model.fit(trainX, trainY, nb_epoch=1,
122. #                batch_size=batch_size, verbose=2, shuffle=False)
123. #    model.reset_states()
124.
125. #Captura el tiempo terminado el entrenamiento
126. tiempo_final = time()
127.
128. #Calcula el tiempo de entrenamiento del modelo
129. tiempo_ejecucion = tiempo_final - tiempo_inicial
130.
131. #Devuelve la estructura de la red
132. model.summary()
133.
134. # TEST
135. # Hacer predicciones
136. trainPredict = model.predict(trainX, batch_size=batch_size)
137. model.reset_states()
138. testPredict = model.predict(testX, batch_size=batch_size)
139.
140. # Invertir predicciones a sus unidades originales
141. trainPredict = scaler.inverse_transform(trainPredict)
142. trainY = scaler.inverse_transform([trainY])
143. testPredict = scaler.inverse_transform(testPredict)
144. testY = scaler.inverse_transform([testY])
145.
146. # Calcular RMSE
147. trainScore = math.sqrt(mean_squared_error(trainY[0],
148.                           trainPredict[:,0]))
149.
150. testScore = math.sqrt(mean_squared_error(testY[0],
151.                           testPredict[:,0]))

```

Anexo 4: Carta de aceptación del artículo



Escuela Superior Politécnica de
Chimborazo
Escuela de Ingeniería en Sistemas



Oficio interno no. 318

Ph.D. Nikolay Aguirre
Rector de la Universidad Nacional de Loja

P R E S E N T E

Por medio de la presente, hago constar que el manuscrito técnico titulado: *Aplicaciones del Deep Learning en la Intermediación Financiera: Revisión Sistemática de Literatura*, presentado por los co-autores: Martha C. Suntaxi-Sarango, Pablo F. Ordoñez-Ordoñez y Manuel A. Pesantez-González, **ha sido aceptado** para ser presentado en la tercera edición del Simposio Iberoamericano en Programación Informática (SIIPRIN'2018), a celebrarse del lunes 26 de noviembre al viernes 30 de noviembre de 2018 en la Escuela Superior Politécnica de Chimborazo. Cabe señalar que todos los manuscritos enviados en esta edición del SIIPRIN'2018 fueron sometidos a un proceso de revisión por pares, por lo que las decisiones tomadas sobre las aceptaciones y rechazos de los manuscritos recibidos se basaron en las recomendaciones emitidas por los respectivos pares evaluadores.

Riobamba, Ecuador a 12 de noviembre de 2018

GRIISCF
GRUPO DE INVESTIGACIÓN EN INGENIERÍA DE SOFTWARE

Omar Salvador Gómez Gómez, Ph.D.
General Chair SIIPRIN'2018
Docente de la Facultad de Informática y Electrónica de la
Escuela Superior Politécnica de Chimborazo

c.c.p. archivo

Anexo 5: Artículo indexado en Knowledge E



KnE Engineering

SIIPRIN-CITEGC

Ibero-American Symposium on Computer Programming jointly held with
the International Congress on Technology Education and Knowledge Management
Volume 2018



Conference Paper

Applications of Deep Learning in Financial Intermediation: A Systematic Literature Review

Aplicaciones del Deep Learning en la Intermediación Financiera: Revisión Sistemática de Literatura

Martha C. Suntaxi-Sarango¹, Pablo F. Ordoñez-Ordoñez^{1,2}, and
Manuel A. Pesantez-González¹

¹Facultad Energía, Carrera de Ingeniería en Sistemas, Universidad Nacional de Loja, Ecuador

²ETSI Sistemas Informáticos, Universidad Politécnica de Madrid, Calle Alan Turing s/n 28031,
Madrid, España

Abstract

In finance, an infinite amount of data is generated daily, which is important for decision-making in the business world. Consequently, there is a need to create models that help to process and interpret this data. Deep learning has demonstrated important advances in the processing of large amounts of data, and for this reason, the objective of this systematic review of literature corresponds to the search for applications, deep learning model and techniques that were used to solve problems in the financial area. For this purpose, out of 346 articles found, 20 were selected that met the inclusion and exclusion criteria corresponding to the research questions. Among the most common applications, models, and techniques were: prediction in market actions, sales forecasting, detection of fraud risks and tax evasion; with respect to the models, convolutional neural networks CNN and recurrent neural networks RNN were among the most executed; the ReLu and Sigmoid techniques turned out to be the most used in these models.

Resumen

En las finanzas se generan infinidad de datos a diario, los cuales son importantes para la toma de decisiones en el mundo de los negocios, en consecuencia, surge aquella necesidad de crear modelos que ayuden a procesar e interpretar estos datos. Deep learning ha demostrado importantes avances en el procesamiento de grandes cantidades de datos, por tal razón, el objetivo de la presente revisión sistemática de literatura corresponde, a la búsqueda de las aplicaciones, modelo de aprendizaje profundo y técnicas que se emplearon para resolver problemas del área financiera. Para ello de 346 artículos encontrados se seleccionaron 20 que cumplen con los criterios de inclusión y exclusión correspondientes a las preguntas de investigación. Entre las aplicaciones, modelos y técnicas más comunes se encontraron: predicción en acciones del mercado, pronóstico de ventas, detección de riesgos de fraude y evasión

OPEN ACCESS

How to cite this article: Martha C. Suntaxi-Sarango, Pablo F. Ordoñez-Ordoñez, and Manuel A. Pesantez-González, (2018), "Applications of Deep Learning in Financial Intermediation: A Systematic Literature Review — Aplicaciones del Deep Learning en la Intermediación Financiera: Revisión Sistemática de Literatura" in Ibero-American Symposium on Computer Programming jointly held with the International Congress on Technology Education and Knowledge Management, KnE Engineering, pages 47-60. DOI 10.18502/keg.v3i9.3645

Page 47



de impuestos; con respecto a los modelos se hallaron entre los más ejecutados las redes neuronales convolucionales CNN y las redes neuronales recurrentes RNN; las técnicas ReLu y Sigmoid, resultaron ser las más empleadas en estos modelos.

Keywords: deep learning, finance, machine learning, Convolutional Neural Network CNN, Recurrent Neural Network RNN

Palabras clave: Aprendizaje Profundo, Finanzas, Aprendizaje Máquina, Redes Neuronales Convolucionales CNN, Redes Neuronales Recurrentes RNN

1. Introducción

El aprendizaje profundo (Deep Learning) DL contempla modelos computacionales que están compuestos de múltiples capas de procesamiento para mejorar el nivel de abstracción y el aprendizaje de grandes cantidades de datos [1]. Estos métodos han mejorado técnicas como el reconocimiento de voz, reconocimiento y detección de objetos visuales y otros dominios. Existen varias áreas de conocimiento de la ingeniería computacional que emplean DL como reconocimiento de dígitos, procesamiento del lenguaje, reconocimiento de voz y técnicas de visión artificial en imágenes y en vídeo. En el campo de estudio de la salud el DL esta extensamente explotado para reconocimiento de patrones en imágenes o datos que determinen la presencia de enfermedades o mutaciones que afectan a las personas [2–4]. En la investigación científica se emplea para solventar problemas en los que se involucran grandes cantidades de datos como procesos de captura de CO₂ [5], detectar ecuaciones o algoritmos para facilitar el desarrollo de ecuaciones [6, 7]. En el área financiera se utiliza en la predicción del comportamiento financiero, por lo tanto, lo que se buscó en esta SLR (revisión sistemática de literatura) fueron los modelos de aprendizaje profundo que se aplicaron a diferentes problemas del área financiera con el fin de identificar los conocimientos actuales en la ciencia. Para lograr este propósito se trazó las siguientes preguntas de investigación: *Q1: ¿Qué aplicaciones del DL existen en el área financiera?*, *Q2: ¿Qué modelos de DL se emplean para resolver el problema de investigación financiera?* y *Q3: ¿Qué técnicas del DL se emplean para resolver el problema de investigación financiera?*

La presente investigación está estructurada de la siguiente manera: La sección 2 ilustra la metodología empleada para obtener los resultados que se observan en la



sección 3. En la sección 4 se presenta la discusión de los hallazgos correspondiente a los 20 estudios primarios seleccionados. Finalmente, a partir de estos resultados en la sección 5 se establecen las respectivas conclusiones y trabajos futuros.

2. Metodología

Para cumplir con el objetivo de la revisión sistemática de literatura (SLR) se usó el método de Bárbara Kitchenham [8] que se desarrolla a continuación:

2.1. Revisión del desarrollo del protocolo

Siguiendo los lineamientos de Kitchenham [8], en la primera etapa se consideraron las siguientes fuentes de búsqueda, basados en la accesibilidad y admisión de consultas avanzadas: Elsevier, Scopus, IEEE Xplorer y GoogleScholar. El protocolo de revisión define los criterios de inclusión y exclusión, estrategia de búsqueda, evaluación de calidad y la extracción y síntesis de datos.

2.1.1. Criterios de Inclusión y Exclusión

Se estableció los criterios de inclusión con el propósito de extraer documentos que aborden la aplicación de DL en el área financiera. Resultando los siguientes requisitos de inclusión:

- Que el título indique alguna aplicación del DL a un caso de estudio de finanzas.
- Que el título, palabras claves señalen modelos del DL para el procesamiento de datos.
- Se considera artículos científicos (revistas, conferencias).
- Estudios en español e inglés.

Luego, los documentos que tienen estas características fueron excluidos:

- Artículos relacionados con finanzas que no aplican modelos DL.
- Estudios que no sean de carácter experimental.
- Estudios que no aporten o respondan a las Q's.

Además, para la selección de los estudios primarios se eliminaron los estudios duplicados.



2.1.2. Proceso de Búsqueda

El proceso de búsqueda de nuestra revisión de literatura comenzó con la selección de fuentes de búsqueda y preguntas de investigación. Esta selección juega un papel clave para mostrar la exhaustividad y la integridad de los artículos acumulados (véase Tabla 1). Los siguientes pasos se usaron para el proceso de búsqueda, los resultados se ilustran en la Figura 1:

1. Seleccionar las bibliotecas digitales (IEEE, Scopus, Google Scholar, ELSEVIER) que están enfocadas a la presente investigación.
2. Con las revistas y los artículos de conferencias filtrar por título, palabras clave y resumen.
3. Adquirir términos de búsqueda considerando las preguntas de investigación y basados en una revisión previa de artículos relacionados con la investigación.
4. Aplicar AND booleano para restringir la búsqueda.
5. Usar el Algoritmo 1, en donde se detalla la secuencia de pasos para aplicar los criterios de inclusión y exclusión.

Algoritmo 1: Selección de artículos para la revisión de literatura, empleando los criterios de inclusión y exclusión detallados en la revisión de protocolo de búsqueda.

```
Entrada: Preguntas de investigación
Entrada: Cadenas de búsqueda
Entrada: Fuentes de búsqueda
Salida: Estudios primarios
1:   ingresar a la fuente de búsqueda;
2:   ingresar cadena de búsqueda;
3:   mientras existe resultados
4:     Si cumple con los criterios de inclusión
5:     Si cumple con los criterios de exclusión
6:     Eliminar (artículo);
7:   Si no
8:     Guardar (artículo);
9:   Si no
10:  Eliminar (artículo);
11: fin mientras.
```

2.1.3. Evaluación de la calidad

Se revisó los criterios de evaluación de calidad descritos en el estudio realizado por Kitchenham [8]. La lista de verificación de la evaluación de la calidad empleada para los documentos seleccionados se muestra en la Tabla 2.



TABLA 1: Términos y fuentes de búsqueda, resultados aplicando el operador booleano AND.

#	Términos de búsqueda	Operador	Scopus	IEEE	Google Scholar	Elsevier
1	"Deep Learning" "finance"	AND	802	36	14800	496
2	"Deep Learning" "business analytics"	AND	77	17	920	56
3	"Deep Learning" "business analytics" "finance"	AND	14	2	30	21
4	"Deep Learning" "sales prediction"	AND	9	4	82	3
5	"Deep Learning" "sales prediction" "finance"	AND	2	2	28	2
6	"Deep Learning" "stock market prediction"	AND	128	27	523	30

TABLA 2: Lista de verificación de evaluación de calidad.

#	Pregunta	Respuesta
1	¿El documento tiene como objetivo identificar el tipo de aplicación de DL en las finanzas?	SI
2	¿La investigación revisa alguno de los artículos anteriores?	SI
3	¿Los hallazgos abordan las preguntas de investigación originales?	SI
4	¿Está el documento sesgado hacia un algoritmo, modelo o técnica de DL?	NO

2.1.4. Extracción y síntesis de datos

Se identificó la extracción y síntesis de datos a fin de registrar y recopilar con precisión los datos mediante el estudio de publicaciones seleccionadas. Se formuló la Tabla 3 para registrar con precisión los datos y proporcionar la solución a las preguntas. Se incluyó datos generales sobre el documento (título, nombre del autor, año de publicación). En la síntesis de datos cada pregunta se evaluó por separado.

3. Resultados

En esta sección se exploran los hallazgos de la SLR mediante un protocolo de revisión determinado. Los resultados se analizan mediante técnicas cualitativas y comparativas.

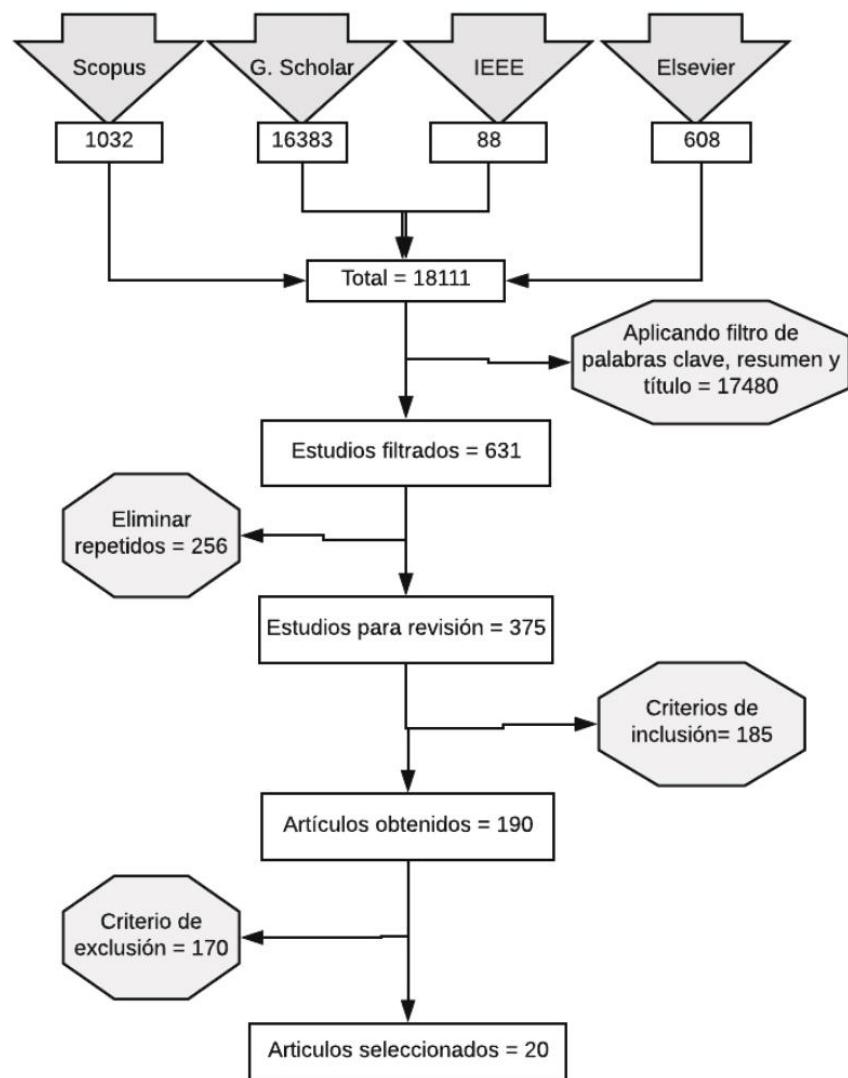


Figure 1: Proceso de búsqueda y selección de artículos, ingresando en las fuentes de búsqueda los términos de búsqueda con el operador AND se obtuvo un total 18111, aplicando filtros se reducen a 631, eliminando los estudios repetidos resultan en 375 artículos, luego de aplicar los criterios de inclusión y exclusión obtenemos 20 artículos.

TABLA 3: Extracción y síntesis de datos.

#	Descripción	Detalle
1	Información bibliográfica	Autor, título, año de publicación
Extracción de datos		
2	Descripción general	Objetivo principal del documento seleccionado
3	Resultados	Resultados adquiridos del artículo seleccionado
4	Suposiciones	Para validar el resultado
5	Conclusiones relevantes	Conclusiones del autor del documento seleccionado
Síntesis de datos		
6	Aplicación	Tipo de problema abordado en el documento seleccionado
7	Modelos	Modelos de DL empleados para procesar los datos
8	Técnicas	Técnicas de DL empleados resolver procesamiento de datos

3.1. Vista general de los estudios seleccionados

Se identificó 20 estudios de carácter financiero que aplican DL para el procesamiento de datos. Todos los estudios seleccionados son de investigación experimental. La Tabla 4 muestra el resumen de los trabajos seleccionados.

TABLA 4: Trabajos seleccionados en cada base de datos científica.

#	Base de datos científica	No. resultados	Resultado de búsqueda RB: Estudios Primarios (Código [Referencia])
1	Scopus	7	RB01[9], RB02[10], RB03[11], RB04[12], RB05[13] RB06[14], RB07[15]
2	IEEE	2	RB08[16], RB09[17]
3	Google Scholar	5	RB10[18], RB11[19], RB12[20], RB13[21], RB14[22]
4	Elsevier	6	RB15[23], RB16[24], RB17[25], RB18[26], RB19[27], RB20[28]

3.2. Q1: ¿Qué aplicaciones del DL existen en el área financiera?

Todos los artículos se analizaron para descubrir: ¿Cómo los autores aplican DL en el área financiera? Las aplicaciones para predicción de acciones, pronóstico de ventas, precios, rendimiento corporativo y otros riesgos financieros se abordan en Q1.

La predicción de acciones de mercado y pronóstico de ventas son las que tienen mayor atención de investigación en los últimos años, estas suman el 65% de los trabajos seleccionados, como se ilustra en la Tabla 5.

TABLA 5: Identificación de las aplicaciones en el área financiera Q1.

#	Aplicación	Porcentaje (%)	Resultado de búsqueda
1	Predecir el valor de acciones (mercado de valores)	45	RB15, RB17, RB18, RB19, RB03, RB04, RB11, RB09, RB14
2	Detectar fraudes y evasión	10	RB16, RB20
3	Comportamiento de la Banca	5	RB01
4	Predecir precio del petróleo	5	RB02
5	Pronóstico de ventas	20	RB08, RB05, RB07, RB10
6	Predecir precio del petróleo	5	RB06
7	Comportamiento del Inversor	5	RB12
8	Determinar riesgo hipotecario	5	RB13

3.3. Q2: ¿Qué modelos de DL se emplean para resolver el problema de investigación financiera?

En esta pregunta de investigación se enumeraron diferentes modelos orientados a la clasificación de datos y detección de patrones con el propósito de predecir eventos (véase Tabla 6). Los modelos CNN y LSTM fueron los más usados en las diferentes investigaciones. En los estudios revisados generalmente se emplean datos históricos o series temporales para la predicción de eventos, sin embargo, se encontró estudios que informaron la predicción de eventos a través de texto (RB18, RB20, RB01, RB12), otros estudios como RB25 y RB14 combinan datos numéricos con información en texto.

En las publicaciones RB11 y RB15 emplean PCA para reducción de la dimensionalidad, otros investigadores hacen uso de autoenconders (RB19, RB02, RB03, RB10) combinando varios modelos diferentes para la predicción. En RB16 se utilizó un algoritmo de aprendizaje por refuerzo para el problema de evasión de impuestos donde se manejan parámetros de las empresas y del gobierno con la finalidad de predecir cómo responderá la empresa frente a cambios en los parámetros de gobierno. La mayoría de los algoritmos analizados en esta investigación emplearon aprendizaje supervisado.

Los trabajos seleccionados para esta revisión comparan el modelo propuesto con otras técnicas empleadas en el mercado, con el objetivo de validar su estudio. El RB04 considera como datos de entrada precios históricos, indicadores técnicos, intercambio de divisas, productos básicos e índices mundiales, resalta que los resultados



obtenidos con el modelo CNN no igualan ni superan a las técnicas de regresión logística y máquinas de vectores de soporte que son los que se emplean frecuentemente en la industria.

Otro estudio a resaltar es el RB19 que reafirma la investigación al considerar que existe una brecha entre las finanzas académicas que explican las anomalías del mercado y la industria financiera, que son propensos a implementar métodos de caja negra debido a la rentabilidad. Esto lo apoyan con The Journal of Finance, donde al realizar búsquedas de implementación de redes neuronales los resultados son limitados, pero para reducir esta brecha ellos proponen una técnica de arbitraje estadístico de manera que se pueda conseguir los mismos resultados con técnicas de machine learning y las técnicas convencionales, los modelos comparados son una DNN, un árbol con aumento de gradiente (GBT) y bosques aleatorios (RAF) concluyeron que los GBT superan en rendimiento a las DNN pero estas últimas tienen ventaja en la selección de parámetros.

TABLA 6: Identificación de Modelos/algoritmos.

#	Modelos / algoritmos	Resultado de búsqueda
Predecir el valor de acciones, comportamiento del inversor y rendimiento corporativo		
1	Análisis de componentes principales (PCA)	RB15, RB11
2	Autoencoder (AE) autoencoder apilado	RB15, RB03
3	Máquina Boltzmann restringida (RBM)	RB15, RB06
4	Redes neuronales convolucionales (CNN)	RB17, RB04, RB12, RB14
5	Larga memoria a corto plazo (LSTM)	RB18, RB03, RB12, RB09
6	Red neuronal profunda (DNN)	RB19, RB11
7	Vectores oracionales	RB09
Detectar fraudes, evasión y riesgo hipotecario		
1	Deep Q -learning (DQN)	RB16
2	Red neuronal profunda (DNN)	RB20, RB13
3	Latent dirichlet allocation (LDA)	RB20
Comportamiento de la Banca		
1	Red neuronal profunda (DNN)	RB01
2	Vectores oracionales	RB01
Pronóstico de ventas y precio del petróleo		
1	Autoencoder (AE) apilado de eliminación de ruido (SDAE)	RB02, RB10
2	Red neuronal profunda (DNN)	RB08, RB10
3	Larga memoria a corto plazo (LSTM)	RB05, RB07

3.4. Q3: ¿Qué técnicas del DL se emplean para resolver el problema de investigación financiera?

Esta sección del trabajo presenta técnicas u hallazgos (véase Tabla 7) adicionales que los autores emplearon para ejecutar los modelos antes mencionados y que se consideró importantes conocer si se pretende replicar estos estudios.

Las funciones de activación tienen como objetivo definir las salidas de los nodos de la red neuronal, siendo ReLu y Sigmoid las más empleadas en los estudios revisados. Encontramos también que los RB17, RB04, RB01 emplean soft-max en la última capa, para la distribución de probabilidad de las posibles salidas y que RB05 usa dropout para optimizar la etapa de entrenamiento.

TABLA 7: Identificación de técnicas/funciones.

#	Técnicas / Funciones	Referencia
1	ReLU: unidad lineal rectificada (función de activación)	RB15, RB16, RB20, RB04, RB05, RB10, RB08, RB17, RB13
2	Sigmoid: función sigmoide (función de activación)	RB15, RB07, RB14, RB02, RB03, RB06, RB07, RB03
3	Maxout (función de activación)	RB19
4	Tanh: Tangente hiperbólica (función de activación)	RB11, RB14
5	Soft-max (última capa)	RB17, RB04, RB01
6	Dropout (entrenamiento)	RB05

4. Discusión

En esta SLR, se identificó los campos de aplicación y modelos de DL empleados en el área financiera. Basado en los resultados encontrados el DL es útil para la clasificación de datos y detección de patrones con el objetivo de predecir eventos. En finanzas las acciones empresariales son un fenómeno cuyas variables se caracterizan por su volatilidad (RB03), lo cual ha inspirado el 45% de las investigaciones examinadas en la ejecución de este trabajo. Entre otras aplicaciones se encontró: pronóstico de ventas, precios, rendimiento corporativo, detección de fraudes, evasión y riesgo hipotecario.

Los conjuntos de datos utilizados en los estudios primarios son paramétricos, históricos, retorno de stock, estado de la empresa, series temporales, a excepción de los estudios que extraen información de texto [RB01, RB09]. El análisis de textos requiere un conjunto diferente de características y técnicas como vectores de párrafo y oracionales. Todos los documentos revisados aplican el preprocesamiento de datos,



para transformar los datos en un formato comprensible, principalmente en los que se procesa texto, para lograr un rendimiento óptimo.

Los autores emplean algoritmos de DL para optimizar la selección de características; coinciden en que la selección correcta de características aumenta la precisión y el rendimiento del modelo. Los modelos más empleados son CNN y LSTM [RB17, RB04, RB12, RB14, RB18, RB03, RB12, RB09, RB05, RB07]. Algunos autores combinan varios modelos de DL para obtener mejores resultados [RB15, RB11, RB10, RB03], el modelo más repetido en estos estudios es Autoencoder para reducir la dimensionalidad y eliminar ruido en los datos.

En cuanto a las técnicas se identificó a ReLu como la función de activación más empleada en las redes neuronales, seguida por la función sigmoide.

5. Conclusiones y trabajo futuro

Los 20 estudios seleccionados permitieron identificar que las técnicas, métodos y modelos de DL también se pueden aplicar a la intermediación financiera. Con respecto a las preguntas de investigación, se encontraron aplicaciones para detectar riesgos financieros, predicciones en la dirección en el valor de las acciones del mercado, detección de fraudes en seguros, evasión de impuestos y pronóstico de ventas. En cuanto a los modelos y técnicas de DL se halló que las CNN y las LSTM son los modelos que más se emplean para resolver un problema financiero y que para cada estudio se aplican diferentes métodos y técnicas que les permiten extraer las características ideales para el modelo.

La revisión sistemática permitió conocer la aplicación actual de técnicas y modelos del DL en el área financiera, se resalta un mayor uso para extraer las características de manera óptima. Para estudios futuros debido al tiempo que consume el entrenar un modelo DL se recomienda desarrollar algoritmos que optimicen el tiempo y la extracción de características. En lo que respecta al campo de las finanzas se recomienda explorar la efectividad de DL en otras áreas como la gestión de cartera y las estrategias de negociación. En los estudios analizados se podrían adicionar otras variables para mejorar los modelos de predicción, relacionar las predicciones basadas en texto con las predicciones basadas en valores para determinar si existe mejora en la predicción de la bolsa mercantil.



Referencias

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [2] S. Gasparini et al., "Information Theoretic-Based Interpretation of a Deep Neural Network Approach in Diagnosing Psychogenic Non-Epileptic Seizures," *Entropy*, vol. 20, no. 2, p. 43, 2018.
- [3] D. Romo-Bucheli, A. Janowczyk, H. Gilmore, E. Romero, and A. Madabhushi, "A deep learning based strategy for identifying and associating mitotic activity with gene expression derived risk categories in estrogen receptor positive breast cancers," *Cytom. Part A*, vol. 91, no. 6, pp. 566–573, 2017.
- [4] Y. Cao et al., "Improving Tuberculosis Diagnostics Using Deep Learning and Mobile Health Technologies among Resource-Poor and Marginalized Communities," 2016 IEEE First Int. Conf. Connect. Heal. Appl. Syst. Eng. Technol., no. 1, pp. 274–281, 2016.
- [5] F. Li, J. Zhang, C. Shang, D. Huang, E. Oko, and M. Wang, "Modelling of a post-combustion CO₂capture process using deep belief network," *Appl. Therm. Eng.*, vol. 130, pp. 997–1003, 2018.
- [6] L. G. Pang, K. Zhou, N. Su, H. Petersen, H. Stöcker, and X. N. Wang, "An equation-of-state-meter of quantum chromodynamics transition from deep learning," *Nat. Commun.*, vol. 9, no. 1, 2018.
- [7] W. E, J. Han, and A. Jentzen, "Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations," *Communications in Mathematics and Statistics*, vol. 5, no. 4, pp. 349–380, 2017.
- [8] B. Kitchenham, "Procedures for performing systematic reviews," *Br. J. Manag.*, vol. 14, no. 0, pp. 207–222, 2003.
- [9] S. Rönnqvist and P. Sarlin, "Bank distress in the news: Describing events through deep learning," *Neurocomputing*, vol. 264, pp. 57–70, Nov. 2017.
- [10] Y. Zhao, J. Li, and L. Yu, "A deep learning ensemble approach for crude oil price forecasting," *Energy Econ.*, vol. 66, pp. 9–16, 2017.
- [11] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PLoS One*, vol. 12, no. 7, p. e0180944, Jul. 2017.
- [12] A. Dingli and K. S. Fournier, "Financial Time Series Forecasting – A Deep Learning Approach," *Int. J. Mach. Learn. Comput.*, vol. 7, no. 5, pp. 118–122, 2017.

- [13] R. Ferreira, M. Braga, and V. Alves, Forecast in the Pharmaceutical Area – Statistic Models vs Deep Learning, vol. 746. Springer International Publishing, 2018.
- [14] J. Lee, D. Jang, and S. Park, “Deep learning-based corporate performance prediction model considering technical capability,” *Sustain.*, vol. 9, no. 6, pp. 1–12, 2017.
- [15] Q. Yu, K. Wang, J. O. Strandhagen, and Y. Wang, “Application of Long Short-Term Memory Neural Network to Sales Forecasting in Retail—A Case Study,” pp. 11–17, 2018.
- [16] Y. Kaneko, “A Deep Learning Approach for the Prediction of Retail Store Sales,” *Icdmw*, 2016.
- [17] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, “Deep learning for stock prediction using numerical and textual information,” 2016 IEEE/ACIS 15th Int. Conf. Comput. Inf. Sci. ICIS 2016 - Proc., 2016.
- [18] O. Chang, I. Naranjo, C. Guerron, D. Criollo, J. Guerron, and G. Mosquera, “A Deep Learning Algorithm to Forecast Sales of Pharmaceutical Products,” no. August, 2017.
- [19] R. Singh and S. Srivastava, “Stock prediction using deep learning,” *Multimed. Tools Appl.*, vol. 76, no. 18, pp. 18569–18584, Sep. 2017.
- [20] S. Sohangir, D. Wang, A. Pomeranets, and T. M. Khoshgoftaar, “Big Data: Deep Learning for financial sentiment analysis,” *J. Big Data*, vol. 5, no. 1, 2018.
- [21] J. Sirignano, A. Sadhwani, and K. Giesecke, “Deep Learning for Mortgage Risk,” *Ssrn*, pp. 1–83, 2016.
- [22] X. Ding, Y. Zhang, T. Liu, and J. Duan, “Deep Learning for Event-Driven Stock Prediction,” no. Ijcai, pp. 2327–2333, 2015.
- [23] E. Chong, C. Han, and F. C. Park, “Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies,” *Expert Syst. Appl.*, vol. 83, pp. 187–205, 2017.
- [24] N. D. Goumagias, D. Hristu-Varsakelis, and Y. M. Assael, “Using deep Q-learning to understand the tax evasion behavior of risk-averse firms,” *Expert Syst. Appl.*, vol. 101, pp. 258–270, 2018.
- [25] H. Gunduz, Y. Yaslan, and Z. Cataltepe, “Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations,” *Knowledge-Based Syst.*, vol. 137, pp. 138–148, 2017.
- [26] M. Kraus and S. Feuerriegel, “Decision support from financial disclosures with deep neural networks and transfer learning,” *Decis. Support Syst.*, vol. 104, pp. 38–48, 2017.



- [27] C. Krauss, X. A. Do, and N. Huck, "Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500," *Eur. J. Oper. Res.*, vol. 259, no. 2, pp. 689–702, 2017.
- [28] Y. Wang and W. Xu, "Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud," *Decis. Support Syst.*, vol. 105, pp. 87–95, 2018.

Anexo 6: Certificado de Participación en el SIIPRIN'2018



Anexo 7: Certificado de Participación en el II Ciclo de Conferencias CIS-UNL 2019



Anexo 8: Declaración de Confidencialidad

Loja, 20 de marzo del 2019

Industria Lojana de Especerías ILE

A quien corresponda:

Por medio de la presente, yo, M. Cristina Suntaxi Sarango, a partir de la fecha de la firma de la presente carta. Me obligo a no divulgar la información de tipo confidencial de manera verbal o escrita de la empresa Industria Lojana de Especerías ILE con domicilio fiscal en el Barrio Consacola Km 1 Vía a Cuenca , Loja - Ecuador.

Dicha información incluye:

Datos acerca de las ventas de la empresa.

Información relacionada con los costos de producción.

Cualquier otra información que, por su naturaleza, se considere sensible y que se me hará notificación de forma escrita que no está sujeta a divulgación hacia la competencia o cualquier persona que no sea proveedor o colaboradora de esta organización.

En caso de que Industria Lojana de Especerías ILE detectara que estoy haciendo del conocimiento de terceros cualquier información relacionada con los términos antes expuesto, me haré acreedor a la sanción administrativa o legal que la empresa considere conveniente.

Atentamente

A handwritten signature in blue ink, enclosed in a circle. The signature appears to read "MARITA SUNTAXI".

Martha Cristina Suntaxi

Anexo 9: Licencia Creative Commons



“Detección de técnicas de aprendizaje profundo aplicadas en las diferentes áreas del conocimiento, empleando el método de revisión sistemática de literatura” está bajo una [licencia de Creative Commons Reconocimiento-CompartirIqual 4.0 Internacional.](#)