



UNIVERSIDAD TÉCNICA DE COTOPAXI

DIRECCIÓN DE POSGRADO

MAESTRÍA EN SISTEMAS DE INFORMACIÓN

MODALIDAD: PROPUESTA METODOLÓGICA Y TECNOLÓGICA AVANZADA

Título:

Algoritmos de Deep Learning utilizando Tensor Flow para el
tratamiento de datos de producción científica en la Universidad
Técnica de Cotopaxi

Trabajo de titulación previo a la obtención del título de magister en Sistemas de
Información

Autor:

Falconí Punguil Diego Geovanny

Tutor:

Rodríguez Bárcenas Gustavo PhD.

**LATACUNGA –ECUADOR
2021**

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Titulación “Algoritmos de Deep Learning utilizando Tensor Flow para el tratamiento de datos de producción científica en la Universidad Técnica de Cotopaxi” presentado por Falconí Punguil Diego Geovanny, para optar por el título magíster en Sistemas de Información.

CERTIFICO

Que dicho trabajo de investigación ha sido revisado en todas sus partes y se considera de que reúne los requisitos y méritos suficientes para ser sometido a la presentación para la valoración por parte del Tribunal de Lectores que se designe y su exposición y defensa pública.

Latacunga, abril, 05, 2021



Firmado electrónicamente por:

**GUSTAVO
RODRIGUEZ
BARCENAS**

.....
PhD. Gustavo Rodríguez Bárcenas
CC.: 1757001357

APROBACIÓN TRIBUNAL

El trabajo de Titulación: Algoritmos de Deep Learning utilizando Tensor Flow para el tratamiento de datos de producción científica en la Universidad Técnica de Cotopaxi, ha sido revisado, aprobado y autorizado su impresión y empastado, previo a la obtención del título de Magíster en Sistemas de Información; el presente trabajo reúne los requisitos de fondo y forma para que el estudiante pueda presentarse a la exposición y defensa.

Latacunga, junio, 28, 2021

.....
Dra. Albán Taípe Mayrá Susana
CC.: 0502311988
Presidente del tribunal

.....
Mg.C Llano Casa Alex Christian
CC.: 0502589864
Lector 2

.....
Dr. Cadena Moreano José Augusto
CC.: 0501552798
Lector 3

DEDICATORIA

A mi Dios porque siempre estuvo conmigo en los momentos que más lo necesitaba y me ayudo siempre a esforzarme a lograr avanzar hacia la meta.

A mis padres y hermano por su cariño y por siempre compartir cada momento que he logrado a lo largo de mi vida.

A mi querida Raquel, por ser una luz en mi vida y porque en este tiempo se ha convertido en una fuente de inspiración para lograr mis objetivos.

Diego Geovanny Falconí Punguil

AGRADECIMIENTO

Primeramente, agradezco a Dios por bendecirme, por la vida que me da, por guiarme a lo largo de mi existencia, por ser el apoyo y fortaleza en aquellos momentos de dificultad y de debilidad y porque a pesar de las circunstancias sé que siempre puedo confiar en Él.

A mis padres José y Gloria, quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por inculcar en mí el ejemplo de esfuerzo y valentía, de no temer las adversidades. A mi padre que siempre quiso verme realizado y me sirvió de ejemplo en todo momento, a mi madre por entregarme tanto amor.

A mi querida Raquel, quien ha sido una ayuda idónea al siempre tenerme presente en sus oraciones y por su amor y apoyo constante que me brinda en todo momento.

A mi tutor el Doctor Gustavo Rodríguez que con tanto agrado nos ha ayudado en el transcurso de este proyecto orientándonos sin ninguna negativa de su parte.

Diego Geovanny Falconí Punguil

RESPONSABILIDAD DE AUTORÍA

Quien suscribe, declara que asume la autoría de los contenidos y los resultados obtenidos en el presente trabajo de titulación.

Latacunga, febrero, 12, 2021.



.....
Diego Geovanny Falconí Punguil
CC.: 05500800774

RENUNCIA DE DERECHOS

Quien suscribe, cede los derechos de autoría intelectual total y/o parcial del presente trabajo de titulación a la Universidad Técnica de Cotopaxi.

Latacunga, febrero, 12, 2021

A handwritten signature in blue ink, appearing to read 'Diego Falconi', with several overlapping loops and strokes.

.....
Diego Geovanny Falconí Punguil
CC.: 0550080774

AVAL DEL VEEDOR

Quien suscribe, declara que el presente Trabajo de Titulación: Algoritmos de Deep Learning utilizando Tensor Flow para el tratamiento de datos de producción científica en la Universidad Técnica de Cotopaxi, contiene las correcciones a las observaciones realizadas por los lectores en sesión científica del tribunal.

Latacunga, junio, 28, 2021



.....
Dra. Albán Taípe Mayra Susana
CC.: 0502311988

**UNIVERSIDAD TÉCNICA DE COTOPAXI
DIRECCIÓN DE POSGRADO**

MAESTRÍA EN SISTEMAS DE INFORMACIÓN

Título: Algoritmos de Deep Learning utilizando Tensor Flow para el tratamiento de datos de producción científica en la Universidad Técnica de Cotopaxi

Autor: Falconí Punguil Diego Geovanny

Tutor: Rodríguez Bárcenas Gustavo PhD.

RESUMEN

La implementación de Inteligencia Artificial, Redes Neuronales y Algoritmos de Deep Learning apoyados en TensorFlow en la actualidad se encuentra en constante evolución ya que han abierto nuevas rutas para el tratamiento y análisis de grandes cantidades de datos en sistemas alojados en la web principalmente. Los algoritmos de aprendizaje profundo se encargan de entrenar y agrupar por similitud una data de entrada sin supervisión denominado aprendizaje automático, los mismos que modelan abstracciones de alto nivel utilizando principalmente datos expresados en forma matricial o tensores. La presente investigación tiene como la finalidad el ayudar el nivel de toma de decisiones no supervisados en la plataforma científica Ecuciencia, la misma que se encuentra alojado en los servidores de la Universidad Técnica de Cotopaxi. Los datos que se tomarán como referencia para los análisis introducidos en los algoritmos, será los referentes a Líneas y Sublíneas de Investigación de acuerdo a la Universidad Técnica de Cotopaxi. El impacto de la implementación de algoritmos de aprendizaje profundo apoyados en TensorFlow en el sistema Ecuciencia, será muy importante, puesto que, gracias a este análisis, la plataforma científica podrá ser capaz de dar una predicción más acertada de las clasificaciones de Líneas y Sublíneas de investigación.

PALABRAS CLAVE: Algoritmos; Redes Neuronales; Aprendizaje Profundo; TensorFlow; Ecuciencia; KDD.

**UNIVERSIDAD TECNICA DE COTOPAXI
DIRECCION DE POSGRADO**

MAESTRIA EN SISTEMAS DE INFORMACIÓN

**Title: DEEP LEARNING ALGORITHMS USING TENSORFLOW FOR
PROCESSING SCIENTIFIC PRODUCTION DATA**

Author: Falconí Punguil Diego Geovanny

Tutor: Rodríguez Bárcenas Gustavo PhD.

ABSTRACT

The Artificial Intelligence, Neural Networks and Deep Learning Algorithms implementation supported by Tensor Flow is currently in continuous evolution since have opened new routes for the treatment and analysis of data large amounts in systems mainly hosted on the web. Deep learning algorithms are responsible for training and grouping an unsupervised input data by similarity called machine learning, the same ones that model high-level abstractions using mainly data expressed in matrix form or tensors. This research purpose is to help the level of unsupervised decision making at Ecuciencia scientific platform, which is hosted on the Technical University of Cotopaxi servers. The data that will be taken as a reference for the analyzes introduced in the algorithms will be referring to Research Lines and Sublines according to the Technical University of Cotopaxi. The impact of Deep Learning Algorithms implementation supported by Tensor Flow in the Ecuciencia system will be very important, since, thanks to this analysis, the scientific platform will be able to give a more accurate prediction of the classifications of Research Lines and Sublines.

KEYWORD: Algorithms, Neural Networks, Deep Learning, TensorFlow, Ecuciencia, KDD.

Lidia Rebeca Yugla Lema con cédula de identidad número:050265234-0 Magister en Ciencias de la Educación con número de registro de la SENESCYT: 1027-15-86068398; **CERTIFICO** haber revisado y aprobado la traducción al idioma inglés del resumen del trabajo de investigación con el título: Algoritmos de Deep Learning utilizando Tensor Flow para el tratamiento de datos de producción científica en la Universidad Técnica de Cotopaxi de: Diego Geovanny Falconí Punguil ,aspirante a magister en Sistemas de Información

Latacunga, junio, 28, 2021



Mg. Lidia Rebeca Yugla Lema
DOCENTE CENTRO DE IDIOMAS
C.C. 050265234-0



**CENTRO
DE IDIOMAS**

ÍNDICE DE CONTENIDO

INTRODUCCIÓN	1
CAPÍTULO I.....	8
1.1. ANTECEDENTES	8
1.1.1. SCIMAGO JOURNAL	8
1.1.2. SCIELO.....	8
1.1.3. CIENCIOMETRÍA	9
1.1.4. INDICADORES CIENCIOMÉTRICOS	10
1.1.5. LA CIENCIOMETRÍA EN AMÉRICA LATINA	12
1.1.6. DATACIENCIA	12
1.1.7. REDCIENCIA	13
1.1.8. REDSEARCH.....	13
1.1.9. REDALYC.ORG	13
1.2. FUNDAMENTACIÓN EPISTEMOLÓGICA.....	14
1.2.1. SISTEMAS INFORMÁTICOS	14
1.2.2. SISTEMAS INFORMÁTICOS EN LA WEB.....	14
1.2.3. BASE DE DATOS.....	15
1.2.4. POSTGRESQL	16
1.2.5. SOFTWARE	18
1.2.6. CICLO DE VIDA DEL DESARROLLO DE SOFTWARE	19
1.2.7. LENGUAJE DE PROGRAMACIÓN	20
1.2.8. PYTHON	20
1.2.9. LIBRERÍAS DE PYTHON	22
1.2.10. FRAMEWORK DJANGO.....	24
1.2.11. MINERÍA DE DATOS	26
1.2.12. MINERÍA DE TEXTO	27

1.2.13. ALGORITMO.....	28
1.2.14. ALGORITMOS EN MINERÍA DE DATOS	29
1.2.15. AUTOAPRENDIZAJE	30
1.2.16. TOMA DE DECISIONES	30
1.2.17. REDES NEURONALES	31
1.2.18. REDES NEURONALES ARTIFICIALES	32
1.2.19. APRENDIZAJE PROFUNDO	33
1.2.20. TENSORFLOW	33
1.3. FUNDAMENTACIÓN DEL ESTADO DEL ARTE	34
1.4. CONCLUSIONES CAPÍTULO I	36
CAPÍTULO II	37
2.1. DIAGNÓSTICO.....	37
2.2. MÉTODOS ESPECÍFICOS DE LA INVESTIGACIÓN	38
2.2.1. METODOLOGÍA KDD.....	38
2.2.2. METODOLOGÍA MODELO ITERATIVO INCREMENTAL	41
2.3. DISEÑO EXPERIMENTAL Y MÉTODO DE CRITERIO DE EXPERTOS.....	45
2.3.1. MÉTODO DE VALIDACIÓN CRUZADA	45
2.4. DESCRIPCIÓN METODOLÓGICA DE LA VALORIZACIÓN ECONÓMICA, TECNOLÓGICA, OPERACIONAL Y MEDIO AMBIENTAL DE LA PROPUESTA.	46
2.4.1. VALORACIÓN ECONÓMICA:.....	46
2.4.2. VALORACIÓN TECNOLÓGICA:.....	47
2.4.3. VALORACIÓN AMBIENTAL:.....	47
2.5. CONCLUSIONES CAPÍTULO II	47
CAPÍTULO III.....	48
3.1. RESULTADOS DE DIAGNÓSTICO DEL PROBLEMA.....	48

3.1.1. TÉCNICAS DE INVESTIGACIÓN.....	49
3.2. RESULTADOS DE LOS MÉTODOS ESPECÍFICOS	50
3.2.1. METODOLOGÍA KDD.....	50
3.2.2. MODELO ITERATIVO INCREMENTAL.....	65
3.3. RESULTADOS DEL DISEÑO EXPERIMENTAL Y/O MÉTODO DE CRITERIO DE EXPERTOS	76
3.3.1. MÉTODO DE VALIDACIÓN CRUZADA	76
3.4. RESULTADOS DE LA VALORIZACIÓN ECONÓMICA, TECNOLÓGICA, OPERACIONAL Y AMBIENTAL.....	83
3.4.1. VALORACIÓN ECONÓMICA:.....	83
3.4.2. VALORACIÓN TECNOLÓGICA:.....	84
3.4.3. VALORACIÓN AMBIENTAL:.....	85
3.5. DISCUSIÓN DE LA APLICACIÓN Y VALIDACIÓN DE LA PROPUESTA	86
3.6. CONCLUSIONES CAPÍTULO III.....	87
CONCLUSIONES GENERALES	88
RECOMENDACIONES.....	89
REFERENCIAS BIBLIOGRÁFICAS.....	90
ANEXOS.....	98
ANEXO I: ENTREVISTA	99
ANEXO II: DIAGRAMAS DE LA ETAPA DE ANÁLISIS DE DESARROLLO DE SOFTWARE.....	102
DIAGRAMA DE CASOS DE USO	102
DIAGRAMAS DE ACTIVIDADES	103
DIAGRAMAS DE SECUENCIA.....	111
DIAGRAMAS DE ENTIDAD - RELACIÓN.....	115
ANEXO III: DISEÑO DE MAQUETACIÓN DE INTERFACES.....	116

MAQUETADOS DE INTERFACES DE USUARIO	116
ANEXO IV: CÓDIGO DE PROGRAMACIÓN.	119
CÓDIGO PYTHON	119
CÓDIGO JAVASCRIPT	120
CÓDIGO HTML.....	121
ANEXO V: CAPTURA DE INTERFACES GRÁFICAS.	123
INTERFACES DE USUARIO	123
ANEXO VI: MATRICES DE CASOS DE PRUEBAS.	126
CASOS DE PRUEBAS	126
ANEXO VII: CONSULTA SQL Y RESULTADO.....	134
ANEXO VIII: LOGS DEL MÉTODO DE VALIDACIÓN.	135

ÍNDICE DE TABLAS

TABLA 1 SISTEMA DE TAREAS	4
TABLA 2 CARACTERÍSTICAS DE POSTGRESQL	17
TABLA 3 ESCALA DE FIABILIDAD.....	46
TABLA 4 PARÁMETROS ENVIADOS AL ALGORITMO PARA SU EJECUCIÓN.	58
TABLA 5 INFORMACIÓN DE INVOLUCRADOS.	68
TABLA 6 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS	68
TABLA 7 REFERENCIAS	68
TABLA 8 USUARIOS	70
TABLA 9 RF01	70
TABLA 10 RF02	71
TABLA 11 RF03	71
TABLA 12 RF04	71

TABLA 13 RF05	72
TABLA 14 RF06	72
TABLA 15 RNF01.....	73
TABLA 16 RNF02.....	73
TABLA 17 RNF03.....	74
TABLA 18 RNF04.....	74
TABLA 19 RNF05.....	74
TABLA 20 GASTOS DIRECTOS	83
TABLA 21 GASTOS INDIRECTOS	84
TABLA 22 GASTOS TOTALES	84
TABLA 23 REQUISITOS MÍNIMOS DE HARDWARE.....	85
TABLA 24 REQUISITOS MÍNIMOS DE HARDWARE.....	85

ÍNDICE DE FIGURAS

FIGURA 1 ESQUEMA DE BASE DE DATOS	16
FIGURA 2 LENGUAJES DE PROGRAMACIÓN MÁS POPULARES.....	22
FIGURA 3 COMPARACIÓN DE NEURONA CEREBRAL Y REDES NEURONALES ARTIFICIALES	31
FIGURA 4 RED NEURONAL ARMADA POR CIENTÍFICOS.....	32
FIGURA 5 ETAPAS DE LA METODOLOGÍA KDD.....	39
FIGURA 6 ETAPAS DE LA METODOLOGÍA DEL MODELO ITERATIVO INCREMENTAL	41
FIGURA 7 TABLAS DE BASE DE DATOS ECUCIENCIA.....	51
FIGURA 8 INDICADORES CIENCIOMÉTRICOS	51
FIGURA 9 DIAGRAMA ENTIDAD - RELACIÓN.....	52

FIGURA 10 DEFINICIÓN DE TABLAS Y ATRIBUTOS.....	54
FIGURA 11 MATRIZ DE CONFUSIÓN	55
FIGURA 12 TEXTO DE RESUMEN DE ARTÍCULO CIENTÍFICO.....	56
FIGURA 13 PALABRAS EXTRAÍDAS AL APLICAR EL ALGORITMO NLTK.....	56
FIGURA 14 LÍNEAS DE CÓDIGO DEL CONSTRUCTOR DEL ALGORITMO NLTK PARA ANÁLISIS EN ESPAÑOL.....	56
FIGURA 15 TEXTO DE RESUMEN DE ARTÍCULO CIENTÍFICO EN INGLÉS	57
FIGURA 16 PALABRAS EXTRAÍDAS AL APLICAR EL ALGORITMO NLTK.....	57
FIGURA 17 INICIO DE LOG PRODUCIDO POR EL ANÁLISIS DE TENSORFLOW.....	60
FIGURA 18 FIN DE LOG PRODUCIDO POR EL ANÁLISIS DE TENSORFLOW CON EL LÍMITE DE 89 ÉPOCAS	60
FIGURA 19 RENDERIZADO DE DATOS DESDE VISTA A TEMPLATE	61
FIGURA 20 CAPTURA DE DATOS ENVIADOS DESDE LA VISTA.....	61
FIGURA 21 LOG DE “CONFUSIONMATRIX”	62
FIGURA 22 LOG DE “LABELS”.....	62
FIGURA 23 LOG DE VALORES PORCENTUALES PRODUCIDOS POR LAS ÉPOCAS	62
FIGURA 24 GRÁFICA DE REPRESENTACIÓN DE VALORES PORCENTUALES.....	62
FIGURA 25 VARIABLES DE EVALUACIÓN DEL MODELO.....	63
FIGURA 26 CÓDIGO DE PREDICCIÓN DE NUEVO TEXTO.....	63
FIGURA 27 DIAGRAMA DE PASTEL CON VALORES PORCENTUALES DE PREDICCIÓN.....	64

FIGURA 28 DIAGRAMA DE BARRAS CON VALORES PORCENTUALES DE PREDICCIÓN	64
FIGURA 29 PORCENTAJE DE PREDICCIÓN DE CLASE CON MAYOR PESO	65
FIGURA 30 METODOLOGÍA KDD – ITERATIVA INCREMENTAL.....	66
FIGURA 31 DIAGRAMA DE ITERACIONES	66
FIGURA 32 RESULTADO DE VALIDACIÓN “HOLD-OUT”	76
FIGURA 33 RESULTADO DE VALIDACIÓN “K-FOLD” CON K = 25	77
FIGURA 34 REPRESENTACIÓN VISUAL DEL MÉTODO “HOLD-OUT” ...	78
FIGURA 35 REPRESENTACIÓN VISUAL DEL MÉTODO “K-FOLD”	78
FIGURA 36 CÁLCULO DE DESVIACIÓN ESTÁNDAR.....	79
FIGURA 37 GRÁFICO DE DESVIACIÓN ESTÁNDAR	80
FIGURA 38 CÁLCULO DE DISTRIBUCIÓN NORMAL	81
FIGURA 39 REPRESENTACIÓN DE DISTRIBUCIÓN NORMAL.....	81
FIGURA 40 CAMPANA DE GAUSS	82
FIGURA 41 PROBABILIDAD DE QUE EL RESULTADO SEA MAYOR AL 90%	82

INTRODUCCIÓN

Antecedentes: La presente propuesta corresponde a la Línea de Investigación Tecnologías de Información y Comunicación y a la Sublínea de Investigación Inteligencia Artificial e inteligencia de negocios para toma de decisiones.

En la Universidad Autónoma de Madrid, conscientes de la importancia de recopilar la producción científica de sus investigadores, se han desarrollado una serie de plataformas que recogen toda la actividad científica de los investigadores (Portal de Producción Científica) y que almacenan los textos completos de las publicaciones en las que se plasma esta producción, en acceso abierto, a través de su repositorio institucional Biblos-e Archivo. [1]

Según [2], en el Ecuador, el objetivo principal de las universidades hasta antes de la década de los setenta era la docencia, sin dar ningún énfasis en el tema de métodos de investigación científica, por lo que el número de investigación bibliográficas era casi nulo. Sin embargo, desde el año 2008 la actividad científica de las universidades del país ha reflejado un incremento positivo en cuanto a su desarrollo. [3]

Prueba de ello son las estadísticas de publicaciones en las bases de datos de Scopus, ya que según [4], las publicaciones que se realizaron en el periodo 2004-2008 solo reportaron 32 instituciones educativas y un total de 866 artículos publicados. En el periodo del 2009 al 2013 las publicaciones aumentaron significativamente pues se llegaron a registrar alrededor de 1992 artículos científicos. Más adelante en los años 2014 y 2015 se evidencia un total de 976 y 1174 artículos publicados, los mismo que superan significativamente los periodos antes descritos. Además de estos datos estadísticos, la revista estadounidense Nature, publica un ranking de artículos registrados con bases científicas, y en el mismo destacan tres universidades ecuatorianas, en primer lugar, la Pontificia Universidad Católica, en segundo la Universidad de Investigación de Tecnología Experimental (Yachay) y, por último, la Escuela Politécnica Nacional. [3]

Según [5], las bases del desarrollo de nuevas tecnologías son impulsadas mayormente por el conocimiento. Gracias a los repositorios que contienen un

conjunto de documentación científica, el ser humano tiene la capacidad de ir desarrollando nuevos datos que fortalezcan la gestión del conocimiento. Además, dichos repositorios permitirán al ser humano realizar una búsqueda de información de una manera más rápida y eficiente, sin necesidad de limitaciones para ningún individuo dentro de la comunidad científica.

Plataformas como SCImago Journal & Country Rank, RedSearch, entre otras, fueron desarrolladas con el objetivo de almacenar grandes repositorios de artículos con alto nivel científico, los mismo brindan a la comunidad científica datos evaluativos del estado de la ciencia, detallados en valores cuantitativos y cualitativos, sin embargo, uno de sus inconvenientes es que sus reportes ofrecen una evaluación y valoración a nivel global, careciendo totalmente de una valoración a nivel local.

Conociendo estos acontecimientos consideramos el siguiente **Planteamiento del Problema**; En la Universidad Técnica de Cotopaxi ubicada en la Av. Simón Rodríguez, barrio El Ejido sector San Felipe, del cantón Latacunga, provincia de Cotopaxi, se está desarrollando una cultura investigativa a través de la creación y recreación de ciencia, tecnología y arte, como la formación científica, generación, difusión y promoción de los saberes y conocimientos, que coadyuven al desarrollo sostenible y sustentable del entorno, con enfoque investigativo progresista y dedicado a promover la sostenibilidad productiva, ambiental y la equidad social de la región y el país.

Todas las investigaciones desarrolladas en la Universidad Técnica de Cotopaxi, son documentadas mediante artículos, libros, ponencias y proyectos; que requieren ser almacenados y visualizados por la comunidad universitaria. Actualmente el Alma Máter contiene una plataforma de gestión del conocimiento llamada Ecuciencia, la misma que está recopilando la producción científica y tecnológica de todas las disciplinas que se estudian en las distintas facultades existentes en la institución, a partir de indicadores cuantitativos.

Toda la información almacenada en la base de datos de la plataforma Ecuciencia, requiere ser visualizada en herramientas que el usuario pueda entender con facilidad, partiendo de estas características, surge la necesidad de establecer un

tratamiento de datos almacenados en el repositorio digital, ya que, si bien es cierto existe una clasificación controlada, la plataforma carece de una herramienta que ayude a gestionar los datos de manera automática, por lo que aún no cuenta con una característica primordial de aprendizaje de máquina.

Este problema se puede apreciar en la clasificación y control de información, puesto que no existe la coherencia de datos proporcionados en la plataforma en cuanto a los documentos científicos y su relación con las líneas de investigación a las que pertenecen, generando de esta manera una inconsistencia de datos al momento de mostrar la información.

Ejemplificado el postulado anterior, se puede tomar el caso de la clasificación con las palabras claves de los documentos científicos, existen varias palabras en común pero que tienen diferente contexto dependiendo de la investigación y la línea a la que pertenece, sin embargo el sistema aún no puede reconocer este tipo de diferenciadores claves, excluyendo las palabras de uso común para su clasificación, es por esta razón que aún no se mantiene una concordancia de datos en la clasificación de textos científicos.

Por tales razones se procede a la **Formulación del Problema** de investigación:

¿Cómo aportar en el Sistema Ecucienca para mejorar el tratamiento de la información almacenada y la clasificación no supervisada correspondiente a datos de producción científica de la plataforma, donde se evidencia una inconsistencia de datos, específicamente con la relación existente entre documentos científicos y sublíneas de investigación?

El **Objetivo General** planteado es aplicar algoritmos de Deep Learning utilizando la herramienta TensorFlow para el tratamiento de datos de producción científica en la Universidad Técnica de Cotopaxi.

Por su parte los **Objetivos Específicos** propuestos para la investigación son:

- Analizar los antecedentes teóricos relacionado con la Cienciometría y su impacto en sistemas basados en inteligencia artificial, a partir de fuentes

bibliográficas certificadas que sirvan de base para la elaboración de la fundamentación teórica o científica de la investigación.

- Investigar antecedentes de los proyectos relacionados con Ecuciencia, mediante documentos y marcos referenciales, para poder interpretar un diagnóstico del funcionamiento de la plataforma científica.
- Investigar metodologías de minería de datos e inteligencia artificial, utilizando información de proyectos similares, para efectuar la aplicación de los mismos en la ejecución del proyecto.
- Realizar una evaluación de factibilidad de la ejecución del proyecto, utilizando métodos de validación para establecer la viabilidad de la propuesta.

A continuación, se detallan las **Tareas** necesarias para la ejecución de la propuesta de investigación.

Tabla 1 Sistema de Tareas

Objetivo	Actividad (tareas)
1. Objetivo específico 1: Analizar los antecedentes teóricos relacionado con la Cienciometría y su impacto en sistemas basados en inteligencia artificial, a partir de fuentes bibliográficas certificadas que sirvan de base para la elaboración de la fundamentación teórica o científica de la investigación.	1. Revisión de fuentes bibliográficas certificadas.
	2. Seleccionar información que tenga más semejanza con la investigación a desarrollar.
	3. Establecer criterios personales del contenido científico de las investigaciones realizadas.
2. Objetivo específico 2: Investigar antecedentes de los proyectos relacionados con Ecuciencia, mediante documentos y marcos referenciales,	1. Revisión de la información relacionada con los algoritmos de Deep Learning y sus principales funcionalidades.

Objetivo	Actividad (tareas)
para poder interpretar un diagnóstico del funcionamiento de la plataforma científica.	2. Revisión de documentación registrada en investigaciones anteriores en el proyecto Ecuciencia.
	3. Verificación y validación del nivel de aprendizaje de máquina de Ecuciencia.
3. Objetivo específico 3: Investigar metodologías de minería de datos e inteligencia artificial, utilizando información de proyectos similares, para efectuar la aplicación de los mismos en la ejecución del proyecto.	1. Optar por las técnicas de investigación necesarias para desarrollar la investigación.
	2. Investigar proyectos científicos relacionados minería de datos e inteligencia artificial y sus metodologías aplicadas.
	3. Elaborar etapas metodológicas que se relacionen con la minería de datos.
4. Objetivo específico 4: Realizar una evaluación de factibilidad de la ejecución del proyecto, utilizando métodos de validación para establecer la viabilidad de la propuesta.	1. Argumentar los resultados obtenidos en el desarrollo de esta investigación.
	2. Realizar un balance de presupuesto y tiempos que probablemente se invierta en la ejecución del proyecto.
	3. Establecer un criterio de viabilidad de ejecución del proyecto.

Elaborado por: Investigador

Se establece como **Justificación** que la implementación de Inteligencia Artificial, Redes Neuronales y Algoritmos de Deep Learning apoyados en TensorFlow en la actualidad se encuentra en constante evolución ya que han abierto nuevas rutas para el tratamiento y análisis de grandes cantidades de datos en sistemas alojados en la web principalmente. [5] Los algoritmos de aprendizaje profundo se encargan de entrenar y agrupar por similitud una data de entrada sin supervisión denominado

aprendizaje automático, los mismos que modelan abstracciones de alto nivel utilizando principalmente datos expresados en forma matricial o tensores. [6]

La presente propuesta de investigación tiene como finalidad el ayudar el nivel de toma de decisiones no supervisadas en la plataforma científica Ecuciencia, la misma que se encuentra alojada en los servidores de la Universidad Técnica de Cotopaxi. Los datos que se tomarán como referencia para los análisis introducidos en los algoritmos, serán los referentes a Líneas y Sublíneas de Investigación de acuerdo a la Universidad Técnica de Cotopaxi. La data a ser manipulada para el proceso de análisis será tomada de los artículos, libros y ponencias científicas publicadas por docentes investigadores de la Universidad Técnica de Cotopaxi en revistas científicas, las cuales serán analizadas por la calidad de las mismas.

El impacto de la implementación de algoritmos de aprendizaje profundo apoyados en TensorFlow en el sistema Ecuciencia, será muy importante, puesto que, gracias a este análisis, la plataforma científica podrá ser capaz de dar una predicción más acertada de las clasificaciones de Líneas y Sublíneas de investigación. Actualmente la plataforma científica no consta con algún tipo de algoritmo que apoye al usuario en cuanto a escoger a qué Línea o Sublínea pertenece su trabajo, y lo que se pretende es que mediante los algoritmos proporcionados por TensorFlow, el sistema tenga la capacidad de predecir la clase y poder dar una respuesta correcta al usuario, además los algoritmos serán capaces de establecer un horario determinado para su retroalimentación, y de esta manera asegurar que los datos analizados, tenga la mayor cantidad de porcentaje de predicción correcta.

Existen muchas **Metodologías** aplicables para todo lo que abarca la Inteligencia Artificial, una de las más conocidas es la metodología KDD la misma que establece todos los pasos necesarios para tener un buen proceso de Inteligencia Artificial. El uso de esta metodología será de mucha ayuda puesto que la misma abarca los procesos desde la concepción y carga de datos, continuando por su posterior entrenamiento y agrupamiento, para finalmente validar y visualizar los resultados. Dado a su alto potencial y compatibilidad en el uso de redes neuronales e inteligencia artificial, esta metodología será de gran ayuda para el manejo de grandes cantidades de datos. Además, se pretende realizar un nexo entre la

metodología KDD que es especializada en Inteligencia Artificial, y la Metodología del modelo Iterativo e Incremental, la misma que permite tener un mayor enfoque en la parte correspondiente al desarrollo de Software de la presente propuesta.

El método de Minería de Datos será también una pieza clave para el desarrollo de la propuesta, ya que será necesario el poder realizar un filtro de solo la información útil para el análisis de datos que serán procesados por los algoritmos de aprendizaje profundo. Con la minería de datos, lo que se pretende es poder tomar una muestra del total de datos existentes, la misma que permita realizar el proceso de evaluación y testing dentro de la programación de la propuesta.

La propuesta de investigación está enfocada al análisis de requerimientos planteados por el usuario encargado de la plataforma científica Ecuciencia, por lo cual resulta prudente tener el primer acercamiento real a lo propuesto mediante la investigación explicativa, esta técnica muestra los procesos realizados en la producción científica en la Universidad Técnica de Cotopaxi. Además, se aplicará la investigación descriptiva como siguiente paso para obtener la situación real de dicha producción, ayudando a identificar la problemática suscitada en el sistema. Además, se considera necesario la revisión en trabajos de investigación antecesoras del proyecto y tener un acercamiento con las personas encargadas, para así poder diseñar de mejor manera cuál es el estado actual de la aplicación y cuáles son los datos que podrían ser claves para su análisis.

Los beneficiarios directos serán los docentes investigadores de la Universidad Técnica de Cotopaxi, los mismos que serán tomados en cuenta como usuarios del sitio web Ecuciencia debido a su importancia en el proceso de producción científica. Además, se considera también un beneficiario directo a los estudiantes de la Universidad Técnica de Cotopaxi, puesto que son los más interesados en la búsqueda de información en la plataforma.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA

1.1. ANTECEDENTES

1.1.1. SCImago Journal

SCImago Journal & Country Rank es un portal de indicadores cuantitativos e informáticos que permite a investigadores, editores, especialistas en información y decisores en materia de política científica, en especial de los países subdesarrollados, seguir el comportamiento y el impacto de sus contribuciones a escala internacional. Para esto emplea la amplia colección de literatura disponible en Scopus de Elsevier. [6] Scopus es una base de datos bibliográfica de resúmenes y citas de artículos de revistas científicas y cubre más de 18 mil revistas siendo más del 90% de ellas del tipo arbitradas y pertenecientes a las áreas de ciencias, tecnología, medicina, ciencias sociales, artes y humanidades. [7]

La plataforma ha sido desarrollada por SCImago Research Group, un grupo de investigación de las universidades de Granada, Extremadura, Carlos III de Madrid y Alcalá de Henares de España, y es hoy en día la plataforma más inclusiva disponible para publicaciones. En su plataforma se encuentran ranking de impacto de las revistas y también de las instituciones de donde provienen los autores. SCImago incluye también un mapa que permite visualizar la investigación que se realiza en los países iberoamericanos y publica todos los años el Ranking de Revistas y Países de SCImago. [7]

1.1.2. SciELO

SciELO (Scientific Electronic Library Online o Biblioteca Científica Electrónica en Línea) es un proyecto de biblioteca electrónica, iniciativa de la Fundación para el Apoyo a la Investigación del Estado de São Paulo, Brasil (Fundação de Amparo à Pesquisa do Estado de São Paulo — FAPESP) y del Centro Latinoamericano y del Caribe de Información en Ciencias de la Salud (BIREME), [8], que permite la publicación electrónica de ediciones completas de las revistas científicas mediante

una plataforma de software que posibilita el acceso a través de distintos mecanismos, incluyendo listas de títulos y por materia, índices de autores y materias y un motor de búsqueda.

El proyecto SciELO, que además cuenta con el apoyo de diversas instituciones nacionales e internacionales vinculadas a la edición y divulgación científica,[9], tiene como objetivo el «desarrollo de una metodología común para la preparación, almacenamiento, diseminación y evaluación de la literatura científica en formato electrónico». Actualmente participan en la red SciELO los siguientes países: Sudáfrica, Argentina, Brasil, Chile, Colombia, Costa Rica, Cuba, España, México, Perú, Portugal, Venezuela; además se encuentran en fase de desarrollo: Bolivia, Paraguay y Uruguay.[10]

1.1.3. Cienciometría

La cienciometría estudia los aspectos cuantitativos de la ciencia como disciplina o actividad económica, forma parte de la sociología de la ciencia y encuentra aplicación en el establecimiento de las políticas científicas, donde incluye entre otras las de publicación. Ella emplea, al igual que las otras dos disciplinas estudiadas, técnicas métricas para la evaluación de la ciencia (el término ciencia se refiere, tanto a las ciencias naturales como a las sociales), y examina el desarrollo de las políticas científicas de países y organizaciones. [11]

A mediados de la década de 1970, se comenzó a reconocer la importancia del análisis cuantitativo de las actividades de ciencia y tecnología como un instrumento útil y eficaz en el aparato público ligado a la política y la planificación. La evaluación de la investigación a través de indicadores cuantitativos ha llegado a ser parte constitutiva de la agenda de la política científica en todo el mundo.[12]

La cienciometría es la ciencia que se encarga del estudio de la producción científica y tecnológica, a través de indicadores que permiten medir y analizar el impacto que genera en la sociedad las investigaciones desarrolladas. Para ejecutar el proceso de la obtención de similitud y distancia entre investigadores, es necesario realizar un estudio previo de la cienciometría, para en base a sus indicadores seleccionar las características correctas de los objetos de estudio.

1.1.4. Indicadores cientiométricos

La evaluación del impacto científico es la valoración que se realiza a través de diferentes indicadores cientiométricos para determinar la novedad y el aporte teórico de los nuevos conocimientos producidos por las investigaciones, a partir de la constatación de los resultados obtenidos, de acuerdo con la intención inicial. [13]

La investigación contribuye resultados probados en diferentes áreas de la ciencia, los mismos que son evaluados, mediante indicadores cientiométricos, que establece parámetros y técnicas, que permiten dar una valoración de la calidad y el impacto que genera en la sociedad la producción científica y tecnológica de los investigadores.

Los indicadores cientiométricos pueden dividirse en dos grandes grupos: los que miden la calidad y el impacto de las publicaciones científicas (indicadores de publicación), y aquellos que miden la cantidad y el impacto de las vinculaciones o relaciones entre las publicaciones científicas (indicadores de citación). En función de estos grupos, se consideran los siguientes indicadores: [14] [15]

- Indicadores de actividad científica, basados en el recuento de publicaciones científicas o patentes de la unidad objeto del estudio. Permiten la realización de series temporales, distribución geográfica, por tipo de institución o por temas de investigación.
- Indicadores de impacto o influencia. Se trata de encontrar medidas indirectas de la calidad intrínseca de los trabajos, como puede ser el uso que la comunidad científica hace de un determinado documento, su impacto o influencia.
- Indicadores de tipo de investigación.
- Indicadores basados en coautoría o Índice de firmas por trabajo, o Colaboración entre departamentos de una institución, entre distintas instituciones, o entre varias ciudades de un país o entre diversos países. A través de las bases de datos en las que figuran las direcciones de todos los autores se pueden determinar redes de colaboración que pueden ser indicativas de la madurez de un sistema investigador, favorecen los intercambios de conocimiento y aumentan la visibilidad.

- Indicadores basados en asociaciones temáticas: mediante un complejo tratamiento matemático se logra una reducción de los datos y una visualización de la estructura de la ciencia y la tecnología y su evolución a través de mapas. Estos pueden ser:
 - De referencias bibliográficas comunes (enlace bibliográfico) permite seleccionar artículos de temática coherente.
 - De citas comunes relacionan temas con una base intelectual común, la constituida por esos artículos fuente que forman el "frente de investigación". Los clústeres pueden identificar especialidades, aunque con una demora temporal.
 - De palabras comunes a través de los términos de indización o lenguaje libre, reflejan la red de relaciones conceptuales; los mapas muestran las interrelaciones de la investigación actual y se pueden aplicar a artículos o patentes.
 - De clasificaciones comunes, la coocurrencia de clasificaciones de artículos o patentes define interrelaciones similares a las de las palabras clave.

- Indicadores de innovación tecnológica basados en recuentos de las patentes solicitadas o concedidas a través de bases de datos especializadas o de las citas en patentes a la literatura científica. Los tipos de análisis que emplean indicadores basados en patentes se pueden estructurar en:
 - Cuantificación de la actividad tecnológica internacional, de un país, sector industrial o empresa y la apertura de nuevos mercados.
 - Evaluación de resultados de los programas de investigación tecnológica, o Estudio de la interfaz entre ciencia y tecnología a través de las citas en primera página de patentes americanas o European Search Report de EPO.
 - Análisis de clúster mediante coocurrencia de citas, palabras o clasificaciones a través de mapas que descubren estructuras de las actividades tecnológicas.

Cada día surgen nuevos indicadores como resultado del desarrollo de las técnicas de análisis y representación de la información, y esto conduce a una revolución en el campo de la bibliotecología y las ciencias de la Información, que facilita la cuantificación de áreas como las ciencias sociales, enfocadas a medir, no sólo la cantidad, sino la calidad de los resultados de la actividad científica. [15]

1.1.5. La cienciometría en América Latina

Como punto de “inflexión” del proceso de estructuración de la cienciometría en América Latina, se podría establecer el año 1995, cuando se creó la Red Iberoamericana de Indicadores de Ciencia y Tecnología (RICYT), auspiciada por el Programa Iberoamericano de Ciencia y Tecnología para el Desarrollo (CYTED) programa perteneciente a la UNESCO y la OEA. Su objetivo central era y sigue siendo el de apoyar técnicamente a los países integrantes para que mejoren en materia de información en el ámbito de la ciencia, la tecnología y la innovación. [12]

En América Latina, el hecho de no haber podido avanzar de forma adecuada en materia de cienciometría se ha convertido en una de las mayores debilidades de los sistemas de ciencia, tecnología e innovación. Carecer de canales formales de interacción que promovieran objetivos colectivos, que apuntarán a un progreso sostenido de esos países utilizando como plataforma el diseño de políticas públicas basadas en información adecuada para tomar decisiones “confiables” en el avance de las actividades tecnocientíficas, se ha transformado en una de las causas de su atraso. [12]

Para corroborar los antecedentes previamente plasmados, se realizó una investigación de las plataformas de visualización científicas dentro de América Latina, a continuación de mencionan las varias de ellas:

1.1.6. DataCiencia

Es una plataforma de visualización de las dimensiones de la producción científica de Chile la que pretende relevar y visualizar la actividad científica de un modo comprensivo y sistémico. En este contexto, no se trata de un ranking de instituciones ni de personas. [16]

Esta herramienta permite visualizar, cuantificar y caracterizar la producción científica chilena segmentada en cuatro grandes categorías: Investigadores, Territorio (Regiones), Instituciones y Revistas Científicas. Todo esto a partir de la base de datos Web of Science (WoS) de Thomson Reuters que contiene la producción científica nacional del período 2008-2016. [16]

1.1.7. RedCiencia

Es un canal de comunicación y encuentro entre quienes viven la ciencia. Un espacio para destacar y diseminar el quehacer de investigadores, estudiantes y profesionales de todas las áreas del conocimiento, tanto a nivel nacional como internacional. Un lugar donde encontrar oportunidades de crecimiento profesional, desde una mirada colaborativa e inclusiva. [17]

1.1.8. RedSearch

Es una herramienta que permite visualizar las relaciones de coautoría de documentos científicos chilenos del período 2008-2016 indizada en Web of Science. Mediante el análisis de estas relaciones de coautoría, la RedSearch entrega al usuario varias métricas relacionadas con la red, pero también con los autores que la conforman. [18]

1.1.9. Redalyc.org

Es un proyecto académico para la difusión en Acceso Abierto de la actividad científica editorial que se produce en Iberoamérica. Es, en principio, una hemeroteca científica en línea de libre acceso y un sistema de información científica, que incorpora el desarrollo de herramientas para el análisis de la producción, la difusión y el consumo de literatura científica. [19]

El nombre Redalyc viene de Red de Revistas Científicas de América Latina, el Caribe, España y Portugal. El proyecto, impulsado por la Universidad Autónoma del Estado de México (en colaboración con cientos de instituciones de educación superior, centros de investigación, asociaciones profesionales y editoriales iberoamericanas), surge en el año 2003 como iniciativa de un grupo de investigadores y editores preocupados por la escasa visibilidad de los resultados de investigación generados en y sobre la región. Se ha propuesto, desde su creación,

ser un punto de encuentro para los interesados en reconstruir el conocimiento científico de y sobre Iberoamérica. [19]

1.2. FUNDAMENTACIÓN EPISTEMOLÓGICA

1.2.1. SISTEMAS INFORMÁTICOS

Según [20] en 2018, los sistemas de información constituyen una de las principales áreas de investigación en el campo de la organización empresarial. El entorno en el que las empresas desarrollan sus actividades es cada vez más complejo. Según [21], un sistema de información se define como: un conjunto de procesos formales que operan sobre un conjunto de datos construido de acuerdo a las necesidades de la empresa, los mismos recolectan, elaboran y distribuyen selectivamente la información necesaria para las operaciones de la empresa y apoyan a las actividades de gestión y control correspondientes, al menos parcialmente en el proceso de toma de decisiones necesarias para ejecutar las funciones comerciales de la empresa de acuerdo con la estrategia de negocio.

Según [22], el desempeño de los sistemas informáticos es importante ya que aporta significativamente, a la toma de decisiones dentro de un ámbito empresarial. El procesamiento de una gran cantidad de información suele ser un problema que los sistemas informáticos pueden resolver, por lo que se puede minimizar el tiempo invertido y el resultado es más rápido. Además, [20] señaló que los otros dos elementos básicos que forman un sistema de información junto con la información son los usuarios (gerentes, empleados y cualquier agente de la organización empresarial que usualmente utilizan información en sus lugares de trabajo), y equipos (TI, software, Hardware y tecnología de almacenamiento de información y telecomunicaciones).

1.2.2. SISTEMAS INFORMÁTICOS EN LA WEB

Los sistemas de escritorio pueden ocasionar inconvenientes a la hora de actualizar el software; además, el tiempo de respuesta del sistema será diferente, pero dependerá de las características del host, lo que impide la escalabilidad del sistema. Incluso a pesar de que el desarrollo de sistemas de escritorio en determinadas circunstancias sea la más conveniente, sin embargo, siempre dependerá de la funcionalidad requerida a implementar [23]. Según EUATM [24], aparte de los

enlaces de hipertexto, los documentos que también contienen elementos multimedia se denominan páginas web. Dado que la mayoría de los servidores contienen enlaces a otras páginas web que pueden estar ubicadas en el mismo servidor o en cualquier otro servidor de Internet, puede acceder a la Web a través de cualquier servicio para navegar por toda la red.

Las arquitecturas de aplicaciones destinadas al uso en la Web utilizan una estructura cliente-servidor, a través de la cual se pueden procesar las solicitudes de diferentes sitios administrados por los usuarios finales.

La mayoría de sistemas informáticos en la Web contiene una arquitectura compuesta principalmente por el software de la aplicación y un motor de base de datos.

1.2.3. BASE DE DATOS

Una base de datos es un conjunto de datos almacenados en una memoria externa, organizados por una estructura de datos. Una base de datos puede verse como un gran almacén de datos que solo se define y crea una vez, y es utilizado por diferentes usuarios al mismo tiempo. Según [25], antes de que exista la base de datos, el programa debe procesar los datos almacenados en archivos desconectados con información redundante. En la base de datos, todos los datos se integran con la menor cantidad de duplicados. De esta forma, la base de datos no pertenece a un solo departamento, sino que es compartida por toda la organización. Además, la base de datos no solo contiene datos organizados, sino que también almacena una descripción de esos datos. Esta descripción se denomina metadatos, se almacena en un diccionario o directorio de datos, y permite la denominada independencia lógica y física de los datos.

La gestión de las bases de datos se realiza a través de sistemas de gestión (denominados DBMS por sus siglas en inglés: Database Management Systems o Database Management Systems), actualmente digitales y automatizados, que permiten el almacenamiento ordenado y rápida recuperación de la información. En esta tecnología está el principio mismo de la informática.

Al crear una base de datos se pueden seguir diferentes modelos y paradigmas, cada modelo y paradigma tiene características, ventajas y dificultades, destacando su

estructura organizativa, estructura jerárquica, capacidad de transmisión o relación mutua, etc. Esto se denomina modelo de base de datos y permite diseñar e implementar algoritmos y otros mecanismos de gestión lógica de acuerdo con situaciones específicas. [26]

Existen diversos gestores de bases de datos, pero los más utilizados sin duda son MySQL y PostgreSQL.

El esquema gráfico de la base de datos y su interacción con la computadora o el sistema de dispositivo inteligente se puede demostrar en la Figura 1 a continuación.



Figura 1 Esquema de Base de Datos

Fuente: Tomado de Márquez [25]

1.2.4. POSTGRESQL

PostgreSQL es un poderoso sistema de base de datos relacional de objetos de código abierto que usa y expande el lenguaje SQL y combina muchas funciones que pueden almacenar y expandir de manera segura las cargas de trabajo de datos más complejas. El origen de PostgreSQL se remonta a 1986 y es parte del proyecto POSTGRES de la Universidad de California, ubicada en Berkeley, ha estado desarrollando activamente la plataforma central durante más de 30 años [27].

PostgreSQL ha establecido una buena reputación por su arquitectura confiable, integridad de datos, poderoso conjunto de características, escalabilidad y la dedicación de la comunidad de código abierto detrás del software para brindar constantemente soluciones innovadoras y de alto rendimiento. PostgreSQL puede

ejecutarse en todos los principales sistemas operativos, ha sido compatible con ACID desde 2001 y tiene potentes componentes adicionales, como la popular extensión de base de datos geoespacial PostGIS. [27]

Características

En la tabla 2 se indica algunas de las diversas características que se encuentran en PostgreSQL:

Tabla 2 Características de PostgreSQL

CARACTERÍSTICA	DESCRIPCIÓN
Tipos de datos	<ul style="list-style-type: none"> • Primitivas: entero, numérico, cadena, booleano • Estructurado: fecha / hora, matriz, rango, UUID • Documento: JSON / JSONB, XML, valor-clave (Hstore) • Geometría: Punto, Línea, Círculo, Polígono
Integridad de los datos	<ul style="list-style-type: none"> • ÚNICO, NO NULO • Llaves primarias • Llaves extranjeras • Restricciones de exclusión • Cerraduras explícitas, cerraduras consultivas
Concurrencia, rendimiento	<ul style="list-style-type: none"> • Indexación: B-tree, Multicolumn, Expresiones, Parcial • Indexación avanzada: Índices de cobertura, filtros Bloom • Partición de tablas • Recopilación de Just-in-time (JIT)

CARACTERÍSTICA	DESCRIPCIÓN
Confiabilidad, Recuperación de Desastres	<ul style="list-style-type: none"> • Registro de escritura anticipada (WAL) • Replicación: asíncrona, síncrona, lógica. • Espacios de tabla
Seguridad	<ul style="list-style-type: none"> • Autenticación • Sistema robusto de control de acceso • Seguridad de columnas y filas
Extensibilidad	<ul style="list-style-type: none"> • Funciones y procedimientos almacenados. • Lenguajes de procedimiento: PL / PGSQL, Perl, Python, etc. Contenedores de datos externos: conéctese a otras bases de datos o flujos con una interfaz SQL estándar.
Internacionalización, búsqueda de texto	<ul style="list-style-type: none"> • Soporte para conjuntos de caracteres internacionales. • Búsqueda de texto completo

Fuente: Tomado de PostgreSQL [27].

1.2.5. SOFTWARE

El software de computadora es un producto construido y mantenido por programadores profesionales durante mucho tiempo. [28]

Según [28], en su publicación afirma que: "Mucha gente relaciona el término software con programas de computadora. Sin embargo, es acertado lanzar una definición más amplia en la que el software no es solo un programa, sino que también incluye todos los documentos relevantes y necesarios para que estos programas funcionen correctamente. Los documentos y los datos se incluyen como parte del llamado software".

Según [29], mencionó las siguientes categorías de software diferentes:

- **Software de Base:** Está compuesto por componentes que actúan como enlaces entre programas escritos por programadores para realizar tareas específicas y el hardware informático.
- **Software de Aplicación:** Están destinados para el uso completo del usuario y, por lo tanto, se puede decir que es el tipo de software que permite realizar casi cualquier tarea. Se lo puede usar en cualquier instalación de computadora, sin importar para qué la vayamos a usar. Según [29], expresa que esta categoría de software de aplicación incluye todo el software diseñado para ayudar a los usuarios a realizar tareas. El software de aplicación puede considerarse como una herramienta que amplía las funciones humanas y puede realizar tareas que de otro modo serían difíciles o imposibles de realizar. Por tanto, la mayor parte del software entra en esta categoría.

1.2.6. CICLO DE VIDA DEL DESARROLLO DE SOFTWARE

Para [30] el modelo de ciclo de vida es un marco de referencia, que contiene los procesos, actividades y tareas relacionadas con el desarrollo, operación y mantenimiento del proceso del desarrollo de software, partiendo desde la definición y establecimiento de los requerimientos del software hasta el despliegue del sistema completo. El ciclo de vida del software, muchas veces depende de la metodología de desarrollo que se encuentre siendo utilizada, y dependiendo del modelo, es posible que cuando un ciclo termine, otro ciclo empiece inmediatamente, esto depende de la capacidad modular del sistema.

En el proceso de desarrollo de software es importante pasar por los siguientes pasos: [28]

- Análisis y definición del problema.
- Definición y especificación de requerimientos.
- Diseño de arquitectura del sistema.
- Codificación. Selección del lenguaje de programación. Escritura del algoritmo utilizando la sintaxis y estructura gramatical del lenguaje seleccionado.

- Pruebas Se realizan todas las pruebas necesarias para garantizar el funcionamiento del sistema, existen dos tipos de pruebas que se deben realizar, las internas a nivel de código o más conocido como “Debug”, y las funcionales, las cuales se las realiza una vez terminado un módulo o el sistema completo.
- Despliegue: En esta etapa involucra la liberación de un módulo o sistema completo, normalmente se lleva de la mano con la entrega al usuario final en el cual también es necesario realizar una demostración del funcionamiento, y de que el sistema cumple con los requerimientos solicitados.

1.2.7. LENGUAJE DE PROGRAMACIÓN

Se denomina lenguaje de programación a los conjuntos formales de instrucciones, que son utilizados por los desarrolladores (programadores) para crear funciones que generan acciones continuas de datos y algoritmos para buscar controlar el comportamiento físico y lógico de las máquinas (computadoras). De acuerdo con [31], es un lenguaje artificial diseñado para expresar cálculos que pueden ser realizados por máquinas como computadoras, pueden usarse para crear programas que controlen el comportamiento físico y lógico de las máquinas para expresar con precisión algoritmos o patrones de comunicación interpersonal.

1.2.8. PYTHON

Actualmente en la industria informática ha tenido un gran impacto el llamado software libre, es decir, se puede acceder al código fuente del programa, de manera que se puede utilizar, ejecutar, distribuir y modificar el programa de forma gratuita. Todo el software creado bajo este concepto se puede utilizar para cualquier propósito, se puede ejecutar en cualquier entorno, se puede distribuir a discreción del usuario o se puede modificar cuando sea necesario.

En la década de 1990, se produjeron una serie de eventos que marcaron algunas pautas para el futuro desarrollo del software libre. Por ejemplo, Linus Torvalds lanzó la primera versión del kernel de Linux en 1991 y Guido Van Rossum lanzó la primera versión del lenguaje de programación Python ese mismo año. [32]

Python es un lenguaje interpretado que se puede ejecutar en modos interactivos y de secuencias de comandos. Fue creado por Guido van Rossum a principios de la década de 1990 mientras trabajaba en Centrum Wiskunde e Informationatics en Ámsterdam. Debido a su simplicidad de codificación, se hizo cada vez más popular en comunidades orientadas al software libre, la ciencia y la educación. [32] [33]

El lenguaje de programación Python es poderoso y fácil de aprender. Tiene una estructura de datos avanzada y eficiente, y tiene un método de programación orientado a objetos simple y efectivo. La sintaxis y su tipado dinámico en combinación con sus propiedades de interpretación lo convierten en un lenguaje ideal para la creación de scripts y el desarrollo rápido de aplicaciones compatibles con diversas plataformas tales como Web, Escritorio o Móvil. [34]

Python combina características de diferentes paradigmas de programación incluyendo programación orientada a objetos. A continuación, se mencionan algunas de ellas: [33]

- **Sintaxis concisa y clara:** La sintaxis no sólo mejora la legibilidad del código, sino que también permite un código fácil de escribir para mejorar la eficiencia de la programación. [33]
- **Código de sangría:** A diferencia de otros lenguajes que usualmente usan marcas explícitas (como bloques de inicio-fin o llaves) para definir la estructura de un programa, Python solo usa el símbolo de dos puntos ":" y la sangría para definir bloques de código. Esto hace que la organización del código sea más concisa y tenga una jerarquía de bloques de código claramente definida. [33]
- **Simple, pero eficaz, el enfoque de programación orientada a objetos:** Los datos se pueden representar mediante objetos y las relaciones entre estos objetos. Las clases pueden definir nuevos objetos capturando información estructural compartida de nuevos objetos y modelando comportamientos relacionados. Python también implementa un mecanismo de herencia de clases, en el que una clase puede extender las funciones de otras clases heredando de una o más clases. Por tanto, desarrollar nuevas clases es una tarea sencilla en Python. [33]

- **Modularidad:** Los módulos son el aspecto central del lenguaje. Estos son fragmentos de código implementados previamente, que se pueden importar a otros programas. Después de la instalación, el uso del módulo es muy sencillo. Esto no solo mejora la simplicidad del código, sino que también mejora la eficiencia del desarrollo. [33]

En el ranking de los lenguajes de programación más populares en 2019, se puede ver en la Figura 2 a continuación, que Python se encuentra en la posición de liderazgo, lo cual se debe principalmente a los resultados obtenidos de la investigación utilizando este lenguaje de programación para el procesamiento de datos.

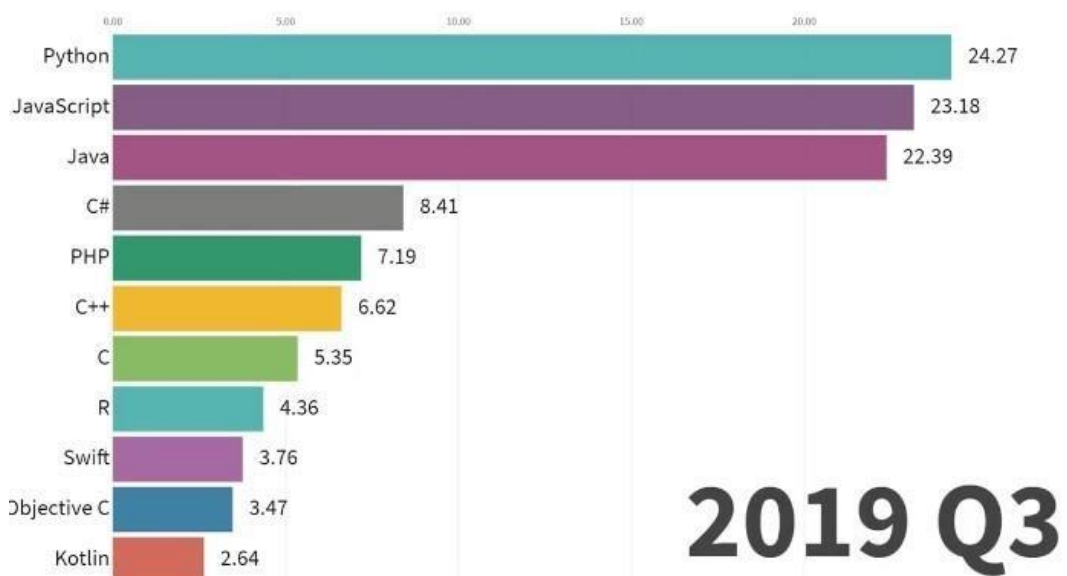


Figura 2 Lenguajes de programación más populares

Fuente: Tomado de Python Software Foundation [35]

1.2.9. LIBRERÍAS DE PYTHON

En el campo de la programación, una librería es un conjunto de archivos que implementan un conjunto de funciones que están codificadas en un lenguaje de programación específico y están listas para ser utilizadas de una manera que es fácil de usar al programar en ese lenguaje. [36]

Una de las particularidades de Python es que posee un amplio catálogo de librerías que permiten a los desarrolladores realizar grandes procesos en porciones simplificadas de código. Las librerías más utilizadas son:

Pandas: Según [37], Pandas es un paquete de software de Python que proporciona estructuras de datos rápidas, flexibles y expresivas diseñadas para manejar datos "relacionales" o "etiquetados" de manera fácil e intuitiva. Su propósito es analizar datos reales y datos reales en Python. Además, su objetivo a futuro es lograr ser una herramienta operativa de análisis de datos de código abierto que se pueda proporcionar en cualquier idioma. Pandas es adecuado para diferentes tipos de datos: [37]

- Datos tabulares con columnas de tipo heterogéneo, como en una tabla de SQL o una hoja de cálculo de Excel. [37]
- Datos de series de tiempo ordenados y desordenados (no necesariamente de frecuencia fija). [37]
- Datos matriciales arbitrarios (tipificados homogéneamente o heterogéneos) con etiquetas de fila y columna. [37]
- Cualquier otra forma de conjuntos de datos observacionales / estadísticos. Los datos realmente no necesitan ser etiquetados en absoluto para ser colocados en una estructura de datos pandas. [37]

Gracias a Pandas, Python ha mejorado en los siguientes aspectos: [37]

- Alineación automática y explícita de datos: los objetos pueden alinearse explícitamente a un conjunto de etiquetas, o el usuario puede simplemente ignorar las etiquetas y dejar que Series, DataFrame, etc., alinee automáticamente los datos en los cálculos. [37]
- Potente y flexible grupo por funcionalidad para realizar operaciones de combinación de aplicación dividida en conjuntos de datos, tanto para agregar como para transformar datos. [37]
- Facilita la conversión de datos irregulares, indexados de manera diferente en otras estructuras de datos de Python y NumPy a Objetos DataFrame. [37]
- Rebanado inteligente basado en etiquetas, indexación elegante y subconjunto de grandes conjuntos de datos. [37]
- Combinación intuitiva y unión de conjuntos de datos. [37]

- Etiquetado jerárquico de ejes (posible tener múltiples etiquetas por tic). [37]

Math: El módulo Math agrega las funciones trigonométricas seno, coseno y tangente que se representan, respectivamente mediante sin, cos y tan. Por defecto, esas funciones asumen que los ángulos se miden en radianes. Por su parte, el logaritmo natural (o neperiano) de base e en Python se denomina como *log* y la exponencial (para calcular e elevado a un número) se denomina como *exp*. Pero, además de funciones, a menudo los módulos de Python contienen otro tipo de objetos. Por ejemplo, el módulo Math contiene valores aproximados de las constantes matemáticas π y e , entre otras. [38]

NumPy: NumPy (abreviatura de Digital Python) es el módulo básico para la computación científica que usa Python. Con él se pueden utilizar herramientas de cálculo para procesar estructuras con gran cantidad de datos, con el objetivo de obtener un buen rendimiento a la hora de procesarlos. Este módulo incorpora un nuevo tipo de dato, el array, que es similar a una lista, pero es más eficiente computacionalmente. También cuenta con una gran cantidad de métodos que permiten manipular los elementos del arreglo de manera no secuencial, lo que se denomina vectorización y proporciona un alto rendimiento. [39]

1.2.10. FRAMEWORK DJANGO

Además de las librerías, Python ofrece la posibilidad de trabajar con Framework para el desarrollo de aplicaciones tanto de escritorio como Web, una de las más conocidas es Django.

Django es un marco web avanzado de Python que fomenta el desarrollo rápido y el diseño simple y práctico. Creado por desarrolladores experimentados, puede resolver la mayoría de los problemas del desarrollo web, por lo que puede centrarse en escribir aplicaciones sin gastar mucho esfuerzo. Es gratuito y de código abierto. [40]

Este es un Framework de desarrollo web destinado para el lenguaje de programación Python, el mismo permite la construcción de aplicaciones web de una manera más fácil y con código más optimizado. Fue desarrollado originalmente para administrar las aplicaciones web de las páginas orientadas a noticias de World

Online, y luego fue lanzado bajo la licencia BSD [41]. Django contiene un conjunto de componentes que hacen que el desarrollo de sitios web sea más fácil y rápido. Según [42], afirma que Django se inventó para lograr estos nuevos objetivos. Django permite crear sitios web en profundidad, dinámicos e interesantes en muy poco tiempo.

Características

- **Rápido:** Django está diseñado para ayudar a los desarrolladores a mover aplicaciones desde el concepto hasta su finalización lo más rápido posible. [42]
- **Seguridad:** Django se toma la seguridad en serio y ayuda a los desarrolladores a evitar muchos errores de seguridad comunes. [42]
- **Altamente escalable:** algunos de los sitios más activos de la Web aprovechan la capacidad de Django para escalar de forma rápida y flexible. [42]

Arquitectura

El Framework Django maneja una arquitectura con tres capas, las cuales son: [43]

- **Models:** Define los datos almacenados, en forma de clase Python, cada tipo de dato que se debe almacenar se encuentra en una variable con ciertos parámetros, y también tiene métodos. Todos estos pueden indicar y controlar el comportamiento de los datos. [43]
- **View:** Se expresa como una función en Python y su propósito es determinar los datos que se mostrarán. El ORM de Django permite escribir código Python en lugar de SQL para ejecutar las consultas requeridas por la vista. También maneja tareas conocidas como enviar correos electrónicos, autenticarse a través de servicios externos y validar datos a través de formularios. [43]
- **Template:** Es una página HTML (incluido XML, CSS, JavaScript, etc.) con algún marcado adicional de Django que se encarga de procesar el estilo de presentación de los datos que observará el usuario final. [43]

1.2.11. MINERÍA DE DATOS

El continuo desarrollo de la informática ha hecho que la información digital sea fácil de procesar, transmitir y almacenar. Gracias a este avance importante en las tecnologías relacionadas con los sistemas de información, se sigue recopilando y almacenando una gran cantidad de información en la base de datos.

El nacimiento de la minería de datos se deriva de las ventajas de dos cosas: grandes cantidades de datos almacenados en campos como el comercio, la banca o la atención médica, y la capacidad de las nuevas computadoras para realizar operaciones de análisis de estos datos. [44] La minería de datos es un conjunto de tecnologías que permiten la exploración automática o semiautomática de grandes bases de datos para encontrar patrones repetidos que expliquen el comportamiento de estos datos. [45]

La minería de datos es un proceso que permite analizar grandes conjuntos de datos para detectar patrones que expliquen su comportamiento. Al aplicar tecnología basada en los resultados que se desea obtener, todos estos procesos se realizarán de la forma más automatizada.

Las técnicas de minería de datos son el resultado de la investigación y el desarrollo de productos a largo plazo. Este desarrollo comenzó con el almacenamiento inicial de datos comerciales en computadoras, y ha seguido evolucionando a medida que mejora el acceso a los datos. Recientemente, han surgido tecnologías que permiten a los usuarios navegar por los datos en tiempo real. La minería de datos hace que este proceso evolutivo vaya más allá del acceso retrospectivo y la navegación de datos, y avance hacia la dirección de entregar información esperada y proactiva. La minería de datos está lista para ser aplicada en las comunidades empresariales porque ha sido respaldada por: [46]

- Recolección masiva de datos.
- Potentes computadoras con multiprocesadores.
- Algoritmos de Data Mining.

1.2.12. MINERÍA DE TEXTO

Uno de los puntos importantes dentro de la minería de datos, es el análisis dedicado a la extracción de patrones dentro de textos previamente descritos y almacenados, a esta práctica se la conoce como minería de texto.

Se refiere a la inspección de la colección de documentos y al descubrimiento de información que no está contenida en un solo documento de la colección, es decir, intenta obtener información en lugar de partir de algo. Dado que el 80% de la información de la empresa se almacena en forma de documentos, se admitirán tecnologías como la clasificación de texto, el procesamiento del lenguaje natural, la extracción y recuperación de información y el aprendizaje automático. [47]

Al ser la minería de textos, parte de la minería de datos, emplea técnicas que a menudo son utilizadas para formar un método conceptual, las mismas que generalmente se implementa a través de varios algoritmos. [48] Se pueden clasificar según su utilidad, como se muestra a continuación:

- **Las técnicas de predicción:** Permiten obtener predicciones de comportamiento futuro a partir de los datos recopilados. [49] Estas tecnologías son útiles en aplicaciones como las predicciones meteorológicas predictivas o la toma de decisiones del cliente en determinadas situaciones.
- Según [50], la predicción es completamente útil en el proceso de toma de decisiones. Por lo tanto, algunos trabajos basados en modelos dinámicos autorregresivos de series de tiempo y modelos de caminata aleatoria han demostrado empíricamente que predecir rezagos basados en un período de tiempo puede ser muy complicado.
- **Las técnicas de Clustering:** También conocido como análisis de datos conglomerados o agrupados. Esta técnica permite analizar y verificar datos sin etiquetar, y formar grupos en función de su similitud [51]. El principal objetivo de esta técnica es dividir conjunto de datos en dos o más grupos en función de sus características comunes. La similitud se puede medir mediante la función de distancia. Los objetos se agrupan de acuerdo a todas

las variables, por lo tanto, las variables irrelevantes generarán ruido en los resultados obtenidos. [51]

- **Las técnicas de reglas de asociación:** Permiten el establecimiento de posibles relaciones o correlaciones entre diferentes acciones o eventos aparentemente independientes; pueden identificar cómo la ocurrencia de un evento o acción induce o produce la aparición de otras cosas. [48]
- **Las técnicas de clasificación:** Definen una serie de clases que pueden agrupar diferentes situaciones. En este grupo se encuentran las técnicas de árboles de decisión y las reglas de inducción. [49]

El objetivo principal de todas las técnicas mencionadas es analizar una amplia gama de datos para obtener información que ayude a explicar el comportamiento del objeto de investigación y ayude a tomar decisiones. Estas técnicas se aplican a través de algoritmos probados e implementados en soluciones de minería de datos.

1.2.13. ALGORITMO

En el proceso de resolución de una problemática, utilizando como herramienta tecnológica una computadora, es preciso fijar una serie de pasos que permitan resolver el problema. A esta secuencia de pasos se le denomina algoritmo, el mismo que debe tener la posibilidad de ser fácilmente transcrito a un determinado lenguaje de programación. [52] El algoritmo representa un conjunto bien definido de instrucciones que la computadora debe ejecutar para obtener resultados predecibles. Para que la computadora pueda interpretar estas instrucciones, deben estar escritas en un lenguaje de programación.

Para Pinales y Velázquez [52], además de las características de fácil transcripción, el algoritmo también debe ser:

- **Preciso:** Debe indicar el orden de ejecución de cada paso que condujo a la resolución del problema.
- **Definido:** Esto significa que el resultado nunca puede cambiar en las mismas condiciones que el problema, siempre debe ser el mismo.

- **Limitado:** No debe meterse en la repetición del proceso innecesariamente; debe terminar en algún momento.

Razón por la cual, un algoritmo contempla una serie de operaciones detalladas y claras que se deben realizar paso a paso, conduciendo a la solución del problema, y expresadas en herramientas o técnicas. [53] Alternativamente, es una descripción de un método para resolver el problema planteado de una manera apropiada y universal. Además, se debe tener en que para convertir el algoritmo en un programa de computadora debe considerar las siguientes partes:

- Los resultados se obtendrán mediante el procesamiento de datos.
- Una descripción de las medidas que se deben tomar para procesar los datos.
- Una descripción de los datos a tratar.

1.2.14. ALGORITMOS EN MINERÍA DE DATOS

Según [54], los algoritmos de minería de datos generalmente realizan la tarea de predecir información desconocida que puede estar contenida en los datos, así como la tarea de describir patrones de comportamiento de los datos.

Los algoritmos son esenciales en la minería y análisis de datos porque incluyen métodos que permiten un análisis de datos rápido y automático, estos procesos ayudan a obtener patrones y modelos del conjunto de información evaluado. Los algoritmos más utilizados para este tipo de análisis son los algoritmos de clasificación.

Los algoritmos de clasificación son algoritmos que intentan clasificar una serie de ejemplos o instancias de cierta información que representa un problema en diferentes categorías. En el campo del aprendizaje automático, estos tipos de algoritmos tienen como principal objetivo el aprender a determinar a qué categoría pertenece un nuevo ejemplo sin etiquetar. Hay dos tipos de clasificación: [55]

- **Supervisados:** En este tipo de clasificación, tenemos un conjunto de datos que ya conocemos su clase, llamados ejemplos de entrenamiento o conjuntos de entrenamiento.

- **No Supervisado:** Los datos no tienen etiquetas y estas se clasifican según su estructura interna considerando patrones comunes en sus atributos y/o características.

1.2.15. AUTOAPRENDIZAJE

Gracias a la aplicación de algoritmos de minería de datos, los sistemas de información tienen la capacidad de desarrollar conocimiento a través del autoaprendizaje.

Las tecnologías de la información y la comunicación (denominadas de ahora en adelante TIC) respaldan claramente la noción de que lo que realmente importa es la consecución del objetivo y el mejor nivel de calidad, no la existencia real en un momento y lugar específicos. Al mismo tiempo, permite la generación de espacios virtuales compartidos (para relaciones, trabajo de investigación). [56]

Este proceso se refiere al autoaprendizaje de eventos anteriores, marcando así el punto de partida de nuevos conocimientos. En el campo de la informática, el autoaprendizaje toma como ejemplo la capacidad de las máquinas (computadoras) para recibir información y usarla para eventos futuros, proporcionando así soluciones rápidas y efectivas a problemas relacionados.

1.2.16. TOMA DE DECISIONES

Gracias a que las máquinas pueden desarrollar su habilidad para aprender, también serán capaces de tomar decisiones en determinadas circunstancias.

La toma de decisiones es una actividad diaria que todas las personas experimentan, y es muy fundamental para el desenvolvimiento de cualquier actividad. Sin embargo, tomar la decisión correcta comienza con un proceso de razonamiento continuo y concentrado, que puede incluir varias disciplinas como la filosofía del conocimiento, la filosofía de la ciencia, la filosofía de la lógica y, lo más importante, la creatividad. El gerente debe tomar muchas decisiones todos los días. Algunas de estas son decisiones de rutina, mientras que otras tienen un impacto significativo en las operaciones de la empresa donde trabaja. [57]

Las decisiones sobre las TIC tienen mucho que ver con los datos existentes y sus métodos de procesamiento. Es importante saber que, para tomar mejores decisiones,

los datos deben estar correctamente clasificados y estructurados, por eso se utilizan algoritmos para aprender y mostrar los resultados de forma interactiva, de manera que los usuarios entiendan claramente la respuesta que brinda el sistema.

1.2.17. REDES NEURONALES

Es un sistema de procesadores en paralelo interconectados en forma de gráfico dirigido. De manera esquemática, cada elemento de procesamiento (neurona) de la red se representa como un nodo. Estas conexiones establecen una estructura jerárquica, tratando de imitar la filosofía del cerebro, buscando nuevos modelos de procesamiento para resolver problemas específicos en el mundo real. [58] Una definición simplificada de un gráfico topológico puede ser que, con respecto a la correspondencia topológica, las unidades que están físicamente próximas entre sí responderán a categorías de vectores de entrada que están igualmente próximas entre sí. Muchos vectores de entrada dimensionales se representan en forma de gráficos bidimensionales para mantener el orden natural de los vectores de entrada. [58]

La figura 3 muestra una comparación gráfica entre la red neuronal del cerebro y la red neuronal artificial creada para la investigación y análisis de datos, menciona que la red neuronal se refiere al arte de imitar el cerebro.

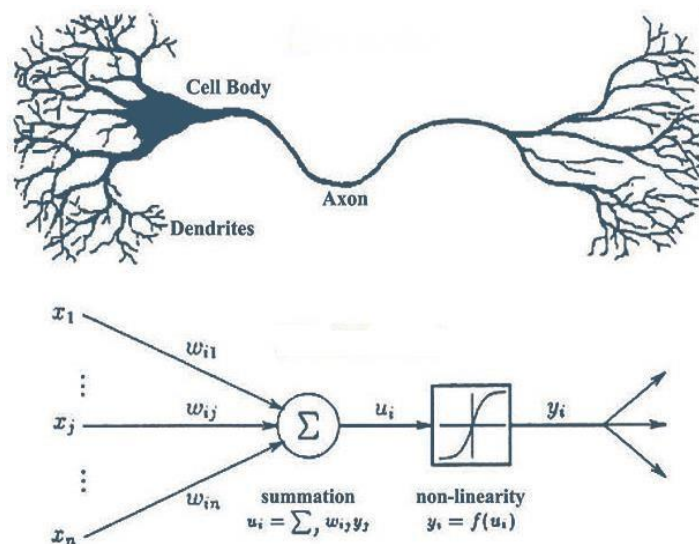


Figura 3 Comparación de Neurona cerebral y Redes neuronales artificiales

Fuente: Tomado de Sotolongo [58]

1.2.18. REDES NEURONALES ARTIFICIALES

La Red Neuronal Artificial (ANN) o sistema de conexión es un sistema de procesamiento de información cuya estructura y funcionamiento están inspirados en las redes neuronales biológicas. Consisten en un conjunto de elementos de procesamiento simples llamados nodos o neuronas, que están conectados entre sí mediante conexiones con valores modificables llamados "pesos". [59]

Las actividades realizadas por la unidad de procesamiento o neurona artificial en tal sistema son simples. Por lo general, consiste en sumar los valores de entrada que recibe de otras unidades conectadas a él, comparar esta cantidad con un umbral, y si es igual o superior al umbral, enviar una activación o salida al conectado. [59] Tanto la entrada recibida por el dispositivo como la salida enviada por el dispositivo dependen del peso o la fuerza de la conexión a través de la cual se realizan estas operaciones. [59]

En la Figura 4 se aprecia una representación realista por científicos que muestra una red neuronal realizando procesamiento e interpretación de datos.

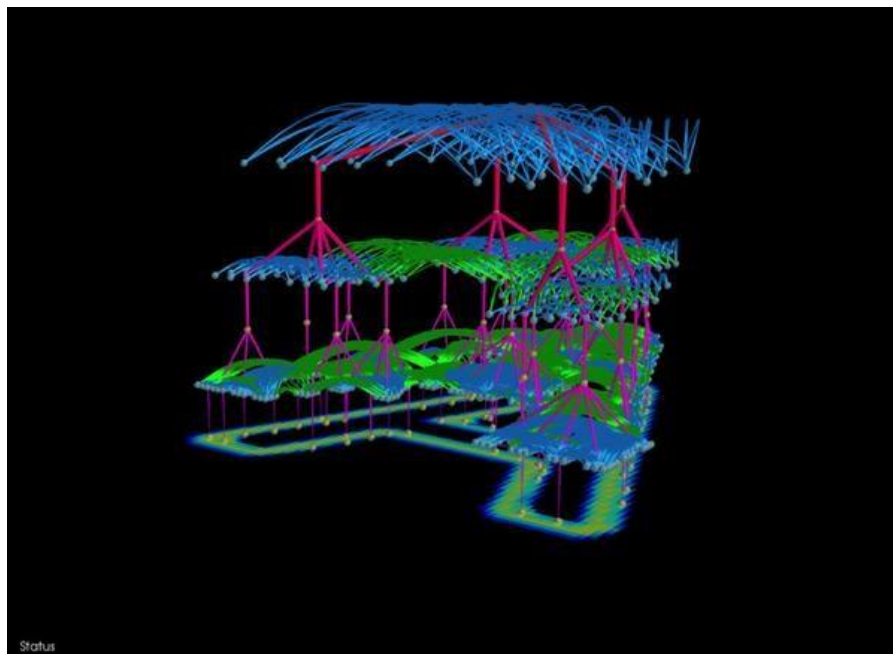


Figura 4 Red neuronal armada por científicos

Fuente: Tomado de Montaña [59]

1.2.19. APRENDIZAJE PROFUNDO

El aprendizaje profundo o Deep Learning, es una de las aplicaciones más poderosas y de mayor crecimiento de la inteligencia artificial. Es un subcampo del aprendizaje automático que se utiliza para resolver problemas muy complejos que suelen involucrar grandes cantidades de datos. [60]

El aprendizaje profundo se realiza mediante el uso de redes neuronales, que se organizan jerárquicamente para identificar relaciones y patrones complejos en los datos. Su aplicación requiere mucha información y potentes capacidades de procesamiento. Actualmente, se utiliza en reconocimiento de voz, procesamiento de lenguaje natural, visión por computadora y reconocimiento de vehículos en sistemas de asistencia al conductor. [60]

Podemos ver un ejemplo obvio en las traducciones realizadas en Facebook, que recientemente demostró que es capaz de alrededor de 4.500 millones de traducciones al día debido al aprendizaje profundo. Suelen ser segmentos de texto breves, como actualizaciones de estado publicadas por los usuarios en su perfil. Sin el aprendizaje profundo, sería muy costoso y requeriría una gran cantidad de personas para brindar el mismo servicio. [60]

1.2.20. TENSORFLOW

TensorFlow es un sistema de aprendizaje automático que funciona en entornos grandes y heterogéneos. TensorFlow usa diagramas de flujo de datos para representar cálculos, estados compartidos y operaciones para cambiar ese estado. Mapea los nodos (TPU) del gráfico de flujo de datos a través de muchas computadoras en el clúster y a través de múltiples dispositivos informáticos, incluidas CPU de múltiples núcleos, GPU de uso general y ASIC de diseño personalizado, llamadas unidades de procesamiento Tensores. [61] TensorFlow permite a los desarrolladores probar nuevos algoritmos de optimización y entrenamiento. TensorFlow admite varias aplicaciones, con un enfoque en el entrenamiento y la inferencia en redes neuronales profundas.

El aprendizaje profundo es un subcampo del aprendizaje automático que utiliza redes neuronales artificiales para estimular la estructura y función del cerebro humano. [62] Aunque este es un método muy nuevo, se ha vuelto muy popular

recientemente. Entre las muchas aplicaciones en las que el aprendizaje automático ha tenido éxito a cierta velocidad, el aprendizaje profundo ha logrado un mayor éxito. En particular, se prefiere en la clasificación de grandes conjuntos de datos porque puede proporcionar resultados rápidos y efectivos.

Los cálculos representados por TensorFlow se pueden realizar en una variedad de sistemas heterogéneos con poca o ninguna modificación, desde dispositivos móviles (como teléfonos móviles y tabletas) hasta sistemas distribuidos a gran escala con cientos de máquinas y varios dispositivos informáticos. [62]

1.3. FUNDAMENTACIÓN DEL ESTADO DEL ARTE

En los últimos años la investigación en temáticas de Inteligencia Artificial y Redes Neuronales Artificiales ha ido creciendo considerablemente, muestra de ellos son los sistemas que tienen la capacidad de poder realizar predicciones a partir de modelos de datos. Afortunadamente TensorFlow aumentó aún más el campo de investigación y facilitó la comprensión de cómo los algoritmos trabajan en un determinado entrenamiento de datos.

Existen algunos antecedentes de investigaciones realizadas en la plataforma Ecuciencia. Ecuciencia nace como una necesidad de gestión de documentos de investigación digitales en el año 2017, el mismo surgió gracias al aporte realizado por [63], ya que, en sus Tesis Doctoral, estableció métodos claves para aprovechar al máximo los datos mediante la utilización de algoritmos de minería de datos.

Posteriormente en el 2018, se crea el proyecto REDEC el mismo que según [64] tiene como objetivo el análisis de la producción científica de los investigadores de las Universidades perteneciente a la Zona 3; a partir del comportamiento de los diferentes indicadores de Ciencia, Técnica e Innovación de las instituciones. [64]

REDEC comenzó específicamente en la Universidad Técnica de Cotopaxi, formando diversos grupos de trabajo, destinados a una investigación específica. A partir de estas investigaciones realizadas surgieron temas de Tesis de los alumnos de la carrera de Ingeniería en Informática y Sistemas Computacionales, que más adelante por la reforma de malla curricular pasaría a llamarse Sistemas de Información.

Publicaciones como las de [65] y [66] fueron las primeras tesis que nacieron del proyecto REDEC, en ambas hablan sobre la importancia de los algoritmos en el tratamiento de datos para la base científica de Ecuciencia.

En [65] se muestra una investigación detallada sobre la minería de texto, los mismos que permiten obtener información de escritos, discriminando palabras comunes o conocidos como “stopwords”. En su investigación [65] propone “recolectar una cantidad determinada de información considerable y posteriormente implementar un algoritmo clasificador automático de textos que permite estructurar datos relevantes a un dominio específico (clase o categorías), siendo en este caso los documentos científicos generados por los docentes investigadores de la Universidad Técnica de Cotopaxi”. En su investigación se puede denotar el uso de la librería NLTK, la cual es muy usada para el tratamiento de datos a través de la minería de texto.

Por otra parte [66] trabajaba en paralelo estableciendo un método para determinar la similitud y distancia entre investigadores, el mismo que se basó en algoritmos de clasificación convencionales para lograrlo. En su investigación detalla la importancia del uso de librerías como SKLearn, la cual contienen diferentes algoritmos dedicados a la clasificación no supervisada, entre los conocidos KMeans, Spectral, MeanShift, los cuales se basan en modelos matemáticos como el cálculo de distancias euclidianas, matrices de pesos, etc.

Estos dos trabajos precursores dieron apertura a futuras investigaciones teniendo como punto de referencia el uso de algoritmos de inteligencia artificial en el tratamiento de datos científicos de la Universidad Técnica de Cotopaxi.

Tiempo más tarde, [67] realiza una investigación centrada en el mapeo autoorganizado (SOM), y a la par propone una metodología para su análisis. Con el resultado de esta investigación el autor demuestra que el método de trabajo alineado a SOM brindará gran ayuda en investigaciones futuras y mejorará el sistema ECUCIENCIA en cuanto a toma de decisiones automáticas y aprendizaje no supervisado se refiere. [67] El uso de los conocidos mapas de calor para la representación de los resultados es el punto clave a destacar de esta investigación.

Por otro lado, [68] también realiza su aporte científico a Ecuciencia, ya que en su tesis de maestría propone un método de análisis de redes sociales para identificar relaciones y colaboraciones científicas entre investigadores. [68] Dicha investigación se centra específicamente en el análisis de redes sociales utilizando una estructura metodológica propuesta por Wasserman y Faust. Además [68] manifiesta que “El propósito de estudiar las redes sociales es comprender el comportamiento de los implicados mediante métricas o similitudes para poder obtener un análisis de datos complejo y puedan ser de forma ordenada mediante la generación de gráficos visuales dónde están estos datos analizados y clasificados”. [68]

Derivado de lo anterior, se puede observar que la plataforma científica Ecuciencia, es relativamente joven sin embargo tiene amplias fuentes de información debido a las investigaciones realizadas en la misma. El uso de algoritmos en Ecuciencia, ha sido la clave para su nivel de aprendizaje automático, sin embargo, aún existen puntos importantes que son necesarios fortalecer.

1.4. CONCLUSIONES CAPÍTULO I

- La cienciaometría es un campo de estudio que ha llegado a diversos países a nivel Latinoamericano. Su principal objetivo es llevar un estándar de evaluación para asegurar la calidad de publicaciones científicas. Dichos estándares son muy aplicados en sistemas de información que gestionan datos de producción científica alrededor del mundo.
- Existe un nivel alto de compatibilidad entre el desarrollo de software y la inteligencia artificial, ya que, gracias al uso de algoritmos y redes neuronales artificiales, y la intervención de frameworks como Django, se puede establecer un sistema de información completo para análisis de datos.
- Existen diversas publicaciones científicas centradas en el funcionamiento de la plataforma científica Ecuciencia, lo cual favorece al crecimiento de nuevas investigaciones derivadas de temáticas sugeridas por los investigadores del proyecto REDEC.

CAPÍTULO II. PROPUESTA

2.1. DIAGNÓSTICO

Desde hace algún tiempo en la Universidad Técnica Cotopaxi se viene desarrollando un sistema en plataforma web denominado Ecuciencia, que tiene como objetivo organizar la información de los docentes investigadores y estudiar la interrelación existente entre ellos a través de sus publicaciones de artículos, libros y trabajos científicos. Debido a la existencia de una gran cantidad de información, la automatización de procesos basada en la clasificación de campos de conocimiento en datos de producción científica se ha vuelto muy complicada, esto se complica aún más debido el gran número de docentes investigadores, que actualmente forman parte de la Universidad Tecnológica Cotopaxi.

Actualmente toda la información almacenada en dicha plataforma, no es utilizada de la manera más óptima, y en algunos casos, ni siquiera se emplea algún tipo de beneficio en su utilización, lo cual lo convierte en datos almacenados sin ningún objeto y por ende se puede catalogar como datos basuras al solo ocupar espacio de almacenamiento innecesario. También cabe recalcar que no existe ningún método que apoye a los usuarios sobre la clasificación de su línea de investigación correspondiente, lo cual deriva en que el usuario tenga la libertad de escoger una opción fuera de su contexto y que dicha información almacenada cada vez vaya aumentando y generando data inconsistente.

La propuesta para solventar los problemas expresados es la construcción de un módulo que permita realizar un análisis de texto de investigaciones para así poder tener un antecedente y clasificar los datos según su línea y sublínea de investigación. Esto se lo podrá realizar mediante un proceso de minería de texto, el cual se encargará de recopilar la información más relevante de todas las investigaciones, realizando un barrido de palabras comunes a través de algoritmos

de análisis de texto, y posteriormente mediante el uso de Redes Neuronales Artificiales proporcionadas por la librería TensorFlow para el análisis y clasificación de texto, y así lograr una predicción en base a la evaluación de publicaciones registradas de manera más acertada.

Además, se pretende que el módulo a desarrollarse tenga la capacidad de realizar un autoaprendizaje para que sus modelos empleados siempre tengan un conocimiento actualizado, lo cual generará que, al momento de hacer una predicción, lo realice en el menor tiempo posible.

2.2. MÉTODOS ESPECÍFICOS DE LA INVESTIGACIÓN

Para la solución de la presente propuesta se ha decidido emplear dos metodologías de desarrollo, la primera, la metodología KDD, la cual es la más utilizada dentro del campo de Inteligencia Artificial y Minería de Datos, y la segunda, la metodología del modelo Iterativo Incremental, debido a que es uno de los métodos de desarrollo de software favoritos por su eficiencia y eficacia en el desarrollo ágil de sistemas de información.

2.2.1. METODOLOGÍA KDD

Knowledge Discovery in Database (KDD) es básicamente un proceso automático en el que se combinan el descubrimiento y el análisis de datos. Este proceso implica extraer patrones en forma de reglas o funciones de los datos para el análisis del usuario. [69] En el desarrollo de la presente propuesta tecnológica se utiliza el método KDD, porque como se muestra en la Figura 5, las distintas etapas que lo constituyen hacen que el desarrollo sea iterativo e interactivo. Es iterativo, porque dependiendo del resultado obtenido en cada etapa, puede volver al paso anterior, y esto también se debe a que generalmente se necesitan varias iteraciones para extraer conocimiento de alta calidad. También se puede catalogar como interactivo porque involucra a los usuarios en muchas decisiones.

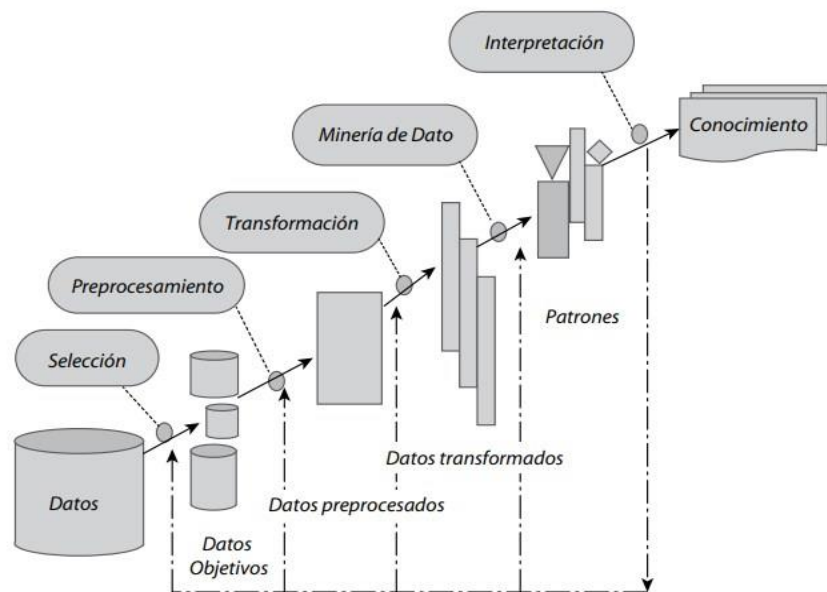


Figura 5 Etapas de la Metodología KDD

Fuente: Tomado de Pereira, Arteaga, Caicedo, Hidalgo y Pérez [69]

Etapas de la Metodología KDD

a) Selección:

En esta etapa, se crea el conjunto de datos de destino, se selecciona todo el conjunto de datos o una muestra representativa del conjunto de datos y luego se realiza el proceso de descubrimiento en el conjunto de datos. La elección de datos varía según los objetivos del negocio. [69]

El primer paso para extraer conocimiento de los datos es identificar y recopilar los datos que se utilizarán. Para ello se identificará la base de datos integrada al sistema Ecuciencia, así como los datos y usos que conforman la base de datos.

b) Preprocesamiento/limpieza.

En la fase de preprocesamiento/limpieza, se analizará la calidad de los datos, se aplicarán operaciones básicas, como eliminar datos ruidosos, elegir estrategias para lidiar con datos desconocidos, datos vacíos, datos duplicados y técnicas estadísticas alternativas para su reemplazo. [69]

En esta etapa se analizarán las tablas que integran la base de datos del sistema Ecuciencia y se seleccionarán aquellas que se consideren necesarias para la aplicación del algoritmo.

c) Transformación/reducción.

En la etapa de transformación/reducción de datos, se buscan características útiles para representar los datos de acuerdo con los objetivos del proceso. [69] Los métodos de transformación o reducción de dimensionalidad se utilizan para reducir el número efectivo de variables consideradas o para encontrar representaciones invariantes de datos. [69]

En esta etapa se realiza la selección de los atributos requeridos, para utilizarlos en los algoritmos. Es importante aclarar que la selección de dichos atributos se los realizará en base a los indicadores cuantitativos.

d) Minería de datos (data mining).

En la fase de minería de datos se aplicarán modelos, tareas, técnicas y algoritmos seleccionados para obtener reglas y patrones. [70] En esta etapa, se seleccionan y aplican técnicas de minería de datos adecuadas, que puedan cumplir con los objetivos propuestos, para lo cual se recopila teóricamente la información necesaria.

Utilizando la tecnología adecuada, se procederá con el análisis manipulando los algoritmos de aprendizaje profundo para obtener patrones que permita dar una clasificación de líneas y sublíneas de investigación en base a los atributos previamente seleccionados. Es importante recalcar que para realizar este proceso se utiliza el lenguaje de programación Python.

e) Interpretación/evaluación

En esta etapa se explicarán los patrones descubiertos y se podrá devolver la etapa anterior para iteraciones posteriores, pudiendo también incluir la visualización de los patrones extraídos. [69] Por otro lado, se consolida el conocimiento descubierto para fusionarlo en otro sistema para operaciones posteriores, o simplemente registrarlo e informar a las partes relevantes; también puede verificar y resolver conflictos potenciales del conocimiento previamente descubierto. [69]

En esta etapa se evaluarán los resultados obtenidos mediante la aplicación del algoritmo y se verificará si cumple con los objetivos de predicción y

clasificación de líneas y sublíneas de investigación, y luego se incluirá en el sistema Ecuciencia.

2.2.2. METODOLOGÍA MODELO ITERATIVO INCREMENTAL

En los últimos años, el desarrollo de software ha ido creciendo hasta convertirse en uno de los campos más competitivos, es por esta razón que las metodologías de desarrollo se encuentran en constante evolución teniendo como objetivo mejorar y optimizar el producto resultante para una mejor interacción con el usuario. Debido a que el ciclo de vida del software es variable es importante identificar la metodología idónea. Según [71], el modelo ágil enfatiza que la estimación del estado del usuario y sus respectivas tareas debe ser realizada por el equipo de desarrollo.

En el proceso de desarrollo de software es importante tener una retroalimentación constante por cada segmento de actividades que se realice, es por eso que la metodología basada en el modelo iterativo incremental es muy acertada, ya que en este se utiliza un conjunto de actividades a realizarse en cada una de sus pequeñas iteraciones, lo cual permite la construcción del software de una forma modulada y con poco lapso de tiempo entre un entregable y otro. En la figura 6 se puede observar el esquema del modelo iterativo incremental y sus etapas:

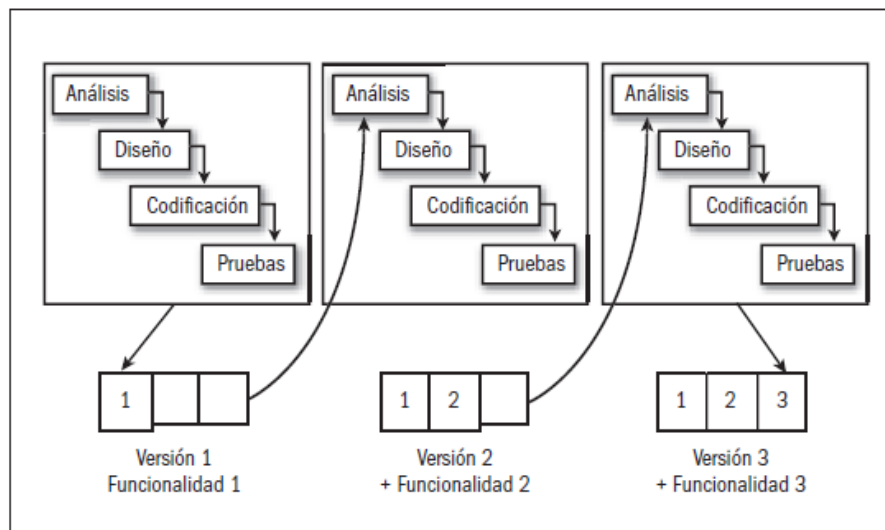


Figura 6 Etapas de la Metodología del modelo Iterativo Incremental

Fuente: Tomado de Ramos, Noriega, Laínez y Durango [72]

Etapas de la Metodología del Modelo Iterativo Incremental

a) Análisis

La etapa de análisis en el sistema de información es el primer paso en el proceso de desarrollo del sistema. Consiste en una serie de documentos y diagramas que representan los requisitos del software, definiendo así las funciones y comportamientos del software, simplificando así el proceso de comprensión y proporcionando estándares desde diferentes ángulos, reduciendo así la posibilidad de errores.

En el proceso de análisis, es esencial que, al recopilar los requisitos funcionales y no funcionales, los desarrolladores de software comprendan completamente la naturaleza del programa que se debe construir para desarrollar la aplicación, la funcionalidad, el comportamiento, el rendimiento y la interconectividad requeridos. [73]

Mediante la aplicación de una entrevista, apoyados en un cuestionario previamente estructurado, se pretende recabar la mayor cantidad de información para fortalecer el proceso de levantamiento de requerimientos.

Una vez se obtenga un panorama más amplio en cuanto a las necesidades del sistema, se traducirá a diagramas propios del método de desarrollo de software, y como se sugiere [67] los mismos serán catalogados como:

Elementos Basados en Escenarios: Para la mejor interacción con el sistema y el usuario: [67]

- Casos de Uso.

Elementos Orientados al Flujo: Tiene una visión del sistema del tipo entrada proceso-salida: [67]

- Diagramas de Actividades.

Elementos Basados en Clases: Contemplan los tipos de diagramas que representan las entidades involucradas dentro de un sistema, y cuáles son sus atributos, para este caso se escogerá: [67]

- Diagrama de Entidad - Relación.

Elementos de Comportamiento: Indica la forma en que el software responderá a los eventos o estímulos externos: [67]

- Diagrama de secuencia.

Para la construcción de los diagramas establecidos en la etapa de análisis, se utilizará la herramienta Draw IO, la misma que ofrece las facilidades del modelado de cualquier tipo de diagramas (incluidos UML) y tiene una licencia de libre acceso y uso.

b) Diseño

La siguiente etapa del desarrollo del sistema es el diseño, que se utiliza para considerar las opciones de arquitectura, es decir, en esta etapa, procesaremos los bocetos de la interfaz que se codificarán. Estos bocetos deben derivarse a partir del modelo de análisis de sistemas de información anteriormente mencionados y analizados, de los mismos deben tomarse en cuenta las diferentes etapas, tomando como principales los casos de uso, diagramas de flujo, de actividad y los que crea pertinentes, en las cuales se detalla los requerimientos en sí del software a desarrollarse. [67]

Para la construcción de los bocetos de interfaz, se utilizará la herramienta Lucidchart, la misma que ofrece un sinnúmero de diagramas principalmente para maquetaciones de software y sistemas de información.

c) Codificación o Implementación

La etapa de implementación del sistema de información consiste principalmente en la codificación o programación del sistema a desarrollar, incluye introducir todos los contenidos diseñados en el paso anterior en el código fuente según el lenguaje de programación elegido por el programador. El programador debe seguir completamente las recomendaciones de la fase de diseño y la fase de análisis. Es notable apreciar que esta etapa es la que más tiempo consume, pero un análisis y un diseño correctos son fundamentales para la ejecución concisa, rápida y básicamente efectiva de esta etapa. [67]

Además, se considera el tiempo de depuración del código por errores en la fase de desarrollo, aunque la siguiente fase se encarga de buscar posibles errores

para liberar las soluciones propuestas por el programador, sin embargo, es necesario que la entrega de esta etapa tenga la menor cantidad de errores posibles. [67] En esta etapa, el código puede adoptar varios estados según el modo de trabajo y el lenguaje de programación utilizado, como el caso de la codificación directa, el caso de referenciar el compilador y el caso relacionado con archivos ejecutables. [67]

En esta fase se empleará el algoritmo que fue previamente evaluado, para lo cual se utiliza el lenguaje de programación Python conjuntamente con el Framework Django y el IDE Visual Studio Code. Se eligió Python porque este lenguaje de programación permite importar fácilmente bibliotecas para algoritmos de minería de datos. Además, otro factor importante que influyó en la elección de este lenguaje de programación fue que el Core del sistema Ecuciencia se lo estableció en Python.

d) Pruebas

En esta etapa, se probará el sistema en desarrollo, utilizando principalmente pruebas unitarias y pruebas de integración. Además de las conocidas pruebas de estrés (que llevan la respuesta del software al límite para probar su tolerancia y robustez), estas pruebas se realizan utilizando un conjunto de datos típicos que el sistema puede elegir. También debe considerar realizar pruebas para situaciones en las que el sistema no se verá afectado o no debería verse afectado en circunstancias normales, pero que generalmente ocurre debido a errores o accidentes.

Las pruebas se centran en el flujo lógico interno del software (para garantizar que se hayan comprobado todas las declaraciones) y el flujo externo funcional (es decir, el rendimiento de la prueba de error). Para poder proporcionar retroalimentación a los desarrolladores, se requiere poder probar el software con sujetos reales que puedan evaluar el comportamiento del software. [73]

Por lo general, se suele realizar una prueba final sobre el software denominada prueba Beta, en el cual el sistema es instalado en condiciones parecidas al que se va encontrar a partir de su versión de lanzamiento, el propósito es encontrar posibles errores, inestabilidad, etc., que fueron omitidas en el testeado dentro del

proceso de desarrollo. [73] Esta prueba generalmente la realizan personas que conocen el funcionamiento final del sistema. En caso de errores o mal funcionamiento, se transmitirán al programador para su depuración.

2.3. DISEÑO EXPERIMENTAL Y MÉTODO DE CRITERIO DE EXPERTOS

2.3.1. MÉTODO DE VALIDACIÓN CRUZADA

En el ámbito de Inteligencia Artificial existen varios métodos que pueden ser utilizados para realizar el proceso de validaciones. Sin embargo, muchos expertos consideran que el método de Validación Cruzada o Cross-Validation es el más recomendado para realizar el análisis del nivel de exactitud que un algoritmo posee. Según [74], la validación cruzada es una técnica que se utiliza para evaluar los resultados del análisis estadístico y garantizar que sea independiente de la división entre los datos de entrenamiento y los datos de prueba.

La aplicación de este método de validación en la presente propuesta constará de varias etapas fundamentales:

- **Etapas 1:** En la primera etapa, se utilizará los datos de documentos científicos, así como sus palabras claves y sublíneas de investigación, pertenecientes a la carrera de Ingeniería en Sistemas de Información de la Universidad Técnica de Cotopaxi esto con la finalidad de poder segmentar la muestra de los datos y tener un conjunto más resumido, pero a la vez con mayor consistencia de datos.
- **Etapas 2:** Posteriormente se analizarán los datos obtenidos, realizando el proceso de entrenamiento utilizando el algoritmo de aprendizaje profundo seleccionado. En esta etapa se separará el conjunto de datos en dos subconjuntos, uno será utilizado para entrenar el modelo, y el otro será utilizado para realizar los test de validación. De esta forma, el modelo se puede crear utilizando solo los datos de entrenamiento. Con el modelo creado, los datos de salida se generarán y se compararán con el conjunto de datos reservados para su verificación. Cuando finalice el análisis se obtendrán los datos estadísticos pertenecientes al nivel de

exactitud de entrenamiento y el número de datos perdidos en el proceso. A esta etapa se la conoce como método “hold-out”.

- **Etapa 3:** Una vez obtenido el nivel de exactitud de entrenamiento del algoritmo de aprendizaje profundo, utilizando los datos seleccionados de Eficiencia, se procederá a aplicar el método “k-fold”, el cual consiste en evaluar “k” número de veces el modelo aplicando la técnica “hold-out”, es decir, se iterará el entrenamiento del algoritmo de aprendizaje profundo la veces que se especifique en la variable “k” para obtener un histórico de porcentaje de exactitud y establecer un promedio general de todas las veces que se realizó el proceso de entrenamiento.
- **Etapa 4:** Y finalmente, gracias a los datos obtenidos tanto en el método “hold-out” como en el “k-fold”, se podrá realizar representaciones gráficas de las estadísticas obtenidas en el proceso de validación para poder determinar el nivel de aceptación establecida en la escala de fiabilidad representada en la tabla 3.

Tabla 3 Escala de fiabilidad.

Muy Baja	Baja	Moderada	Buena	Alta
0 - 0,20	0,21 - 0,40	0,41 - 0,60	0,61 - 0,80	0,81 - 1,0

Elaborado por: Investigador

Un dato importante a tomar en cuenta, es que se puede aprovechar las etapas establecidas en la metodología KDD, puesto que la construcción de los métodos de Variación Cruzada se lo puede contemplar en la etapa de Minería de datos, y el proceso de ejecución del entrenamiento y validación utilizando los métodos hold-out y k-fold se lo realizaría en la etapa de Interpretación y Evaluación.

2.4. DESCRIPCIÓN METODOLÓGICA DE LA VALORIZACIÓN ECONÓMICA, TECNOLÓGICA, OPERACIONAL Y MEDIO AMBIENTAL DE LA PROPUESTA.

2.4.1. VALORACIÓN ECONÓMICA:

En la valoración económica, es importante mencionar los costos directos e indirectos, a través de los cuales se obtendrán los costos reales de la propuesta de

investigación. Con respecto al desarrollo de software, lo mejor es utilizar herramientas de código abierto, que pueden reducir considerablemente los costos al final del proyecto.

2.4.2. VALORACIÓN TECNOLÓGICA:

En esta valoración se considera las características y recursos necesarios para la ejecución del módulo realizado. Se realiza un análisis en base a los tiempos de respuesta del host en el cual se implementó la solución y se toma como referencia para establecer los requisitos mínimos de hardware y software.

2.4.3. VALORACIÓN AMBIENTAL:

Dado que Ecuciencia es un sistema web, implica que los docentes investigadores de la universidad puedan digitalizar sus artículos, libros y ponencias teniendo la posibilidad de acceder desde cualquier lugar, con lo que disminuye considerablemente el uso de archivos físicos para el traslado de la información.

2.5. CONCLUSIONES CAPÍTULO II

- Debido a la compatibilidad entre la metodología KDD y la del modelo Iterativo e Incremental, se puede realizar un híbrido para mejorar los tiempos empleados en la resolución de la propuesta.
- El Método Cross-Validation será de gran ayuda en la validación de la propuesta, ya que el mismo permite verificar el porcentaje de exactitud que un algoritmo genera en el proceso de entrenamiento. Este método es el más recomendado ya que utiliza la información del dataset de Ecuciencia para realizar el test de validación, lo que genera una evaluación directa del impacto que tienen los algoritmos en resolver la problemática.

CAPÍTULO III. APLICACIÓN Y/O VALIDACIÓN DE LA PROPUESTA

3.1. RESULTADOS DE DIAGNÓSTICO DEL PROBLEMA

La presente investigación se orienta al mejoramiento del proceso de clasificación en la plataforma Ecuciencia, utilizando principalmente algoritmos de aprendizaje profundo que nos proporciona la librería TensorFlow de Google. La idea principal es que mediante la explotación de los datos almacenados en la base de datos de Ecuciencia, el sistema sea capaz de realizar un barrido y depuración de campos necesarios para obtener los patrones que influirán en el proceso de clasificación. Actualmente la plataforma Ecuciencia, tiene almacenado información de artículos, libros y ponencias realizados por los investigadores de la Universidad Técnica de Cotopaxi; dichos documentos científicos contienen una importante cantidad de información relevante en contexto científico, la misma que podría ser utilizada para realizar un proceso de minería de datos y texto para lograr entrenar algoritmos de inteligencia artificial y de esta manera generar que la herramienta tenga la capacidad de predecir la clasificación de líneas y sublíneas de investigación de futuros ingresos de publicaciones.

Como se menciona durante el proceso de investigación, la solución más viable que se considera para afrontar la problemática es la aplicación de algoritmos de Deep Learning o aprendizaje profundo, los cuales son proporcionados por la herramienta TensorFlow. Para la aplicación y evaluación de los algoritmos, se emplea la metodología KDD, la misma que es muy útil para los procesos de inteligencia artificial, y por ende será más sencillo la manipulación de los datos y la visualización de resultados.

El proceso de autoaprendizaje del algoritmo se lo realizará gracias al análisis constante de datos; lo que se pretende es implementar una política de activación del

algoritmo para que pueda estar regularmente realizando un análisis con los datos históricos y los nuevos datos que se vayan ingresando desde la última ejecución realizada. Esto con la finalidad de garantizar que las predicciones sean lo más acertadas posibles y que no tenga problemas al momento de realizar una evaluación de nuevos datos.

3.1.1. TÉCNICAS DE INVESTIGACIÓN

a) Observación:

Al realizar un proceso de observación de la funcionalidad de la plataforma Ecuciencia, se pudo constatar la poca eficiencia que poseen los algoritmos de inteligencia artificial presentes en el sistema. Se pudo evidenciar que el proceso de predicción de Líneas y Sublíneas de investigación posee un problema, puesto que los algoritmos empleados, no tienen un adecuado proceso de retroalimentación, por lo que es muy fácil que, si existe un error en el análisis, no exista una política de reintentos y sus datos procesados erróneamente sean los mostrados al usuario final.

b) Entrevista:

La entrevista se realizó con el PhD. Gustavo Rodríguez quien se desempeñaba como coordinador del proyecto de la Red de Estudios Cuantitativos (REDEC), el mismo que fue uno de los pioneros en el desarrollo del sistema Ecuciencia. Con el análisis de las respuestas proporcionadas, se puede establecer el primer paso de la metodología de desarrollo de software establecido para la resolución de la presente propuesta. Se realizaron preguntas referentes al estado actual del Sistema Ecuciencia y las nuevas funcionalidades que hacen falta implementar en el software para su mejora.

Para lo cual se realizó las siguientes preguntas:

1. ¿Cuál es el objetivo de desarrollar el sistema denominado Ecuciencia?

El desarrollo del sistema Ecuciencia tiene como propósito recolectar información y documentos científicos relacionados con las investigaciones realizada en la Universidad Tecnológica Cotopaxi, y el alcance del proyecto es cubrir todas las universidades del Distrito 3 e incluso todo el país.

Este proyecto se centra en la cienciometría, que es la ciencia a cargo del campo de investigación, cabe mencionar que las métricas que establece son amplias y se actualizarán constantemente. Hoy en día, en este país, la investigación juega un papel fundamental en su desarrollo e innovación.

Las demás preguntas se encuentran en el Anexo I.

3.2. RESULTADOS DE LOS MÉTODOS ESPECÍFICOS

A través de la entrevista con el coordinador de REDEC se obtuvieron los requisitos necesarios para la implementación de la propuesta de investigación. En la entrevista se pudieron concluir los problemas actuales del sistema ECUCIENCIA, por lo que es significativo mejorar el proceso de clasificación de los datos de producción científica según la línea y sublínea de investigación en la Universidad Tecnológica Cotopaxi. Los requisitos derivados a partir de la encuesta y de la observación serán el punto de partida para la correcta resolución del problema de investigación.

Una vez especificado la necesidad a resolver, es preciso comenzar aplicando las metodologías de desarrollo e inteligencia artificial expuestas en el capítulo anterior.

3.2.1. METODOLOGÍA KDD

Gracias a la entrevista realizada con anterioridad, y también como se constató en el análisis realizado mediante el proceso de la observación, se pudo evidenciar el uso de PostgreSQL como el gestor de base de datos principal. Una vez identificado el almacenamiento de datos, es posible realizar las etapas establecidas por la metodología KDD.

a) Selección

En la primera etapa del método se seleccionó e identificó la estructura interna de la base de datos que utiliza el sistema Ecuciencia, se observó que la base de datos hasta el momento de la que se desarrollaba la investigación supera las 60 tablas interrelacionadas. En la figura 7 se muestra en detalle el listado de las tablas que la componen, así como también la interfaz gráfica del gestor PostgreSQL.

Name	Owner	Partitioned table?	Comment
Articulos_Cientificos_articulos_cientificos	postgres	False	
Articulos_Cientificos_articulos_cientificos_baseDatos	postgres	False	
Articulos_Cientificos_articulos_cientificos_palabraClave	postgres	False	
Formacion_Academica_formacion_academica	postgres	False	
Investigador_investigador	postgres	False	
Investigador_investigador_roles	postgres	False	
Investigador_investigadorauxiliar	postgres	False	
Libro_libro	postgres	False	
Libro_libro_BaseDatos	postgres	False	
Libro_libro_PalabrasClave	postgres	False	
Linea_Investigacion_linea_investigacion	postgres	False	
Palabra_clave_palabra_clave	postgres	False	
Ponencia_ponencia	postgres	False	
Ponencia_ponencia_palabrasClave	postgres	False	
Privilegios_privilegios	postgres	False	
Proyectos_proyecto	postgres	False	
Proyectos_proyecto_palabrasClaves	postgres	False	
Revista_revista	postgres	False	
Revista_revista_base	postgres	False	
Sub_Lin_Investigacion_sub_lin_investigacion	postgres	False	
TextMining	postgres	False	
Unidad_Investigacion_unidad_investigacion	postgres	False	
Unidades_Investigacion_unidades_investigacion	postgres	False	
atributo_palabracarrera	postgres	False	
auth_group	postgres	False	
auth_group_permissions	postgres	False	
auth_permission	postgres	False	
auth_user	postgres	False	

Figura 7 Tablas de Base de Datos Ecuciencia

Fuente: Investigador

b) Preprocesamiento/limpieza.

La base de datos del sistema Ecuciencia posee un amplio número de tablas en donde se almacenan sus datos gestionados, sin embargo, para realizar la implementación de la presente propuesta es necesario solo las relacionadas con los documentos científicos. Para seleccionar los parámetros necesarios en el proceso de minería de datos y la aplicación del algoritmo, se basa en los indicadores de medición científica que se muestran en la figura 8.

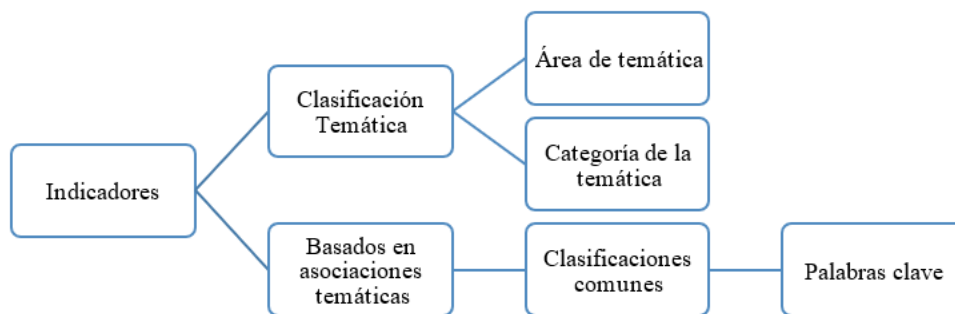


Figura 8 Indicadores Cienciométricos

Fuente: Investigador

Gracias al apoyo del gráfico, se pudo interpretar que los datos más acertados para realizar el análisis mediante los algoritmos, son todos los relacionados a las palabras

claves. Para tener una mejor concepción de las relaciones existentes entre las tablas de la base de datos, se ha elaborado en la figura 9 el diagrama entidad relación de sus entidades principales.

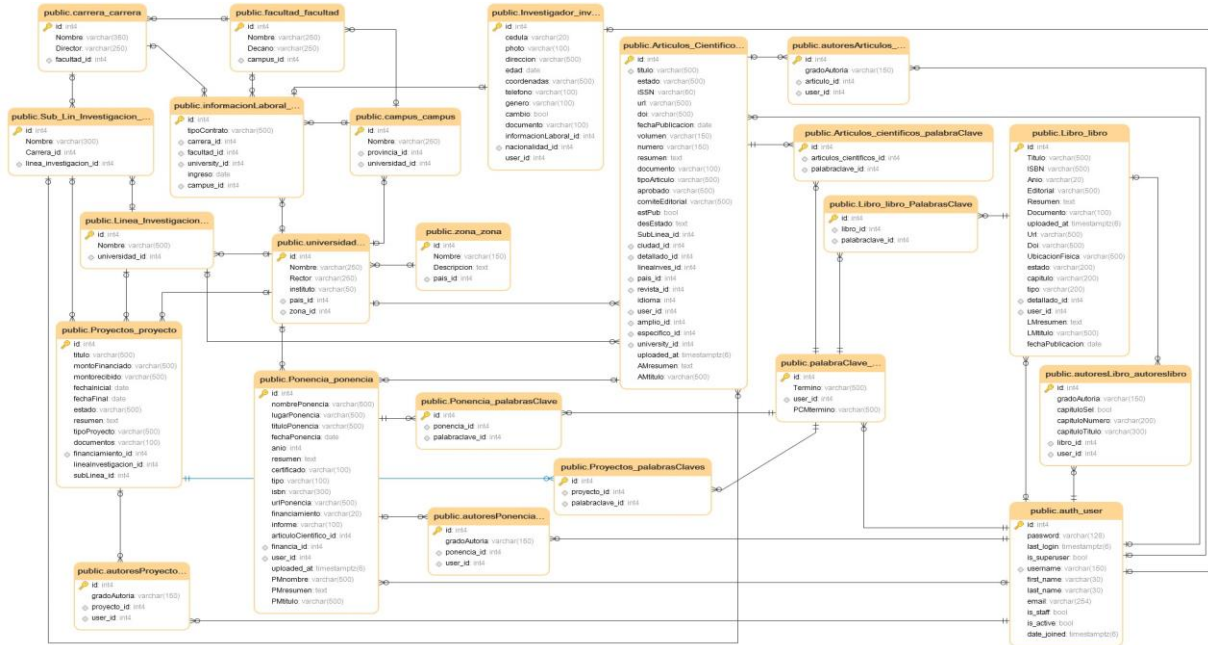


Figura 9 Diagrama Entidad - Relación

Fuente: Investigador

c) Transformación/reducción.

En la minería de datos, un proceso muy importante es definir y seleccionar los atributos de las tablas que componen la base de datos, estos atributos se utilizan para realizar un análisis correcto de información y así lograr obtener los resultados esperados, evitando así inconsistencias de datos. La figura 10 muestra la definición de la tabla y los atributos que se utilizarán en el algoritmo seleccionado. De la misma manera, se mencionará el tipo de datos general del atributo, es decir, el tipo de datos numérico, alfabético o alfanumérico, y también se señalarán los atributos que sean establecidos como claves primarias.

Tabla	Descripción de Tabla	Atributo	Tipo de Dato	Descripción de Atributo
Articulos_Cientificos	En esta tabla se almacenan los datos respectivos de un artículo científico.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		PalabraClave	varchar	Atributo que registra las palabras claves del artículo científico
AutoresArticulos	En la tabla se almacena la información relacionada al grado de autoría que tiene un investigador en un artículo científico.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Articulo_id	foreign key	Clave foránea, que indica con que artículo científico se relaciona el autor.
		User_id	foreign key	Clave foránea, que indica con que registro de la tabla auth_user está relacionada la el autor del artículo.
Libro	Tabla en la cual se almacena la información relacionada con los datos de los libros.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		PalabraClave	varchar	Atributo que registra las palabras claves del libro
AutoresLibro	En la tabla se almacena la información relacionada al grado de autoría que tiene un investigador en un libro.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Libro_id	foreign key	Clave foránea, que indica con que libro se relaciona el autor.
		User_id	foreign key	Clave foránea, que indica con que registro de la tabla auth_user está relacionada la el autor del libro.
Ponencia	En esta tabla se almacenan los datos respectivos a una ponencia.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		PalabraClave	varchar	Atributo que registra las palabras claves de la ponencia
AutoresPonencia	En la tabla se almacena la información relacionada al grado de autoría que tiene un investigador en una ponencia	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Ponencia_id	foreign key	Clave foránea, que indica con que ponencia se relaciona el autor.
		User_id	foreign key	Clave foránea, que indica con que registro de la tabla auth_user está relacionada la el autor de la ponencia.
Proyecto	En esta tabla se almacenan los datos respectivos de un proyecto.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		PalabraClave	varchar	Atributo que registra las palabras claves del proyecto
AutoresProyecto	En la tabla se almacena la información relacionada al grado de autoría que tiene un investigador en un proyecto.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Proyecto_id	foreign key	Clave foránea, que indica con que proyecto se relaciona el autor
		User_id	foreign key	Clave foránea, que indica con que registro de la tabla auth_user está relacionada la el autor del proyecto.

Tabla	Descripción de Tabla	Atributo	Tipo de Dato	Descripción de Atributo
Auth_user	En esta tabla se almacena la información relacionada con el investigador que registra la producción científica. Los atributos de esta tabla se utilizarán para obtener los datos personales del investigador.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		First_name	varchar	Atributo que registra los nombres del investigador
		Last_name	varchar	Atributo que registra los apellidos del investigador
		Email	varchar	Atributo que registra el email del investigador
Investigador	En esta tabla se almacena los datos personales de los investigadores registrados en la plataforma científica Ecuciencia. El id de estos registros servirá para obtener la información de las tablas con las cuales se relaciona el investigador.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Cedula	varchar	Atributo que registra el numero de identificación del investigador
		User_id	foreign key	Clave foránea, que indica con que registro de la tabla auth_user está relacionado el investigador
		InformacionLaboral_id	foreign key	Clave foránea, que permite identificar con que registro de la tabla información laboral está relacionado el investigador
InformacionLaboral	En esta tabla se almacena los datos relacionados con la información laboral del investigador. El atributo carrera_id servirá para obtener algunos datos del registro con el cual esté relacionado en la tabla carrera.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Carrera_id	foreign key	Clave foránea, que permite identificar con que registro de la tabla carrera está relacionado la información laboral.
Carrera	En esta tabla se almacén los datos referentes a las carreras de la Universidad. El id de esta tabla se utiliza como clave foránea en las tablas informacionLaboral y Sub_Lin_Investigacion.	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Nombre	varchar	Atributo que registra el nombre de la carrera
Palabra_Clave	En esta tabla se almacena las palabras claves que contienen los artículos, libros y ponencias elaborados por los investigadores registrados en plataforma científica	Id	int autoincremental (primary key)	Atributo que identifica de forma única a cada registro
		Termino	varchar	Atributo que registra la palabra clave
		User_id	foreign key	Clave foránea, que indica con que registro de la tabla auth_user está relacionada la palabra clave.

Figura 10 Definición de Tablas y Atributos

Fuente: Investigador tomando como referencia a *Falconí y Gualpa [66]*

d) Minería de datos.

Para el desarrollo de la presente propuesta se han empleado varios algoritmos de inteligencia artificial, los mismos que han logrado establecer un trabajo en comunión con la finalidad de lograr el objetivo trazado. Los algoritmos principales que se han utilizado en la resolución de la propuesta son:

Algoritmo SKLearn: Esta librería es una de las más conocidas en el proceso de Machine Learning, internamente tiene varios módulos que son utilizados dependiendo el caso para análisis, clasificación, predicción, entre otras opciones más.

Los módulos que fueron utilizados de esta librería fueron el “preprocessing” y el “metrics”, los cuales fueron de mucha ayuda para el tratamiento de datos de entrenamiento y test.

Con el “preprocessing” se pudo codificar las etiquetas de las clases que para este caso vendrían a ser las líneas y sublíneas de investigación, y el “metrics” fue necesario para poder evaluar los datos de “train” y “test” a través de la matriz de confusión. En la figura 11 se muestra el resultado del proceso de evaluación considerando tres clases correspondientes a las sublíneas de investigación de la carrera de Ingeniería en Sistemas de Información:

MODELACIÓN DE SISTEMAS DE INFORMACIÓN	17	0	0
REDES Y SEGURIDAD COMPUTACIONAL	0	32	0
ROBÓTICA E INTELIGENCIA ARTIFICIAL	4	0	19
	MODELACIÓN DE SISTEMAS DE INFORMACIÓN	REDES Y SEGURIDAD COMPUTACIONAL	ROBÓTICA E INTELIGENCIA ARTIFICIAL

Figura 11 Matriz de confusión

Fuente: Investigador

Algoritmo NLTK: Es una de las librerías de minería de texto más utilizadas en la actualidad, internamente tiene módulos de análisis apoyados en la clasificación supervisada, y es muy útil para tokenizar palabras de un texto y realizar el tratamiento dependiendo de los parámetros especificados. Para la presente propuesta se lo empleó en una tarea muy importante, el realizar una limpieza de palabras comunes o “stopwords”, lo cual genera que todas las palabras que no tengan relevancia en el contexto científico, sean excluidas, esto con el fin de garantizar que los textos analizados tengan el menor ruido posible.

En la figura 12 se observa un ejemplo de un texto almacenado en la base de datos correspondiente a un artículo científico, y en la figura 13 se observa las palabras extraídas por el algoritmo para su tratamiento posterior.

La antigua jurisdicción de Bayamo, como división política de la colonia comenzó su andadura en el siglo XVI, tras su fundación en 1513 como la segunda villa de Cuba. Actualmente el municipio de Bayamo lo integran 15 consejos populares, una superficie de 835,12 km² y una población de 68690 habitantes. A pesar de haber transcurrido más de 470 años desde la llegada de los primeros cerdos desde España, esta región ha mantenido 6176 reproductores de la raza cerdo Criollo como descendiente directo de los cerdos mediterráneos. En este trabajo se presenta los factores racial, ecológico y humano, que han permitido la perdurabilidad de esta raza descendiente del cerdo Ibérico.

✕ Cancel OK

Figura 12 Texto de Resumen de Artículo Científico

Fuente: Investigador

ecologico, iberico, jurisdiccion, criollo, perdurabilidad, divis, cerdo, mantenido, racial, factor, municipio, consejo, antigua, poblacion, superficie, directo, raza, humano, bayamo, permitido, habitante, transcurrido, reproductor, colonia, espana, politica

✕ Cancel OK

Figura 13 Palabras extraídas al aplicar el algoritmo NLTK

Fuente: Investigador

Una de las ventajas del uso de NLTK es que puede soportar múltiples idiomas en un solo análisis, ya que como se lo aprecia en la figura 14, uno de los parámetros de su instancia es el lenguaje a ser procesado.

```
for word in x2:
    if word not in stopwords.words('spanish'):
        x3.append(word)
```

Figura 14 Líneas de código del constructor del Algoritmo NLTK para análisis en español

Fuente: Investigador

Y considerando que Ecuciencia es una plataforma con visión de expansión a varios países e idiomas es necesario contar con estos soportes a multi idiomas, en la figura 15 se puede apreciar un texto de un artículo científico desarrollado en el idioma inglés, y en la figura 16 se visualiza los resultados del algoritmo.

This research studied the role of knowledge organization in the process of decision making in the field of energy efficiency and rational use of energy (EERUE). Theoretical contributions to knowledge organization and decision making are stressed. We chose to work with a type of methodology known as multiple criteria decision making –Saaty’s analytical hierarchies process (AHP). This made it possible to develop a detailed analysis of the decision-making process and arrive at a hierarchical model. The model provided a structure representing the studied field, where an order of priority could be given to the decision-making process. The knowledge derived may be used in other fields of study such as information retrieval and knowledge representation.

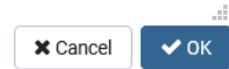


Figura 15 Texto de Resumen de Artículo Científico en Inglés

Fuente: Investigador

analyt, multipl, model, knowledge, contribute, inform, criteria, decision, hierarch, stress, ration, order, studi, field, saati, energi, role, retriev, methodolog, work, organ, analysi, research, hierarchi, represent, detail, type, process

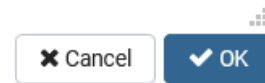


Figura 16 Palabras extraídas al aplicar el algoritmo NLTK

Fuente: Investigador

TensorFlow: Es una herramienta prácticamente nueva, fue desarrollada por Google y es licencia libre. TensorFlow posee una amplia gama de algoritmos de aprendizaje profundo los cuales pueden ser utilizados sin restricción alguna. TensorFlow utiliza el lenguaje de programación Python como su Core principal, por lo que es totalmente compatible para el desarrollo de esta propuesta.

Como se mencionó anteriormente, TensorFlow cuenta con varios algoritmos de Deep Learning de licencia libre. El algoritmo que se utilizó para el desarrollo del análisis y aprendizaje profundo fue Keras.

Keras es un algoritmo que emplea Redes Neuronales Artificiales orientadas al Aprendizaje Profundo. Es muy sencilla de utilizar, pero ofrece un análisis potente. Es muy utilizada para el análisis predictivo de un conjunto de datos establecidos por clases y patrones comunes. Posee un motor de clasificación muy avanzado y en ocasiones puede ser utilizado a través del análisis media API’s.

Keras es de mucha ayuda para el desarrollo de la propuesta, puesto que el análisis y clasificación depende en su mayoría de los resultados arrojados por este algoritmo. El trabajo de Keras estará dividido en dos partes fundamentales. En la primera, utilizará módulos para el análisis de los textos presentados por el algoritmo NLTK, en esta parte Keras se comunicará con los métodos nativos de TensorFlow

y realizará un análisis evaluando los datos por épocas. Es necesario establecer algunos parámetros para el análisis de tensores, los mismo vienen detallados en la tabla 4; estos parámetros son los que regulan los datos de entrenamiento y test y cuál es el número máximo de épocas en el proceso de análisis.

Tabla 4 Parámetros enviados al algoritmo para su ejecución.

Nombre de Variable	Descripción	Valor
epochs	Máximo número de veces que el algoritmo repetirá el análisis antes de detenerse.	1000
training_percentage	Porcentaje de datos destinados para el entrenamiento del modelo	80
training_size	Número de datos destinados para el entrenamiento del modelo	Se calcula multiplicando la longitud del dataset con el porcentaje de entrenamiento
x_train	Datos de entrenamiento	Datos obtenidos del dataset de entrenamiento
x_test	Datos de test	Datos obtenidos del dataset destinados para test
y_train	Clases de entrenamiento	Sublíneas de investigación
y_test	Clases de test	Sublíneas de investigación

Fuente: Investigador

En la segunda parte, Keras recibirá un nuevo texto a analizar; en esta parte Keras realizará un análisis predictivo de los nuevos datos ingresados con los resultados previamente obtenidos. En este punto Keras entregará al usuario final los porcentajes de predicción y clasificación que se le dio para cada una de las clases existentes en el modelo precargado.

Para el análisis de datos de entrenamiento y test se utilizan los siguientes módulos:

- **Models:** El cual se encarga de parsear los datos de las listas obtenidas de la base de datos en modelos requeridos por Keras para su análisis.
- **Layers:** Se utiliza para agrupar todas las listas de datos en un solo paquete que será más manejable al momento del análisis.
- **Preprocessing:** Tiene dos funciones principales, la primera, se encarga de tokenizar el conjunto de datos establecidos en los “Layers”, y la segunda enviar los datos al modelo predictivo mediante la función “fit_on_texts()”.

Para el proceso de evaluación de nuevos datos ingresados se utiliza el siguiente módulo:

- **Models:** Es utilizado para cargar el modelo obtenido en el proceso de análisis y dar una clasificación al nuevo texto de ingreso a través del método “model.predict()”.

e) Interpretación/evaluación

Cuando se ejecuta el algoritmo enviando los parámetros especificados en la tabla 4, lo primero que se puede observar es como TensorFlow realiza el proceso de entrenamiento en las diferentes épocas hasta encontrar la menor cantidad de datos perdidos. En la figura 17 se puede evidenciar parte de los logs producidos en el proceso de análisis que TensorFlow realiza. Adicionalmente en la figura 18 se puede visualizar el final de los logs producidos; el parámetro enviando en la variable de épocas fue 1000 sin embargo como se aprecia en la figura 18, el algoritmo solo necesito de 89 épocas para entrenar el modelo de manera eficiente.

En los logs también se pueden apreciar algunas variables importantes que se producen en el proceso de análisis, entre ellas se encuentra:

- Loss
- Categorical_accuracy
- Val_loss
- Val_categorical_accuracy

```

2021-01-08 09:53:48.044186: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default Version
2021-01-08 09:53:48.104266: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1257] Device interconnect StreamExecutor with strength 1 edge matrix:
2021-01-08 09:53:48.140534: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1263]
Epoch 1/1000
8/8 [=====] - 0s 21ms/step - loss: 0.6556 - categorical_accuracy: 0.3477 - val_loss: 0.6316 - val_categorical_accuracy: 0.3793
Epoch 2/1000
8/8 [=====] - 0s 5ms/step - loss: 0.6351 - categorical_accuracy: 0.3711 - val_loss: 0.6210 - val_categorical_accuracy: 0.3793
Epoch 3/1000
8/8 [=====] - 0s 4ms/step - loss: 0.6264 - categorical_accuracy: 0.3398 - val_loss: 0.6113 - val_categorical_accuracy: 0.5517
Epoch 4/1000
8/8 [=====] - 0s 3ms/step - loss: 0.6046 - categorical_accuracy: 0.4453 - val_loss: 0.6010 - val_categorical_accuracy: 0.6552
Epoch 5/1000
8/8 [=====] - 0s 3ms/step - loss: 0.5916 - categorical_accuracy: 0.4609 - val_loss: 0.5892 - val_categorical_accuracy: 0.6552
Epoch 6/1000
8/8 [=====] - 0s 3ms/step - loss: 0.5711 - categorical_accuracy: 0.5430 - val_loss: 0.5734 - val_categorical_accuracy: 0.6897
Epoch 7/1000
8/8 [=====] - 0s 4ms/step - loss: 0.5710 - categorical_accuracy: 0.4727 - val_loss: 0.5560 - val_categorical_accuracy: 0.6552
Epoch 8/1000
8/8 [=====] - 0s 3ms/step - loss: 0.5318 - categorical_accuracy: 0.5742 - val_loss: 0.5350 - val_categorical_accuracy: 0.6552
Epoch 9/1000
8/8 [=====] - 0s 4ms/step - loss: 0.5297 - categorical_accuracy: 0.5586 - val_loss: 0.5145 - val_categorical_accuracy: 0.6897
Epoch 10/1000
8/8 [=====] - 0s 3ms/step - loss: 0.5089 - categorical_accuracy: 0.5898 - val_loss: 0.4921 - val_categorical_accuracy: 0.6897
Epoch 11/1000
8/8 [=====] - 0s 3ms/step - loss: 0.4931 - categorical_accuracy: 0.5938 - val_loss: 0.4667 - val_categorical_accuracy: 0.6897
Epoch 12/1000
8/8 [=====] - 0s 3ms/step - loss: 0.4564 - categorical_accuracy: 0.6250 - val_loss: 0.4425 - val_categorical_accuracy: 0.6897
Epoch 13/1000
8/8 [=====] - 0s 3ms/step - loss: 0.4534 - categorical_accuracy: 0.6367 - val_loss: 0.4158 - val_categorical_accuracy: 0.6897
Epoch 14/1000
8/8 [=====] - 0s 3ms/step - loss: 0.4334 - categorical_accuracy: 0.6602 - val_loss: 0.3899 - val_categorical_accuracy: 0.6897

```

Figura 17 Inicio de log producido por el Análisis de TensorFlow

Fuente: Investigador

```

Epoch 78/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0673 - categorical_accuracy: 0.9844 - val_loss: 0.1067 - val_categorical_accuracy: 0.9310
Epoch 79/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0944 - categorical_accuracy: 0.9609 - val_loss: 0.1043 - val_categorical_accuracy: 0.9310
Epoch 80/1000
8/8 [=====] - 0s 4ms/step - loss: 0.0674 - categorical_accuracy: 0.9883 - val_loss: 0.1029 - val_categorical_accuracy: 0.9310
Epoch 81/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0876 - categorical_accuracy: 0.9688 - val_loss: 0.1019 - val_categorical_accuracy: 0.9310
Epoch 82/1000
8/8 [=====] - 0s 4ms/step - loss: 0.0802 - categorical_accuracy: 0.9727 - val_loss: 0.1023 - val_categorical_accuracy: 0.9310
Epoch 83/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0605 - categorical_accuracy: 0.9883 - val_loss: 0.1012 - val_categorical_accuracy: 0.9310
Epoch 84/1000
8/8 [=====] - 0s 4ms/step - loss: 0.0953 - categorical_accuracy: 0.9688 - val_loss: 0.0999 - val_categorical_accuracy: 0.9310
Epoch 85/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0719 - categorical_accuracy: 0.9766 - val_loss: 0.1000 - val_categorical_accuracy: 0.9310
Epoch 86/1000
8/8 [=====] - 0s 4ms/step - loss: 0.0931 - categorical_accuracy: 0.9688 - val_loss: 0.0996 - val_categorical_accuracy: 0.9310
Epoch 87/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0729 - categorical_accuracy: 0.9883 - val_loss: 0.0990 - val_categorical_accuracy: 0.9310
Epoch 88/1000
8/8 [=====] - 0s 3ms/step - loss: 0.0924 - categorical_accuracy: 0.9648 - val_loss: 0.0995 - val_categorical_accuracy: 0.9310
Epoch 89/1000
8/8 [=====] - 0s 4ms/step - loss: 0.0948 - categorical_accuracy: 0.9453 - val_loss: 0.0992 - val_categorical_accuracy: 0.9310
Epoch 00089: early stopping
3/3 [=====] - 0s 664us/step - loss: 0.0949 - categorical_accuracy: 0.9444
[08/Jan/2021 09:53:44] "GET /keras/index HTTP/1.1" 200 8722

```

Figura 18 Fin de log producido por el Análisis de TensorFlow con el límite de 89 épocas

Fuente: Investigador

Una vez que el algoritmo realiza el proceso de análisis, genera listas de datos, las mismas que gracias a la estructura de Django es muy fácil renderizarlas. En las vistas de Django se realiza el encapsulamiento de datos en un diccionario genérico

el mismo que será enviado al template especificado. Esta programación se la puede evidenciar en la figura 19.

Con el diccionario de datos enviados al template, se lo recibe y trata en JavaScript para posteriormente poder ser representados mediante gráficos estadísticos como por ejemplo el “Linear Plot”, que es uno de los más utilizados en Inteligencia Artificial. En la figura 20 se puede apreciar una porción del código de JavaScript que captura los datos recibidos desde el template.

```
for i in range(0, len(y_softmax)):
    probs = y_softmax[i]
    predicted_index = np.argmax(probs)
    y_pred_1d.append(predicted_index)

cnf_matrix = confusion_matrix(y_test_1d, y_pred_1d)
plt.figure(figsize=(44,37))
mat = []
for i in range(len(cnf_matrix)):
    vet = []
    for j in range(len(cnf_matrix[i])):
        vet.append(cnf_matrix[i][j])
    mat.append(vet)
labels = []
for i in range(len(text_labels)):
    labels.append(text_labels[i])
return render(request, 'keras/index.html', context={'cnf_matrix': mat, 'labels': labels, 'lista': listajson})
```

Figura 19 Renderizado de datos desde Vista a Template

Fuente: Investigador

```
<script>
var confusionMatrix = {{ cnf_matrix | safe }};
console.log('confusionMatrix', confusionMatrix);
var labels = {{ labels | safe }};
console.log('labels', labels);
const xrange = confusionMatrix.length - 1;
const yrange = confusionMatrix[0].length - 1;
const limit = confusionMatrix[xrange][yrange];
```

Figura 20 Captura de datos enviados desde la vista

Fuente: Investigador

Además, si podemos apreciar en la figura 20, existen líneas de códigos referentes a la impresión de logs en la consola, en la figura 21 se puede apreciar el log de la variable “confusionMatrix”, el mismo que presenta la matriz producida por los resultados del análisis de TensorFlow. En la figura 22 se imprime también el log de la variable “labels”, el cual representa el array de las clases de clasificación, que para nuestro caso de estudio corresponde a las sublíneas de investigación. Y por último en la figura 23, se muestra la lista de valores porcentuales de datos perdidos

generados en cada una de las épocas representadas en la figura 17, esto con la finalidad de visualizarlos a través de un gráfico lineal como se lo podrá apreciar en la figura 24.

Adicionalmente, en la figura 25 se muestra también una tabla de valores que indican el nivel de exactitud arrojada por la predicción, para este caso se muestran los datos correspondientes a “F1”, “PRECISION”, “RECALL” y “ACCURACY”.

Cabe recalcar que los datos representados en la figura 21 se utilizarán para visualizar una matriz de confusión dibujada a nivel de la interfaz gráfica del usuario como se lo mostró en la figura 11.

```
confusionMatrix (3) [...]
  ▶ 0: Array(3) [ 15, 0, 2 ]
  ▶ 1: Array(3) [ 0, 32, 0 ]
  ▶ 2: Array(3) [ 0, 2, 21 ]
      length: 3
  ▶ <prototype>: Array []
```

Figura 21 Log de “confusionMatrix”

Fuente: Investigador

```
labels (3) [...]
  0: "MODELACIÓN DE SISTEMAS DE IONFORMACIÓN"
  1: "REDES Y SEGURIDAD COMPUTACIONAL"
  2: "ROBÓTICA E INTELIGENCIA ARTIFICIAL"
  length: 3
  ▶ <prototype>: Array []
```

Figura 22 Log de “labels”

Fuente: Investigador

```
lista (..)
  ▶ 11: Array(54) [ 0.6359300017356873, 0.6227053999900818, 0.6041340827941895, ... ]
  ▶ 12: Array(54) [ 0.6316166520118713, 0.620806872844696, 0.6099954843521118, ... ]
  ▶ <prototype>: Object { ... }
```

Figura 23 Log de valores porcentuales producidos por las épocas

Fuente: Investigador

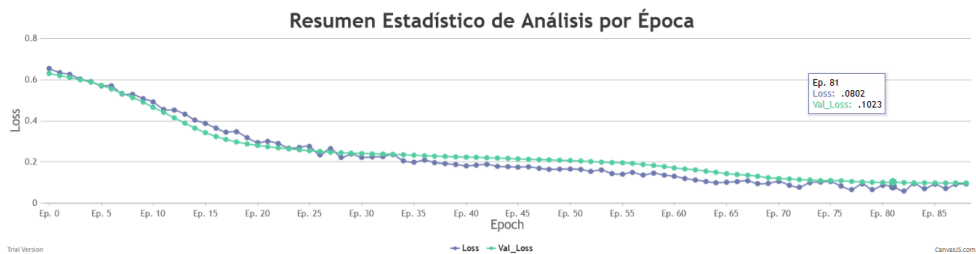


Figura 24 Gráfica de representación de valores porcentuales

Fuente: Investigador

FI	PRECISION	RECALL	ACCURACY
0.91	0.91	0.91	0.94

Figura 25 Variables de evaluación del modelo

Fuente: Investigador

Una vez entrenado el modelo, lo siguiente es probar la predicción, para ello es necesario introducir nuevos datos de entrada y visualizar que los resultados generados tengan coherencia. En la figura 26 se muestra la porción de código que es utilizado para evaluar el nuevo texto ingresado.

```

user_input = palabra
dataframe_input_test = pd.Series([user_input])
x_test = tokenizer_new.texts_to_matrix(dataframe_input_test, mode='tfidf')
prediction = model.predict(np.array([x_test[0]]))
sorting = (-prediction).argsort()
sorted_ = sorting[0][:sorting.size]
array = []
barArray = []
for value in sorted_:
    predicted_label = text_labels[value]
    prob = (prediction[0][value]) * 100
    prob = "%.2f" % round(prob,2)
    lista = {"y": prob,"name": predicted_label}
    barLista = {"y": prob,"label": predicted_label}
    array.append(lista)
    listajson = json.dumps(array)
    barArray.append(barLista)
    barlistajson = json.dumps(barArray)
return render(request, 'keras/analisis.html', context={'palabra': palabra, 'datajson': listajson, 'barlistajson': barlistajson})

```

Figura 26 Código de predicción de nuevo texto

Fuente: Investigador

Para el ejemplo emplearemos la frase de prueba “Los algoritmos de clasificación son muy utilizados en el análisis de modelos predictivos”, en base a la experiencia, se entiende que la frase corresponde a la sublínea de Robótica e Inteligencia Artificial. Para poder visualizar de mejor manera cuales son los resultados de predicción arrojados por el algoritmo se empleó dos gráficos estadísticos, el diagrama tipo pastel y el diagrama de barras; en la figura 27 se muestra el gráfico tipo pastel, en el cual se visualiza claramente la superioridad de la sublínea “Robótica e Inteligencia Artificial”, quien muestra un 99.98% de predicción, dejando a “Redes y Seguridad Computacional” con un 0.02% lo cual es casi nulo:

Porcentaje de Coincidencias

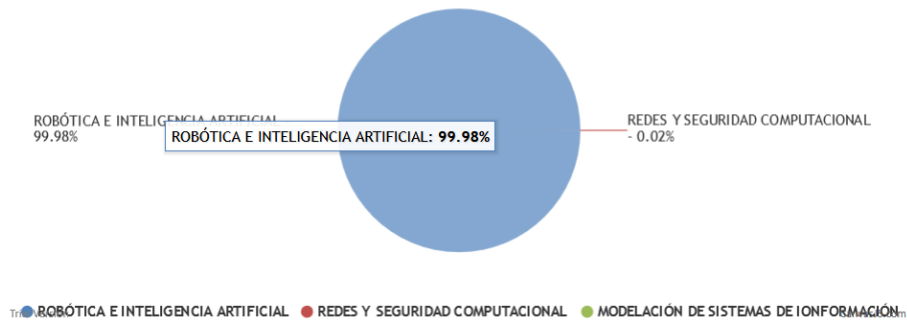


Figura 27 Diagrama de Pastel con valores porcentuales de predicción

Fuente: Investigador

En la figura 28, se muestra también una comparativa de los porcentajes obtenidos, sin embargo, esta vez en forma de barras:

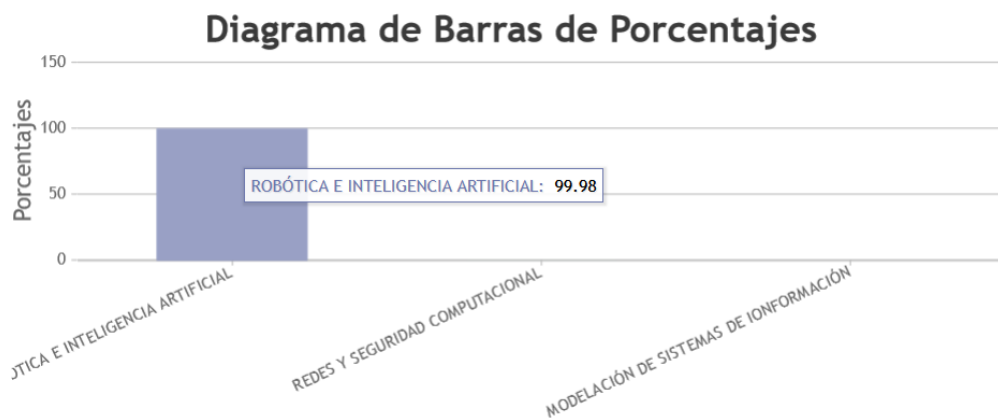


Figura 28 Diagrama de Barras con valores porcentuales de predicción

Fuente: Investigador

Entonces como conclusión existe una correcta predicción ya que como vimos las estadísticas porcentuales se apegan a la realidad por lo que se concluye que el modelo entrenado es confiable y evalúa datos coherentes y consistentes. Por último, como resultado también se establece en un diagrama de porcentaje, cual es el valor porcentual que obtuvo la clase con más puntuación, esta información se la puede observar en la figura 29.

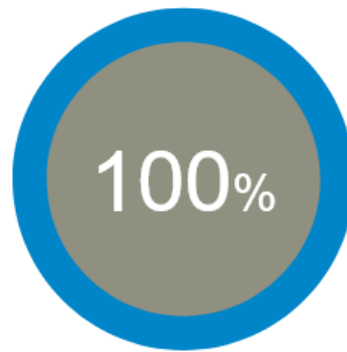


Figura 29 Porcentaje de predicción de clase con mayor peso

Fuente: Investigador

En conclusión, la aplicación del algoritmo Keras de TensorFlow conjuntamente con el resto de algoritmos citados, realizaron un trabajo correcto de predicción, y por ende se puede aprobar el uso de los mismo en el proceso del desarrollo del nuevo módulo.

3.2.2. MODELO ITERATIVO INCREMENTAL

Gracias a que la metodología del modelo Iterativo Incremental es muy flexible y puede ser utilizada para otros marcos de trabajo que no sea precisamente el desarrollo de software, se puede realizar un trabajo en conjunto con la metodología KDD para complementar de mejor manera la propuesta tecnológica. En la figura 30, se puede visualizar un híbrido resultante de la combinación de ambas metodologías, siempre respetando los pasos que cada una establece de manera individual. Cabe recalcar que esta nueva metodología mostrada es el resultado de la presente propuesta, por lo que es posible que no se tenga algún antecedente de la misma.

Para la aplicación de la Metodología del Modelo Iterativo Incremental en la presente propuesta, se decidió dividirla en cuatro Iteraciones. La primera que hace referencia al tratamiento de datos utilizando la librería NLTK; el trabajo realizado en esta iteración fue netamente a nivel de Backend por consiguiente no existe una representación visual del resultado obtenido. En la segunda iteración, se realizó una minería de texto aplicando el algoritmo SKLearn, el mismo que se encargó de recopilar los datos salientes de la primera iteración y analizarlos para generar patrones de clasificación y como resultado se obtuvo los valores para la matriz de

confusión observados en la figura 11. La tercera iteración se la dedicó al desarrollo del algoritmo Keras de TensorFlow, en dicha iteración se realizó tanto la programación para el entrenamiento del algoritmo, así como también la predicción cuando existe un nuevo texto a analizar. Y la última iteración se la dedico netamente a la parte del Frontend, puesto que, con los datos resultantes tanto de la segunda como la tercera iteración, se puede realizar gráficas estadísticas presentables al usuario. En la figura 31 se establece el diagrama correspondiente a las iteraciones realizadas.

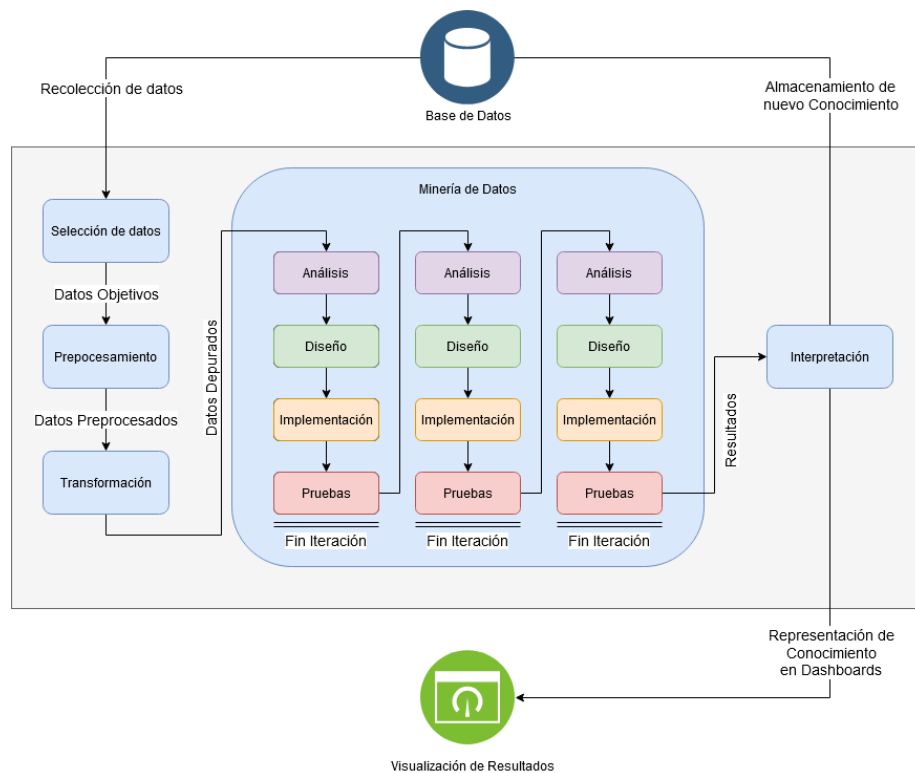


Figura 30 Metodología KDD – Iterativa Incremental

Fuente: Investigador

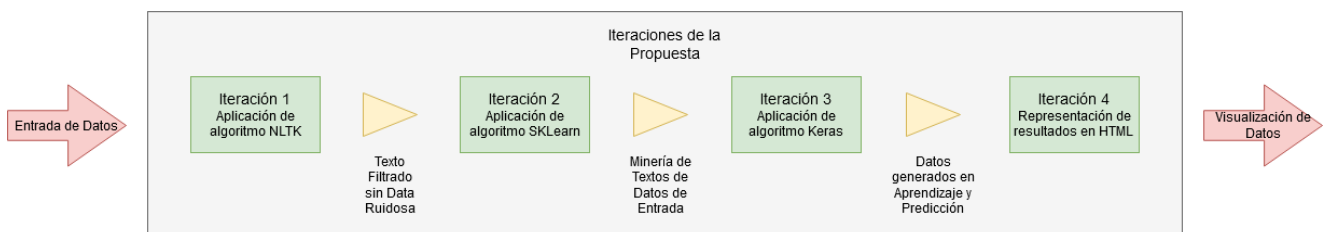


Figura 31 Diagrama de Iteraciones

Fuente: Investigador

A continuación, se recopilarán las distintas etapas de la metodología de manera general, es decir, no se especificarán en base a iteraciones, sino en base a la propuesta en su totalidad:

a) Análisis

Esta etapa se la realizó en dos puntos importantes, el análisis de requerimientos, el mismo que contendrá toda la especificación de requerimientos obtenidos gracias a los instrumentos de recopilación de información (entrevista y observación), y los diagramas propios de la fase, los mismo que establece una mejor comprensión de los requerimientos solicitados.

Análisis de requerimientos:

Para el análisis de requisitos se utilizará la especificación de requisitos de acuerdo con la plantilla IEEE, que es propicia para este proceso y es muy importante para la recopilación de información para desarrollar correctamente el módulo.

Introducción:

El presente apartado del documento, se dedica exclusivamente al análisis de los requerimientos obtenidos en el proceso de investigación y recopilación de información. El apartado considerará puntos importantes proporcionados por las normas IEEE 830 acorde al análisis y establecimiento de requerimientos. Todos los requerimientos detallados, corresponden al desarrollo del módulo de clasificación y predicción de Líneas y Sublíneas de investigación generados a través de minería de texto apoyados en algoritmos de Deep Learning propios de TensorFlow.

Propósito:

La documentación proporcionada es responsable de definir los requisitos funcionales y no funcionales para el diseño e implementación del nuevo módulo.

Alcance:

Está destinado para todos los docentes investigadores, y usuarios que interactúen directamente con el sistema Ecuciencia.

Personal involucrado:

En la tabla 5 se especifica el personal involucrado:

Tabla 5 Información de involucrados.

INFORMACIÓN DE INVOLUCRADO	
Nombre:	Diego Geovanny Falconí Punguil.
Rol:	Analista, diseñador y programador.
Categoría Profesional:	<ul style="list-style-type: none"> • Ingeniero en Informática y Sistemas Computacionales. • Estudiante de Maestría de Sistemas de Información.
Responsabilidad:	Analista de requerimientos, diseño y programación del nuevo módulo.
Información de contacto:	dfalconi0774.pos@utc.edu.ec

Elaborado por: Investigador

Definiciones, acrónimos y abreviaturas:

Tabla 6 Definiciones, acrónimos y abreviaturas

NOMBRE	DESCRIPCIÓN
INVESTIGADOR	Docente investigador con libros, revistas científicas publicadas.
USUARIO EXTERNO	Persona que externamente a la Universidad Técnica de Cotopaxi visualizará la información subida por el investigador.
UTC	Universidad Técnica de Cotopaxi.
ERS	Especificaciones de Requisitos de Software.
RF	Requisitos Funcionales.
RNF	Requisitos no Funcionales.

Elaborado por: Investigador

Referencias:

Tabla 7 Referencias

TÍTULO DEL DOCUMENTO	REFERENCIA
Standard IEEE 830-1998	IEEE

Elaborado por: Investigador

Resumen:

Este documento está dividido en dos partes, la primera parte está relacionada con la descripción general de las especificaciones del módulo y se introduce de acuerdo con los requisitos de los usuarios responsables del sistema Ecuciencia. La segunda está destinada hacia la definición detallada de los requerimientos para satisfacer las necesidades de los usuarios, esto se realizará por medio de los requerimientos funcionales y no funcionales.

DESCRIPCIÓN GENERAL

Perspectiva del producto:

El módulo de análisis y predicción de datos a través de algoritmos de Deep Learning, dentro de Ecuciencia generará un gran impacto, ya que apoyará a los usuarios que generan publicaciones a sugerir las sublínea y línea de investigación que está apuntando, esto producirá dos efectos importantes, la optimización de tiempo para que el investigador publique su artículo, y la restricción de que el usuario proponga esa información, puesto que muchas de las veces se lo realiza de manera errónea llenando así la base de datos con data inconsistente.

Funcionalidad del producto:

La funcionalidad principal del módulo es permitir a la plataforma analizar la información existente en la base datos, y a partir de ella generar conocimiento y mejorar su nivel de predicción.

Características de los usuarios:

En la tabla 8 se muestra las características de los usuarios:

Tabla 8 Usuarios

Usuarios	Actividades
Docente Investigador	Encargado de subir sus libros, revistas y artículos científicos.
Usuario externo.	Visualizar la información presentada por el sistema Ecuciencia.

Elaborado por: Investigador

DEFINICIÓN DE REQUERIMIENTOS

Los requerimientos considerados para esta propuesta se dividen en, los funcionales y no funcionales. Los requerimientos funcionales son aquellos que surgen a partir de los datos recabados en la entrevista, es decir aquellos que el usuario solicita se dé un mantenimiento. Los requerimientos no funcionales son aquellos que no necesariamente son solicitados por el usuario pero que se deben considerar al desarrollar un software, por ejemplo, el establecimiento de colores de la organización en la interfaz, aplicación de seguridades, entre otros. A continuación, se detallará cada uno de ellos.

Requerimientos Funcionales (RF):

A continuación, se detallan los requerimientos funcionales obtenidos a partir de la entrevista realizada al PhD. Gustavo Rodríguez. En las tablas 9, 10, 11, 12, 13 y 14 se observan los requerimientos con su detalle correspondiente:

Tabla 9 RF01

HISTORIA DE USUARIO			
Número:	1	Usuario:	Investigador
Nombre de la Historia:	Mejoramiento del proceso de predicción del sistema		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El Usuario requiere un módulo que mejore el proceso de predicción al momento de establecer una línea y sublínea de investigación cuando un nuevo documento científico esté por ingresar.		

Elaborado por: Investigador

Tabla 10 RF02

HISTORIA DE USUARIO			
Número:	2	Usuario:	Investigador
Nombre de la Historia:	Uso de algoritmos de aprendizaje profundo		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El Usuario propone utilizar algoritmos de aprendizaje profundo para mejorar el nivel de predicción y clasificación de datos.		

Elaborado por: Investigador

Tabla 11 RF03

HISTORIA DE USUARIO			
Número:	3	Usuario:	Investigador
Nombre de la Historia:	Análisis con minería de texto		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El Usuario considera que el sistema se debe encargar de realizar un análisis de texto para obtener una clasificación de líneas y sublíneas de investigación.		

Elaborado por: Investigador

Tabla 12 RF04

HISTORIA DE USUARIO			
Número:	4	Usuario:	Investigador
Nombre de la Historia:	Predicción de clasificación a nuevos datos ingresados		
Prioridad:	Alta		

Programador Responsable:	Diego Falconí
Descripción:	El Usuario manifiesta que cuando un investigador ingrese un nuevo documento científico, el sistema sea el encargado de sugerir a qué línea y sublínea pertenece.

Elaborado por: Investigador

Tabla 13 RF05

HISTORIA DE USUARIO			
Número:	5	Usuario:	Investigador
Nombre de la Historia:	Uso de TensorFlow		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El Usuario sugiere el uso de TensorFlow para la implementación del nuevo módulo.		

Elaborado por: Investigador

Tabla 14 RF06

HISTORIA DE USUARIO			
Número:	6	Usuario:	Investigador
Nombre de la Historia:	Visualización de resultados generados		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El Usuario requiere una visualización de resultados los mismo que se utilizarán para el análisis de la predicción efectuada.		

Elaborado por: Investigador

Requerimientos No Funcionales (RNF):

A continuación, se detallan los requerimientos no funcionales mismos que se encuentra descritos en las tablas 15, 16, 17, 18 y 19:

Tabla 15 RNF01

REQUERIMIENTO NO FUNCIONAL			
Número:	1	Usuario:	Investigador
Nombre de la Historia:	Interfaz del Sistema		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	Las interfaces del sistema deben respetar los colores propios de Ecuciencia. Además, deben ser sencillas y tener características responsive.		

Elaborado por: Investigador

Tabla 16 RNF02

REQUERIMIENTO NO FUNCIONAL			
Número:	2	Usuario:	Investigador
Nombre de la Historia:	Nivel de Eficiencia Alto		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El módulo debe tener un tiempo de respuesta considerablemente bajo, puesto que es de vital importancia en la navegabilidad dentro de internet.		

Elaborado por: Investigador

Tabla 17 RNF03

REQUERIMIENTO NO FUNCIONAL			
Número:	3	Usuario:	Investigador
Nombre de la Historia:	Restricción de usuarios		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	El módulo contendrá un acceso público para cualquier usuario.		

Elaborado por: Investigador

Tabla 18 RNF04

REQUERIMIENTO NO FUNCIONAL			
Número:	4	Usuario:	Investigador
Nombre de la Historia:	Confiabilidad		
Prioridad:	Alta		
Programador Responsable:	Diego Falconí		
Descripción:	Los resultados obtenidos en la predicción deben tener al menos el 95% de credibilidad para evitar problemas de confusión en futuros análisis.		

Elaborado por: Investigador

Tabla 19 RNF05

REQUERIMIENTO NO FUNCIONAL			
Número:	5	Usuario:	Investigador
Nombre de la Historia:	Seguridad		
Prioridad:	Alta		

REQUERIMIENTO NO FUNCIONAL	
Programador Responsable:	Diego Falconí
Descripción:	El sistema debe evitar lo más posible generar logs públicos los cuales pueden ser utilizados de forma irresponsable.

Elaborado por: Investigador

Diagramas de Análisis:

Los diagramas realizados en el proceso de análisis que se han considerado son:

- Casos de Uso.
- Diagramas de Actividades.
- Diagrama de Entidad - Relación.
- Diagrama de secuencia.

Los mismos se podrán visualizar en el Anexo II al final del presente documento.

b) Diseño

En la etapa de Diseño se realizó una interpretación de los diagramas expuestos en la etapa de Análisis. En la etapa de diseño se propuso el maquetado de las principales interfaces gráficas de usuario que serán habilitadas para la interacción con los Stakeholders, en el Anexo III se destinó un espacio para las maquetaciones mencionadas.

c) Implementación

Esta etapa es la más extensa del proceso de desarrollo de software, aquí se establece la codificación necesaria, en el Anexo IV, se puede visualizar parte del código desarrollado tanto en el lenguaje Python como en el Lenguaje JavaScript para la solución del problema. Adicionalmente en el Anexo V se muestran capturas de pantalla de las interfaces gráficas que componen el proceso y la comunicación del usuario final.

d) Pruebas

En esta etapa, los resultados serán verificados de acuerdo con los requerimientos tomados en las etapas anteriores. Para la verificación, es necesario utilizar una matriz de prueba, que se puede ver en el Anexo VI. Este trabajo lo debe realizar la persona encargada del test del sistema.

3.3. RESULTADOS DEL DISEÑO EXPERIMENTAL Y/O MÉTODO DE CRITERIO DE EXPERTOS

3.3.1. MÉTODO DE VALIDACIÓN CRUZADA

Como se mencionó anteriormente, el método utilizado para la validación de la presente propuesta es el Cross-Validation, y se separó por etapas, a continuación, se muestran los resultados de cada una de ellas:

Etapa 1: En esta se realiza una extracción de los datos pertenecientes a documentos científicos ligados a la carrera del Ingeniería en Sistemas de Información, para ello se emplea el Query disponible en el Anexo VII, en el mismo que también se puede apreciar el resultado de la consulta extraída.

Etapa 2: En esta etapa se realizó la primera validación utilizando el método “hold-out”, para ellos el 80% del total de datos se los destinó para entrenamiento y el 20% restante se lo utilizó para los test del modelo. En la figura 32 se muestra el resultado de la validación obtenida a través de la consola de Python, en el cual se aprecia un porcentaje de exactitud del 94.44%.

```
Epoch 10/25
8/8 [=====] - 0s 6ms/step - loss: 0.5570 - categorical_accuracy: 0.6328 - val_loss: 0.5505 - val_categorical_accuracy: 0.8276
Epoch 11/25
8/8 [=====] - 0s 4ms/step - loss: 0.5289 - categorical_accuracy: 0.6133 - val_loss: 0.5354 - val_categorical_accuracy: 0.8621
Epoch 12/25
8/8 [=====] - 0s 4ms/step - loss: 0.5129 - categorical_accuracy: 0.6758 - val_loss: 0.5178 - val_categorical_accuracy: 0.8966
Epoch 13/25
8/8 [=====] - 0s 5ms/step - loss: 0.5078 - categorical_accuracy: 0.6484 - val_loss: 0.5005 - val_categorical_accuracy: 0.8966
Epoch 14/25
8/8 [=====] - 0s 6ms/step - loss: 0.4581 - categorical_accuracy: 0.7500 - val_loss: 0.4819 - val_categorical_accuracy: 0.8966
Epoch 15/25
8/8 [=====] - 0s 5ms/step - loss: 0.4549 - categorical_accuracy: 0.7734 - val_loss: 0.4566 - val_categorical_accuracy: 0.8966
Epoch 16/25
8/8 [=====] - 0s 5ms/step - loss: 0.4359 - categorical_accuracy: 0.7188 - val_loss: 0.4289 - val_categorical_accuracy: 0.8966
Epoch 17/25
8/8 [=====] - 0s 5ms/step - loss: 0.4019 - categorical_accuracy: 0.7930 - val_loss: 0.4017 - val_categorical_accuracy: 0.8966
Epoch 18/25
8/8 [=====] - 0s 5ms/step - loss: 0.3838 - categorical_accuracy: 0.7656 - val_loss: 0.3713 - val_categorical_accuracy: 0.8966
Epoch 19/25
8/8 [=====] - 0s 5ms/step - loss: 0.3502 - categorical_accuracy: 0.8359 - val_loss: 0.3394 - val_categorical_accuracy: 0.8966
Epoch 20/25
8/8 [=====] - 0s 5ms/step - loss: 0.3450 - categorical_accuracy: 0.8164 - val_loss: 0.3100 - val_categorical_accuracy: 0.9310
Epoch 21/25
8/8 [=====] - 0s 5ms/step - loss: 0.3347 - categorical_accuracy: 0.7617 - val_loss: 0.2845 - val_categorical_accuracy: 0.9655
Epoch 22/25
8/8 [=====] - 0s 4ms/step - loss: 0.3308 - categorical_accuracy: 0.8086 - val_loss: 0.2623 - val_categorical_accuracy: 0.9655
Epoch 23/25
8/8 [=====] - 0s 5ms/step - loss: 0.2772 - categorical_accuracy: 0.8555 - val_loss: 0.2401 - val_categorical_accuracy: 0.9655
Epoch 24/25
8/8 [=====] - 0s 6ms/step - loss: 0.2701 - categorical_accuracy: 0.8359 - val_loss: 0.2186 - val_categorical_accuracy: 0.9655
Epoch 25/25
8/8 [=====] - 0s 5ms/step - loss: 0.2691 - categorical_accuracy: 0.8438 - val_loss: 0.2027 - val_categorical_accuracy: 0.9655
3/3 [=====] - 0s 2ms/step - loss: 0.2055 - categorical_accuracy: 0.9444
```

Figura 32 Resultado de validación “hold-out”

Fuente: Investigador

Etapa 3: En esta etapa se consideró el uso del método “k-fold”, para ello se estableció un k de 25 iteraciones, en la cual recopilará los datos de nivel de exactitud por cada iteración y representará un promedio general del porcentaje de exactitud. En el Anexo VIII se muestran los logs resultantes del análisis, en donde se aprecian los puntajes de cada iteración y el promedio general de todas las iteraciones. Adicionalmente en la figura 33 se muestran los resultados tabulados del análisis obtenido con el método “k-fold”.

Epocas	Datos
1	93,33
2	93,33
3	93,33
4	93,33
5	100
6	100
7	93,33
8	100
9	100
10	100
11	92,86
12	100
13	92,86
14	100
15	100
16	92,86
17	100
18	100
19	92,86
20	100
21	100
22	100
23	92,86
24	100
25	100

Figura 33 Resultado de validación “k-fold” con k = 25

Fuente: Investigador

Etapa 4: Finalmente para tener una representación visual de los resultados, se realizó un diagrama general de los datos obtenidos en cada una de las iteraciones. En la figura 34 se muestra los resultados del método “hold-out” y en la figura 35 se puede visualizar el resumen del método “k-fold”.

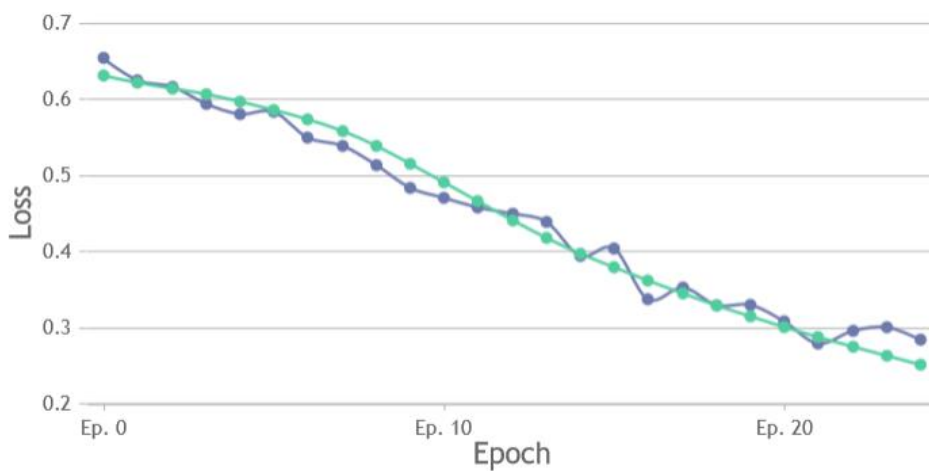


Figura 34 Representación visual del método “hold-out”

Fuente: Investigador

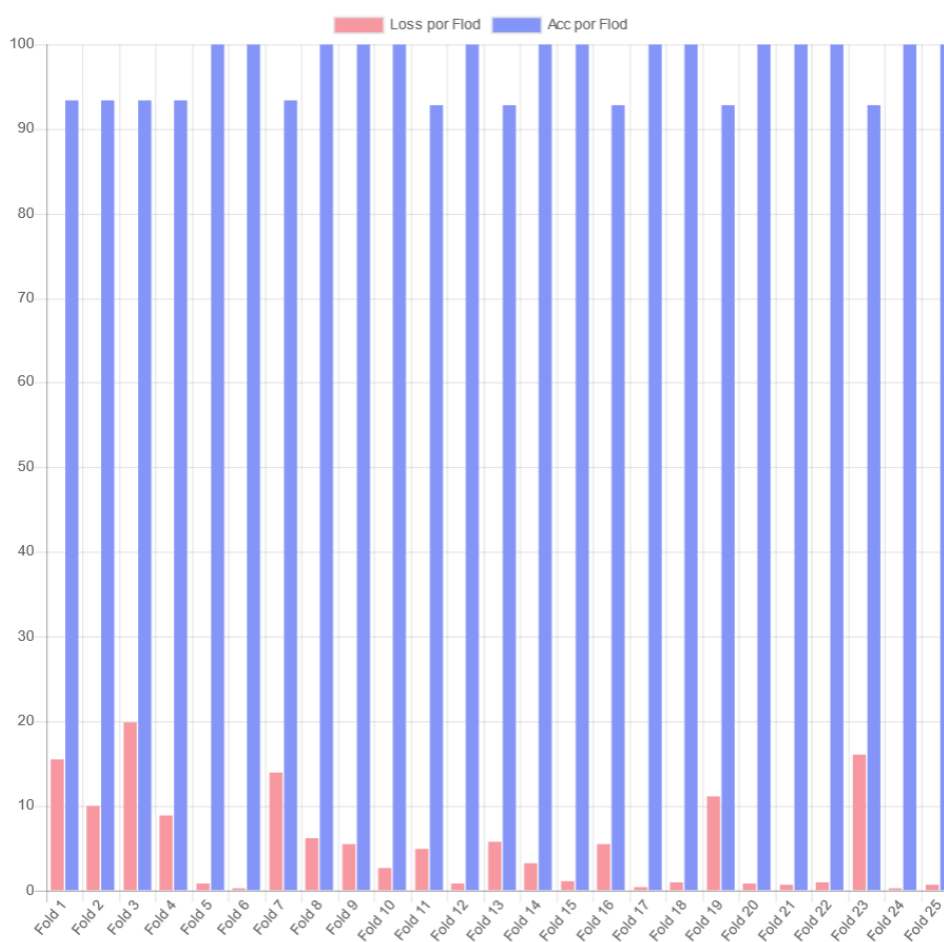


Figura 35 Representación visual del método “k-fold”

Fuente: Investigador

Interpretación: Gracias a los datos obtenidos y representados en la figura 33, se puede emplear diferentes métodos de interpretación de resultados. Uno de los más conocidos, y compatibles con el método “Cross Validation” es el de desviación estándar. En la figura 36 se muestra la tabulación de los datos con sus respectivos cálculos de Promedio y Desviación Estándar:

Epocas	Datos	Promedio	Limite Max	Limite Min
1	93,33	97,238	100,693831	93,7821692
2	93,33	97,238	100,693831	93,7821692
3	93,33	97,238	100,693831	93,7821692
4	93,33	97,238	100,693831	93,7821692
5	100	97,238	100,693831	93,7821692
6	100	97,238	100,693831	93,7821692
7	93,33	97,238	100,693831	93,7821692
8	100	97,238	100,693831	93,7821692
9	100	97,238	100,693831	93,7821692
10	100	97,238	100,693831	93,7821692
11	92,86	97,238	100,693831	93,7821692
12	100	97,238	100,693831	93,7821692
13	92,86	97,238	100,693831	93,7821692
14	100	97,238	100,693831	93,7821692
15	100	97,238	100,693831	93,7821692
16	92,86	97,238	100,693831	93,7821692
17	100	97,238	100,693831	93,7821692
18	100	97,238	100,693831	93,7821692
19	92,86	97,238	100,693831	93,7821692
20	100	97,238	100,693831	93,7821692
21	100	97,238	100,693831	93,7821692
22	100	97,238	100,693831	93,7821692
23	92,86	97,238	100,693831	93,7821692
24	100	97,238	100,693831	93,7821692
25	100	97,238	100,693831	93,7821692

Promedio	97,238
Desviacion	3,45583082
Limite Max	100,693831
Limite Min	93,7821692

Figura 36 Cálculo de Desviación Estándar

Fuente: Investigador

El resultado se lo representa de manera visual en la figura 37:

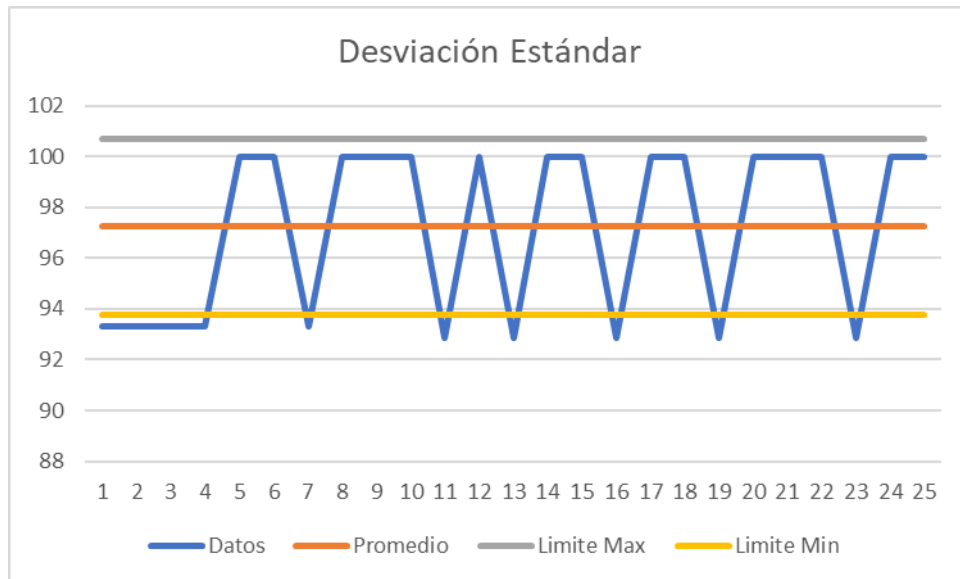


Figura 37 Gráfico de Desviación Estándar

Fuente: Investigador

La desviación estándar es la medida más común de dispersión, representa el grado de distribución entre los datos y el promedio. Cuanto mayor sea la desviación estándar, mayor será la dispersión de los datos. Como se muestra en la figura 37, los resultados de la validación utilizando el método “Cross Validation”, en su mayoría se encuentra ubicados dentro del rango establecidos entre el límite superior e inferior de la desviación estándar, lo cual representa la estabilidad de los resultados de las pruebas realizadas.

Con estos datos también es posible realizar la prueba de Distribución Normal; En la figura 38 se puede apreciar el resultado del cálculo de Distribución Normal:

Segmentos	Distribucion
93	0,05442411
93,5	0,06431355
94	0,07442561
94,5	0,08434342
95	0,09360279
95,5	0,10172677
96	0,10826562
96,5	0,11283782
97	0,1151669
97,5	0,11510905
98	0,11266788
98,5	0,107994
99	0,10136965
99,5	0,09318052
100	0,08387859
100,5	0,07394111
101	0,06383071

Figura 38 Cálculo de Distribución Normal

Fuente: Investigador

En la figura 39 se representa gráficamente los resultados obtenidos en la figura 38.

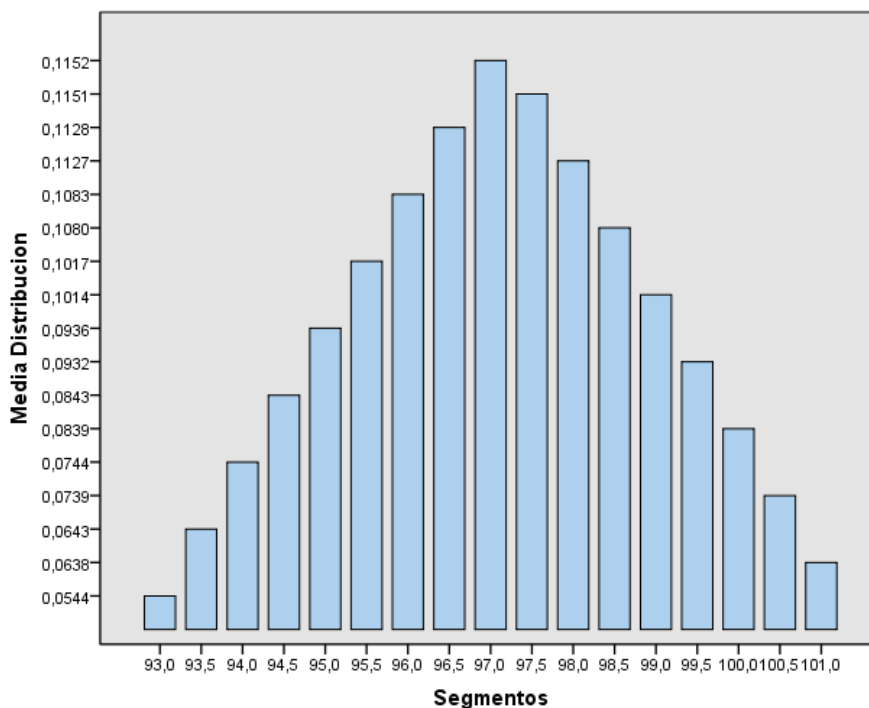


Figura 39 Representación de Distribución Normal

Fuente: Investigador

En la figura 40 se muestra la Campana de Gauss trazada en la herramienta GeoGebra.

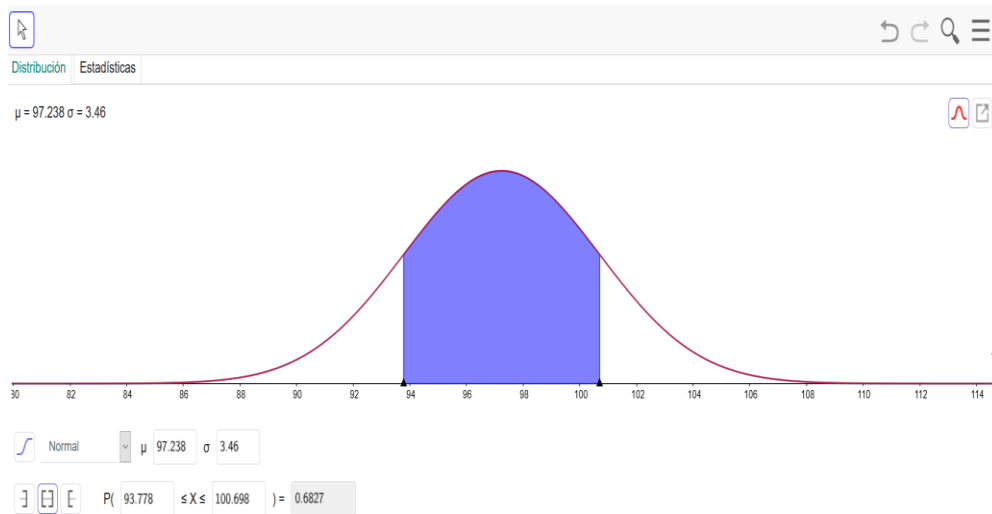


Figura 40 Campana de Gauss

Fuente: Investigador

Como se puede apreciar en la figura 40 existe un 68% de probabilidades de que, en pruebas futuras, el resultado supere el 94% de porcentaje de exactitud. En la figura 41 se puede visualizar que casi el 95% de las pruebas realizadas, probablemente tendrán un valor superior al 90%.

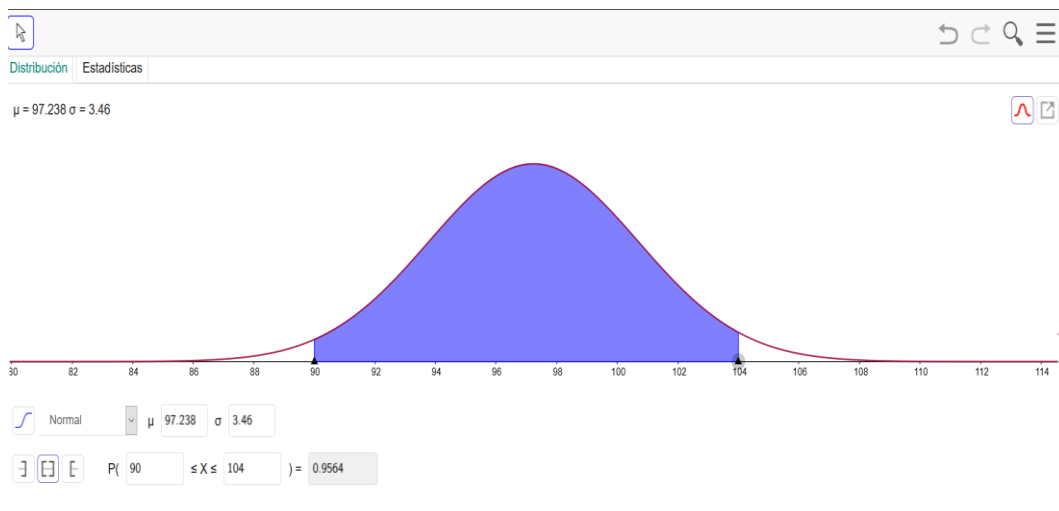


Figura 41 Probabilidad de que el resultado sea mayor al 90%

Fuente: Investigador

En base a estas validaciones, si se realiza una ponderación utilizando los valores establecidos en la tabla 3, la validación de la hipótesis se lo catalogaría en un nivel “Alto”, lo cual prueba que la implementación de la presente propuesta es aceptada.

3.4. RESULTADOS DE LA VALORIZACIÓN ECONÓMICA, TECNOLÓGICA, OPERACIONAL Y AMBIENTAL

3.4.1. VALORACIÓN ECONÓMICA:

Para la valoración económica de la presente propuesta, se considerarán todos los aspectos importantes que se llevaron a cargo en el proceso del desarrollo. Los gastos producidos se los dividirán en dos, los directos, y los indirectos, para posteriormente establecer una sumatoria total lo cual representará el costo total del proyecto.

a) Gastos Directos

Tabla 20 Gastos Directos

GASTOS DIRECTOS				
DESCRIPCIÓN	DETALLE	CANTIDAD / HORAS	PRECIO	TOTAL
Desarrollo de Software	Iteración I	100	15,00	1500,00
	Iteración II	100	15,00	1500,00
	Iteración III	100	15,00	1500,00
	Iteración IV	100	15,00	1500,00
Equipos Tecnológicos	Computador	1	1500,00	1500,00
	Mejoras en equipos	1	500,00	500,00
	Servicio de Internet	320	0,60	192,00
Insumos de Oficina	Impresiones	400	0,10	40,00
	Resma de papel	2	3,50	7,00
	Carpetas	5	0,75	3,75
	Anillados	5	0,80	4,00
<i>Elaborado por: Investigador</i>			Total	8246,75

b) Gastos Indirectos

Tabla 21 Gastos Indirectos

GASTOS INDIRECTOS				
DESCRIPCIÓN	DETALLE	CANTIDAD / HORAS	PRECIO	TOTAL
Otros Gastos	Transporte	10	1,50	15,00
	Alimentación	10	2,00	20,00
	Imprevistos	1	100,00	100,00
<i>Elaborado por: Investigador</i>				Total 135,00

c) Gastos Totales

Tabla 22 Gastos Totales

COSTO TOTAL DEL PROYECTO	
DETALLE	TOTAL
Gastos Directos	8246,75
Gastos Indirectos	135,00
TOTAL	8.381,75

Elaborado por: Investigador

Como se puede visualizar en las tablas 20, 21 y 22 el costo total de la propuesta tendría un valor aproximado de 8.381,75 dólares. Este valor es un estimado, sin embargo, es probable que el valor ascienda debido a las horas utilizadas para la investigación y test de funcionamiento.

3.4.2. VALORACIÓN TECNOLÓGICA:

Para poder dar una valoración de cuáles son los requisitos mínimos de Hardware y Software para el correcto funcionamiento del módulo, se considerará como referencia el equipo en el cual fue implementado. Las características serán representadas en la tabla 23 y 24.

a) Requisitos mínimos de Hardware

Tabla 23 Requisitos Mínimos de Hardware.

REQUISITOS MÍNIMOS DE HARDWARE	
Procesador	1.4 GHz
Arquitectura	64 bits
Memoria RAM	6 GB
Espacio en Disco	512 GB
Ethernet	Adaptador de Ethernet de 10/100 base T Gigabit

Elaborado por: Investigador

b) Requisitos mínimos de Sistema Operativo (Software)

Tabla 24 Requisitos Mínimos de Hardware.

REQUISITOS MÍNIMOS DE SISTEMA OPERATIVO	
Sistema Operativo	Windows 8
Versión de Python	3.6
Versión de Django	1.11
Versión de TensorFlow	2.0

Elaborado por: Investigador

3.4.3. VALORACIÓN AMBIENTAL:

Gracias a que Ecuciencia lleva el registro de Documentos Científicos de manera digital, esto implica que aporta con la disminución de hojas de impresión, lo cual ayuda significativamente en el tema ambiental. Adicionalmente cabe recalcar que al ser un sistema Web, cualquier usuario podrá acceder a su información desde cualquier lugar, esto implica que no necesitarían descargar e imprimir ningún archivo para tenerlo a la mano.

3.5. DISCUSIÓN DE LA APLICACIÓN Y VALIDACIÓN DE LA PROPUESTA

La investigación planteada al inicio de la propuesta tiene como objetivo implementar funciones basadas en la tecnología que actualmente se investiga y se pone en práctica en todo el mundo. Hoy en día uno la aplicación de algoritmos de Inteligencia Artificial se ha ido incrementando en gran medida en los grandes sistemas de información alrededor del mundo, por ende, también se lo consideró como parte de la plataforma Web Ecuciencia. Con el apoyo de TensorFlow, el cual nos ofrece un catálogo amplio de algoritmos de aprendizaje profundo, se ha desarrollado un nuevo módulo que permita la clasificación y predicción de líneas y sublíneas de investigación a nuevas publicaciones que se vayan a registrar en el sistema.

En el transcurso de la investigación se pudo notar el alto nivel de compatibilidad existente entre las metodologías KDD y la del modelo Iterativo e Incremental, es por esta razón que se optó por generar un híbrido para así crear un nuevo marco de trabajo que puede ser utilizado para trabajos futuros, el mismo está especificado en la figura 30.

En cuanto a la implementación de la solución de la propuesta, se pudo generar una combinación de diferentes algoritmos para el trabajo en conjunto, esta sociedad hizo posible el perfeccionamiento del nivel de predicción y entrenamiento de los datos existentes en Ecuciencia.

Es importante señalar que el apoyo de TensorFlow para el desarrollo de esta propuesta fue de vital importancia, puesto que, gracias a su arquitectura, fue totalmente compatible con la estructura actual de Ecuciencia. Además, durante el proceso de validación de resultados, se pudo evidenciar que los tiempos que le toma a los algoritmos en el entrenamiento fue relativamente bajo, lo cual garantiza tener una eficiencia al momento de entrenar el modelo. Por otra parte, también se pudo apreciar, que, al momento de dar una predicción, lo hizo de una manera rápida y precisa lo cual garantiza que sus predicciones tienen un margen de error muy reducido, generando así confianza en ejecuciones futuras.

Además, gracias al método Cross-Validation, se pudo demostrar la eficiencia y el alto nivel de exactitud del entrenamiento de los algoritmos seleccionados para resolver la problemática. Esto demuestra que la aplicación de algoritmos de Deep Learning utilizando TensorFlow, mejorará significativamente el nivel del tratamiento de datos de la producción científica correspondiente a la relación entre documentos científicos y líneas de investigación de la Universidad Técnica de Cotopaxi.

3.6. CONCLUSIONES CAPÍTULO III

- El uso de la metodología KDD, y la metodología del modelo Iterativo e Incremental, jugaron un papel muy importante para la resolución de la propuesta. Incluso se pudo realizar una combinación efectiva de estas dos metodologías para efectuar el desarrollo de software y minería de datos.
- El proceso de entrenamiento de los algoritmos de aprendizaje profundo resultó sencillo gracias a la versatilidad de TensorFlow, puesto que, con su experiencia, aportó al entrenamiento de manera eficiente y eficaz.
- El uso del método Cross-Validation, aportó significativamente a la validación de la propuesta, ya que, en la serie de test realizados sobre el conjunto de entrenamiento, ratifico un alto grado de exactitud, estableciendo un promedio de sobre el 90% en casi todas sus iteraciones de pruebas.

CONCLUSIONES GENERALES

- La Cienciometría aporta significativamente en el crecimiento de sistemas de información de producción científica con alto nivel de conocimiento puesto que la misma, regula en base a estándares de evaluación para asegurar la calidad de publicaciones científicas.
- En los últimos tres años, se han realizado diversas publicaciones de artículos y tesis que han centrado su atención en Ecuciencia, dichas investigaciones muestran claramente la importancia del análisis de datos aplicando diversos algoritmos de Inteligencia Artificial, sin embargo, también se evidenció la falta de una herramienta de aprendizaje profundo para mejorar la calidad de procesamiento de información en base a clasificación y predicción.
- Existen diversas metodologías orientadas a minería de datos e Inteligencia Artificial, sin embargo, la que se considera la más completa es precisamente la metodología KDD, la misma que guía al investigador a través de varias etapas partiendo desde la recolección de datos hasta la producción del conocimiento, utilizando técnicas de filtrado y limpieza de información innecesaria.
- Gracias al método Cross-Validation, el cual brinda un lineamiento para el proceso de validación de algoritmos de inteligencia artificial, se pudo apreciar la factibilidad de ejecución de la presente propuesta al obtener una calificación de sobre el 90% de exactitud, lo cual, en términos cualitativos, se lo cataloga dentro de un rango relativamente alto.

RECOMENDACIONES

- Es importante tener en cuenta la importancia que se le debe dar a la consistencia de datos antes de realizar un proceso de entrenamiento dentro de un algoritmo de inteligencia artificial para garantizar la calidad de predicción que se puede obtener.
- Se recomienda emplear un recurso humano para la limpieza de datos para el entrenamiento inicial del algoritmo, puesto que así se garantizará mayor veracidad en los datos obtenidos.
- Si se considera pertinente, se podría emplear el híbrido entre la metodología KDD y la del modelo Iterativo e Incremental, para la resolución de trabajos futuros con contextos similares al presente.
- Se sugiere aprovechar al máximo el uso de TensorFlow en la producción científica de Ecuciencia, debido a que es sencillo de comprender y tiene un amplio catálogo de funcionalidades aún por descubrir.
- Se invita a los futuros investigadores de Ecuciencia a que se dé continuidad al módulo presente para ampliar las funcionalidades del mismo.

REFERENCIAS BIBLIOGRÁFICAS

- [1] UAM_Biblioteca, “Producción científica: Producción Científica de la UAM,” 2018. [Online]. Available: https://biblioguias.uam.es/produccion_cientifica.
- [2] E. Ayala Mora, “La investigación científica en las universidades ecuatorianas,” *Anales. Rev. la Univ. Cuenca*, vol. 3, no. 57, pp. 61–72, 2015.
- [3] C. G. Rivera García, J. M. Espinosa Manfugás, and Y. D. Valdés Bencomo, “La investigación científica en las universidades ecuatorianas. Prioridad del sistema educativo vigente,” *Rev. Cuba. Educ. Super.*, vol. 36, no. 2, pp. 113–125, 2017.
- [4] Scimago Institutions Rankings, “SIR liber 2015, Rank output 2009-2013,” *Scopus*, 2010. [Online]. Available: <https://www.scimagoir.com/>.
- [5] M. del P. Fernández Díaz, S. Martínez Bernal, C. Rivalta Bermúdez, M. Díaz Ríos, and G. Jiménez Santander, “Repositorio de búsquedas y recuperación de la información científica en ciencias de la salud,” *EDUMECENTRO*, vol. 5, no. 2, pp. 198–211, 2013.
- [6] R. Cañedo Andalia and A. J. Dorta Contreras, “SCImago Journal & Country Rank, una plataforma para la evaluación del comportamiento de la ciencia según fuentes documentales y países,” *ACIMED*, vol. 21, no. 3, pp. 310–320, 2010.
- [7] “SCImago,” *Form. Univ.*, vol. 5, no. 5, pp. 1–1, 2012.
- [8] SciELO - Scientific Electronic Library Online, “SciELO.” [Online]. Available: <http://www.scielo.org.ve/>. [Accessed: Feb-2020].
- [9] T. Dalglish *et al.*, *Open Access Indicators and Scholarly Communications in Latin America*, 1a ed. Buenos Aires: CLACSO, 2014.
- [10] C. Canales Bojo, “La red SciELO (Scientific Electronic Library Online): perspectiva tras 20 años de funcionamiento,” *HAD*, vol. 1, no. 4, pp. 211–220, 2017.

- [11] A. Ochoa Contreras, A. Muñoz García, and H. Morales López, “Perspectivas de la Bibliometría en las Ciencias Médicas,” *Arch. en Med. Fam.*, vol. 17, no. 1, pp. 1–3, 2016.
- [12] I. De la Vega, “El uso de la cienciometría en la construcción de las políticas tecnocientíficas en américa latina: una relación incierta,” *Redes*, vol. 15, no. 29, pp. 217–240, 2009.
- [13] E. Ortiz Torres, M. V. Gonzáles Guitián, C. González Calzadilla, and I. Infante Pérez, “INDICADORES PARA EVALUAR EL IMPACTO CIENTÍFICO DE LAS TESIS DOCTORALES EN CIENCIAS PEDAGÓGICAS,” *Rev. Pedagog. Univ.*, vol. 14, no. 2, pp. 81–89, 2009.
- [14] E. Spinak, “Indicadores cienciométricos,” *ACIMED*, vol. 9, no. 4, pp. 16–18, 2001.
- [15] M. V. González Guitián and M. Molina Piñeiro, “LA EVALUACIÓN DE LA CIENCIA: REVISIÓN DE SUS INDICADORES,” *Eumed.net*, 2009. [Online]. Available: <http://www.eumed.net/rev/cccss/06/ggmp.htm>. [Accessed: Feb-2020].
- [16] DATACIENCIA- Dimensiones de la Producción Científica Nacional, “Programa de Información Científica CONICYT.” [Online]. Available: <https://dataciencia.conicyt.cl/interfaz/>. [Accessed: Feb-2020].
- [17] RedCiencia, “Información| Redciencia.” [Online]. Available: <http://www.redciencia.net/contact>. [Accessed: Feb-2020].
- [18] REDSEARCH, “REDSEARCH - AYUDA.” [Online]. Available: <https://redsearch.conicyt.cl/help.php>. [Accessed: Feb-2020].
- [19] Redalyc.org, “Acerca de Redalyc.org,” 2017. [Online]. Available: http://www.redalyc.org/redalyc/media/redalyc_n/Estaticas3/mision.html. [Accessed: Feb-2020].
- [20] J. Moreno Cevallos y B. Dueñas Holguín, «Sistemas de información empresarial: la información como recurso estratégico,» *Revista Científica Dominio de las Ciencias*, vol. 4, nº 1, pp. 141 - 154, 2018.

- [21] R. Andreu, J. Ricart y J. Valor, *Estrategia y sistemas de información*. Madrid: Mc Graw Hill, 1991.
- [22] S. Havre, B. Hetzler y L. Nowell, «ThemeRiver: In Search of Trends, Patterns, and Relationships,» *Inu*, vol. 1, p. 4, 2015.
- [23] E. G. Aguilar Riera y D. A. Davila Garzon, “Análisis, diseño e implementación de la aplicación web para el manejo del distributivo de la Facultad de Ingeniería”. 2013. [En línea]. Available: <http://dspace.ucuenca.edu.ec/bitstream/123456789/4303/1/tesis.pdf>.
- [24] EUATM, “Introducción a la Web”. [En línea]. Available: <http://www.edificacion.upm.es/informatica/documentos/www.pdf>.
- [25] M. Márquez, “Base de datos”. 2009. [En línea]. Available: http://www3.uji.es/~mmarques/apuntes_bbdd/apuntes.pdf.
- [26] Borgatti, S.P., Everett, M.G., & Freeman, L.C. (2002). *UCINET 6 for Windows: Software for Social Network Analysis*. Harvard, MA, Analytic Technologies.
- [27] PostgreSQL, “PostgreSQL: About.” [Online]. Available: <https://www.postgresql.org/about/>.
- [28] L. Joyanes Aguilar, *Metodología de la programación diagramas de flujo, algoritmos y programación estructurada*. Madrid: McGraw-Hill, Interamericana de España, 1988.
- [29] E. G. Aguilar Riera y D. A. Dávila Garzón, «Análisis, diseño e implementación de la aplicación web para el manejo del distributivo de la Facultad de Ingeniería,» 2013. [En línea]. Available: <http://dspace.ucuenca.edu.ec/bitstream/123456789/4303/1/tesis.pdf>.
- [30] I. 12207, «ISO/IEC 12207,» 2008.
- [31] E. «ECURED,» ECURED, 2020. [En línea]. Available: [https://www.ecured.cu/Lenguaje_de_programaci%C3%B3n_\(inform%C3%A1tica\)](https://www.ecured.cu/Lenguaje_de_programaci%C3%B3n_(inform%C3%A1tica)).

- [32] Challenger-Peréz Ivet, Y. Díaz-Ricardo, and R. A. Becerra-García, “El lenguaje de programación Python/The programming language Python,” *Ciencias Holguín*, vol. 20, no. 2, pp. 1–12, Apr. 2014.
- [33] M. Rocha and P. G. Ferreira, *Bioinformatics algorithms: design and Implementation in Python*. Portugal: Academic Press, 2018.
- [34] G. Rossum van, “El tutorial de Python,” Argentina, 2017.
- [35] Python Software Foundation (PSF), “Applications for Python,” 2019.
- [36] A. Vara Serrano, “PREDICCIÓN DE VISITAS MEDIANTE GEOLOCALIZACIÓN A TRAVÉS DE DISPOSITIVOS MÓVILES,” Universidad de Barcelona, 2017.
- [37] Wes McKinney & PyData Development Team, “pandas: powerful Python data analysis toolkit Release 0.23.4 Wes McKinney & PyData Development Team,” 2018.
- [38] N. Aguilera, *Matemáticas y programación con Python*. 2014.
- [39] S. van der Walt, S. C. Colbert, and G. Varoquaux, “The NumPy Array: A Structure for Efficient Numerical Computation,” *Comput. Sci. Eng.*, vol. 13, no. 2, pp. 22–30, Mar. 2011.
- [40] Django, “Meet Django,” 2018. [Online]. Available: <https://www.djangoproject.com/>.
- [41] J. Molina Ríos, N. Loja Mora, P. Zea Ordoñez y E. Loaiza Sojos, «Evaluación de los Frameworks en el desarrollo de Aplicaciones Web con Python,» *UNLA*, vol. 4, n° 4, 2016.
- [42] A. Holovaty y J. Kaplan-Moss, Holovaty, Adrian; Kaplan-Moss, Jacob. *The definitive guide to Django: Web development done right*, Apress, 2009.
- [43] L. J. Ayala Condori, “Phython – DjangoFramework de desarrollo web para perfeccionistas basado en el Modelo MTV,” *Rev. Inf. Tecnol. y Soc.*, no. 7, pp. 36–37, 2012.

- [44] R. Mata, “Minería de datos: qué es, cómo es el proceso y a qué áreas se puede aplicar - ICEMD,” *ICEMD*, 2017. [Online]. Available: <https://www.icemd.com/digitalknowledge/articulos/mineria-datos-proceso-areas-se-puede-aplica/>.
- [45] H. F. Vallejo Ballesteros, E. Guevara Iñiguez, and S. R. Medina Velasco, “Minería de Datos,” *Recimundo*, vol. 2, pp. 339–349, 2018.
- [46] S. Vallejos, “Minería de Datos,” Universidad Nacional del Nordeste, 2006.
- [47] E. Botta-Ferret y J. Cabrera-Gato, «Minería de textos: una herramienta útil para mejorar la gestión del bibliotecario en el entorno digital,» Acimed, 2007.
- [48] J. M. Molina López and J. García Herrero, “TÉCNICAS DE ANÁLISIS DE DATOS APLICACIONES PRÁCTICAS UTILIZANDO MICROSOFT EXCEL Y WEKA,” Universidad Carlos III. Madrid, 2006.
- [49] Y. Aranda Robles and A. R. Sotolongo, “Integración de los algoritmos de minería de datos 1R, PRISM E ID3 A POSTGRESQL,” *J. Inf. Syst. Technol. Manag.*, vol. 10, no. 2, pp. 389–406, Aug. 2013.
- [50] I. Guzmán Raja, «Guzmán Raja, Isidoro, et al. Predicción de resultados empresariales versus medidas no paramétricas de eficiencia técnica: evidencia para pymes de la Región de Murcia,» Murcia, 2005.
- [51] J. Hernández Cáceres, “Clustering technique based on k- means algorithm for the identification of clusters of surgical patients,” *Univ. St. Tomás, Secc. Bucaramanga*, pp. 1–8, 2016.
- [52] F. J. Pinales Delgado and C. E. Velázquez Amador, *Algoritmos resueltos con Diagramas de Flujo y Pseudocódigo*. México: Universidad Autónoma de Aguascalientes, 2018.
- [53] L. Joyanes Aguilar, *Metodología de la programación diagramas de flujo, algoritmos y programación estructurada*. Madrid: McGraw-Hill, Interamericana de España, 1988.

- [54] L. I. Roque Montalvo, “Análisis comparativo de técnicas de minería de datos para la predicción de ventas,” Universidad Señor de Sipán, 2016.
- [55] X. M. Martín Uriz and M. Galar Idoate, “Aprendizaje de distancias basadas en disimilitudes para el algoritmo de clasificación kNN,” Universidad Pública de Navarra, 2015.
- [56] M. Alaminos, A. Campos Sánchez, M. D. Caracuel, A. Rodríguez Morata, I. A. Rodríguez, «Modelos didácticos para el autoaprendizaje,» *Actual Med*, 2009.
- [57] J. Amaya, «Toma de decisiones gerenciales: métodos cuantitativos para la administración,» Ecoe ediciones, 2010.
- [58] G. Sotolongo Aguilar, M. V. Guzmán Sánchez y H. Carrillo, «VIBLIOSOM: Visualización de Información Bibliométrica mediante el Mapeo Autoorganizado,» *Redalyc*, 2011.
- [59] J. J. Montaña Moreno, «Redes Neuronales Artificiales aplicadas al Análisis de Datos,» *Scielo*, vol. 1, nº 1, p. 315, 2002.
- [60] L. Rouhiainen, “Inteligencia Artificial, 101 cosas que debes saber hoy sobre nuestro futuro”, Editorial Planeta S.A., Barcelona, 2018.
- [61] M. Abidabi, P. Barham, J. Chen y Z. Chen, TensorFlow: A system for large-scale machine learning, USENIX, 2016.
- [62] M. Abadi et al., “TensorFlow: A System for Large-Scale Machine Learning TensorFlow: A system for large-scale machine learning,” in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), 2016, pp. 265–284.
- [63] G. Rodríguez Bárcenas, “Red de Inteligencia Compartida Organizacional como soporte a la toma de decisiones”, Granada, 2013.
- [64] G. Rodríguez Bárcenas, REDEC, utc.edu.ec, 2020, [online]. Available at: <http://www.utc.edu.ec/INVESTIGACION/PROYECTOS-EJECUCION/REDEC>.

- [65] J. Allauca y E. Chicaiza, “Aplicación de algoritmo de extracción de Texto en Perfiles de Usuario en caso de los investigadores de la Universidad Técnica De Cotopaxi”, Universidad Técnica de Cotopaxi, 2019 [Online]. Available: <http://repositorio.utc.edu.ec/handle/27000/5752>.
- [66] D. Falconí y J. Gualpa, “Método para determinar la Similitud y Distancia entre investigadores a partir de Algoritmos de Clasificación”, Universidad Técnica de Cotopaxi, 2019 [Online]. Available: <http://repositorio.utc.edu.ec/handle/27000/5698>.
- [67] A. Rivera, “Visualización de Información mediante mapeo auto-organizado en datos de producción científica de la Universidad Técnica de Cotopaxi”, Universidad Técnica de Cotopaxi, 2020.
- [68] E. Chango, “Método de Análisis de Redes Sociales para identificar relaciones y colaboraciones científicas entre investigadores de la Universidad Técnica de Cotopaxi”, Universidad Técnica de Cotopaxi, 2020.
- [69] I. Timarán-Pereira, S. R. Hernández-Arteaga, S. J. Caicedo-Zambrano, A. Hidalgo- Troya, and J. C. Alvarado- Pérez, “El proceso de descubrimiento de conocimiento en bases de datos.,” in *Descubrimiento de patrones de desempeño académico con árboles de decisión en las competencias genéricas de la formación profesional*, Bogotá: Ediciones Universidad Cooperativa de Colombia, 2016, pp. 63–86.
- [70] R. Brito Sarasa, A. Rosete Suárez, and R. Acosta Sánchez, “Desarrollo de un proceso de KDD en el ámbito docente: Preparación de los datos,” *CUAJAE*, pp. 2–7, 2008.
- [71] D. Ramos, R. Noriega, J. R. Láñez y A. Durango, *Curso de Ingeniería de Software: 2ª Edición.*, IT Campus Academy, 2017.
- [72] S. Paloma, A. Bernal, T. Rodríguez. *Desarrollo Iterativo e Incremental*, SENA, 2014.
- [73] C. E. Peralta Ríos, E. Villa Aburto, M. J. Pozas Cárdenas, and A. Curiel Anaya, “Ciclo de vida del desarrollo de sistemas de realidad virtual.”. 2015.

- [74] L. Pérez-Planells, J. Delegido, J. Rivera-Caicedo and J. Verrelst, "Análisis de métodos de validación cruzada para la obtención robusta de parámetros biofísicos", Universitat Politècnica de València, 2016. [Online]. Available: <http://hdl.handle.net/10251/80558>. [Accessed: Jan- 2021].

ANEXOS

ANEXO I: Entrevista

Para lo cual se realizó las siguientes preguntas:

2. ¿Cuál es el objetivo de desarrollar el sistema denominado Ecuciencia?

El desarrollo del sistema Ecuciencia tiene como propósito recolectar información y documentos científicos relacionados con las investigaciones realizada en la Universidad Tecnológica Cotopaxi, y el alcance del proyecto es cubrir todas las universidades del Distrito 3 e incluso todo el país.

Este proyecto se centra en la cienciometría, que es la ciencia a cargo del campo de investigación, cabe mencionar que las métricas que establece son amplias y se actualizarán constantemente. Hoy en día, en este país, la investigación juega un papel fundamental en su desarrollo e innovación.

3. ¿Cómo contribuye la implementación de este sistema a la Universidad Técnica de Cotopaxi?

El sistema ayudará a las autoridades universitarias a tomar decisiones, ya que la información recopilada se utilizará para generar diversas formas de imágenes visuales, por ejemplo, las carreras que están produciendo más material científico o cuales son las líneas y sublíneas más copadas.

4. ¿Cuánto tiempo lleva funcionando el sistema Ecuciencia?

ECUCIENCIA, es parte del proyecto investigativo Red de Estudios Cienciométricos (REDEC), el mismo que comenzó en el año 2018.

5. ¿Cuáles son las funciones actuales del sistema?

Actualmente el sistema tiene una infraestructura totalmente funcional, y permite la gestión de usuarios, investigadores, así como sus aportes científicos, además tienen algunos módulos de análisis de datos apoyados en inteligencia artificial, por ejemplo, los grafos de redes sociales, mapas de clasificación dependiendo de sus campos de estudios, entre otros.

5. ¿Cuál es el uso de la información recopilada?

El departamento de investigación utiliza la información recopilada para analizar los documentos de cada investigador y emitir un certificado, así como ver cuánta producción científica se realiza en cada carrera.

6. ¿Con qué lenguaje de programación fue desarrollado el sistema?

El lenguaje de programación que está utilizando es Python, apoyado en el Framework Django.

7. ¿Cuál es el Gestor de Base de Datos con el que está trabajando el sistema?

Se está trabajando con el Gestor de bases de Datos PostgreSQL, porque es de código abierto, robusto y, lo más importante, tiene una buena conexión con el lenguaje de programación Python. Además, otro de los puntos a considerar para la elección de este gestor es que los estudiantes y profesores que participan en el proyecto tienen experiencia en el uso de PostgreSQL.

8. ¿Cree que el sistema necesita implementar nuevas funciones?

Si requiere de nuevas funcionalidades, porque siempre se está innovando, tratando de buscar nuevos complementos que permitan mejorar el sistema. Además, se requiere un módulo que mejore el proceso de predicción al momento de establecer una línea y sublínea de investigación cuando un nuevo documento científico esté por ingresar.

9. ¿Cuáles son las funcionalidades que requiere el sistema?

Actualmente se ha visto la necesidad de implementar las funcionalidades relacionadas a la clasificación de información, utilizando herramientas de visualización que permitan un mejor análisis de los datos recolectados.

Es necesario utilizar algoritmos de aprendizaje profundo para mejorar el nivel de predicción y clasificación de datos. Existe una herramienta llamada TensorFlow que está escrita para Python y contiene muchos algoritmos de aprendizaje profundo.

10. ¿Qué espera ver en el sistema al implementar nuevas funciones?

Lo que considero importante es que el sistema se encargue de realizar un análisis de texto para obtener una clasificación de líneas y sublíneas de investigación. Cuando un investigador ingrese un nuevo artículo científico, que el sistema sea el encargado de sugerir a qué línea y sublínea pertenece dependiendo del análisis del texto previamente realizado.

11. ¿Cuáles son los beneficios de implementar estas funcionalidades?

Esta funcionalidad permitirá restringir el campo línea y sublínea que antes era de libre elección. Al ser de libre elección, el usuario era propenso a establecer una línea y sublínea totalmente errónea, generando así que el sistema almacene datos basura. Con esta funcionalidad se pretende tener una integridad de datos y por ende información verídica.

ANEXO II: Diagramas de la Etapa de Análisis de Desarrollo de Software.

En este anexo se muestra los resultados de la etapa de Análisis de desarrollo de software.

Diagrama de Casos de Uso

En la figura 42 se puede apreciar el diagrama de Casos de Uso de la propuesta.

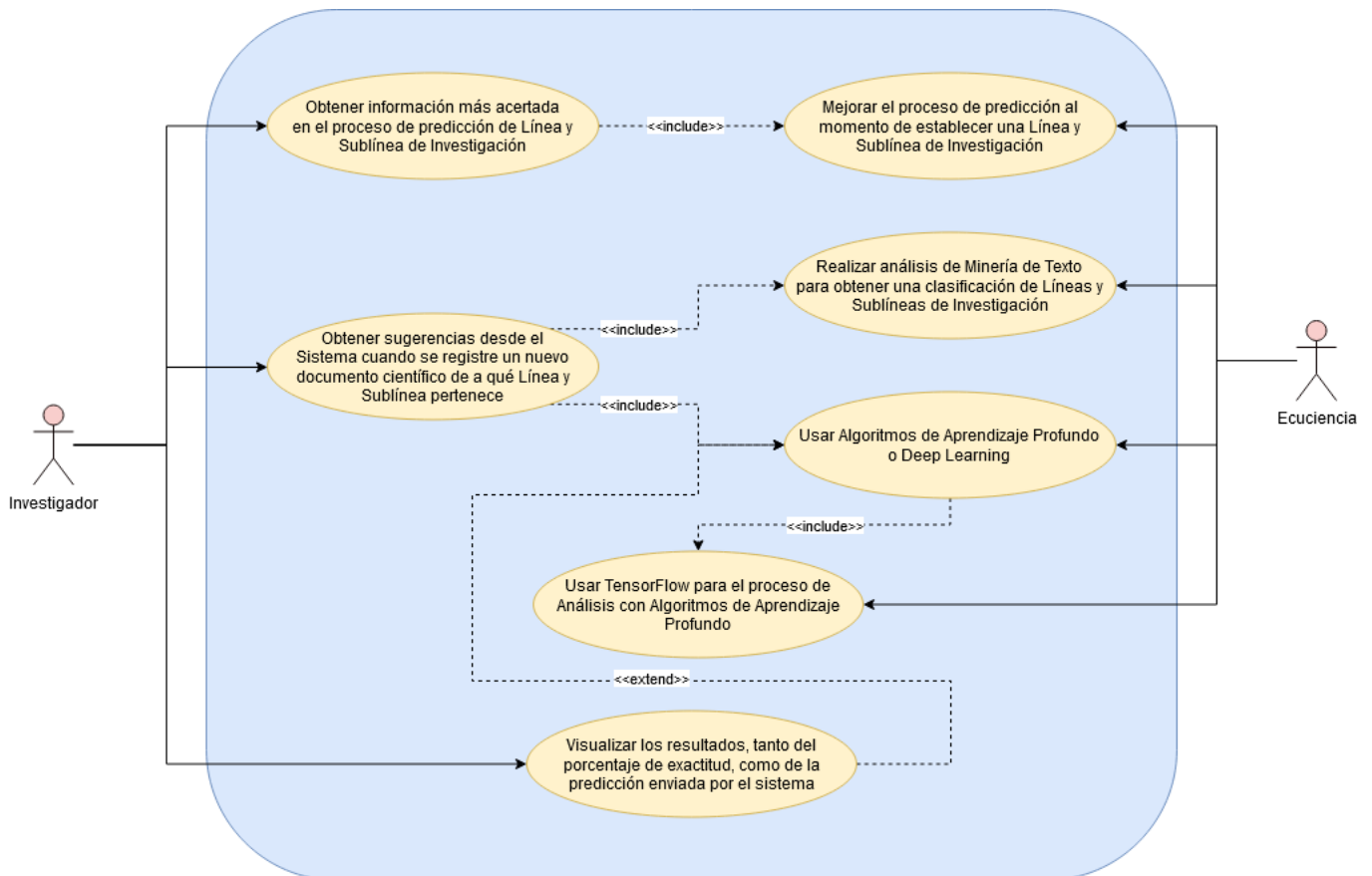


Figura 42: Diagrama de Casos de Uso

Fuente: Investigador

Diagramas de Actividades

En la figura 43 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 01.

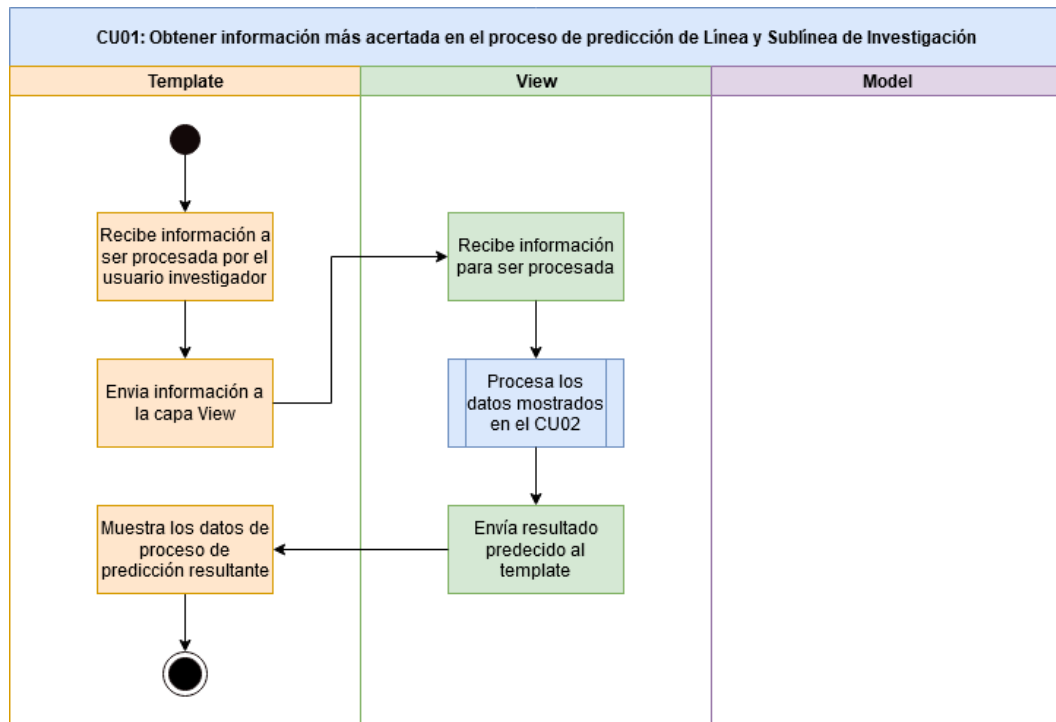


Figura 43: Diagrama de Actividades del Caso de Uso 01

Fuente: Investigador

En la figura 44 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 02.

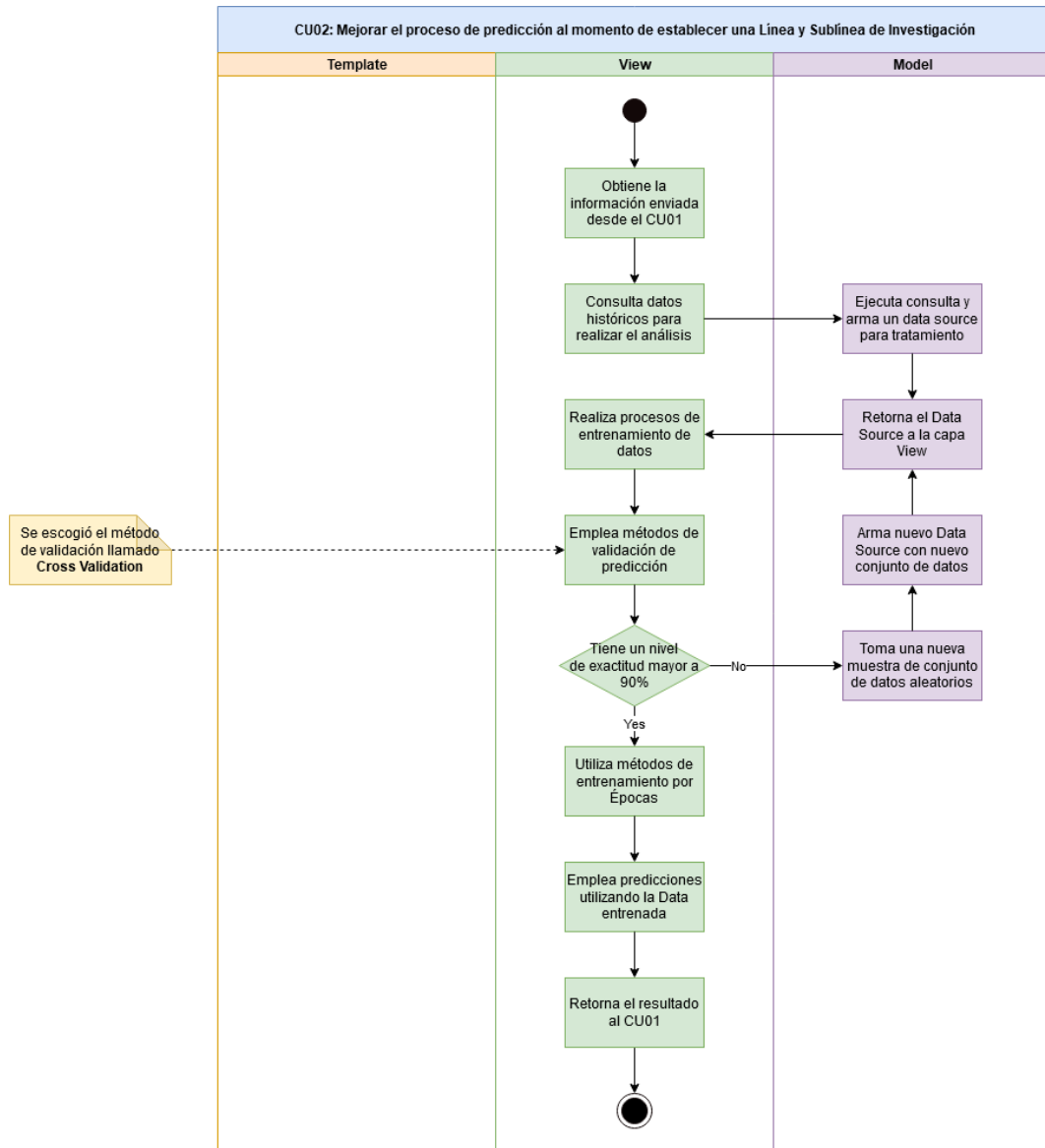


Figura 44: Diagrama de Actividades del Caso de Uso 02

Fuente: Investigador

En la figura 45 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 03.

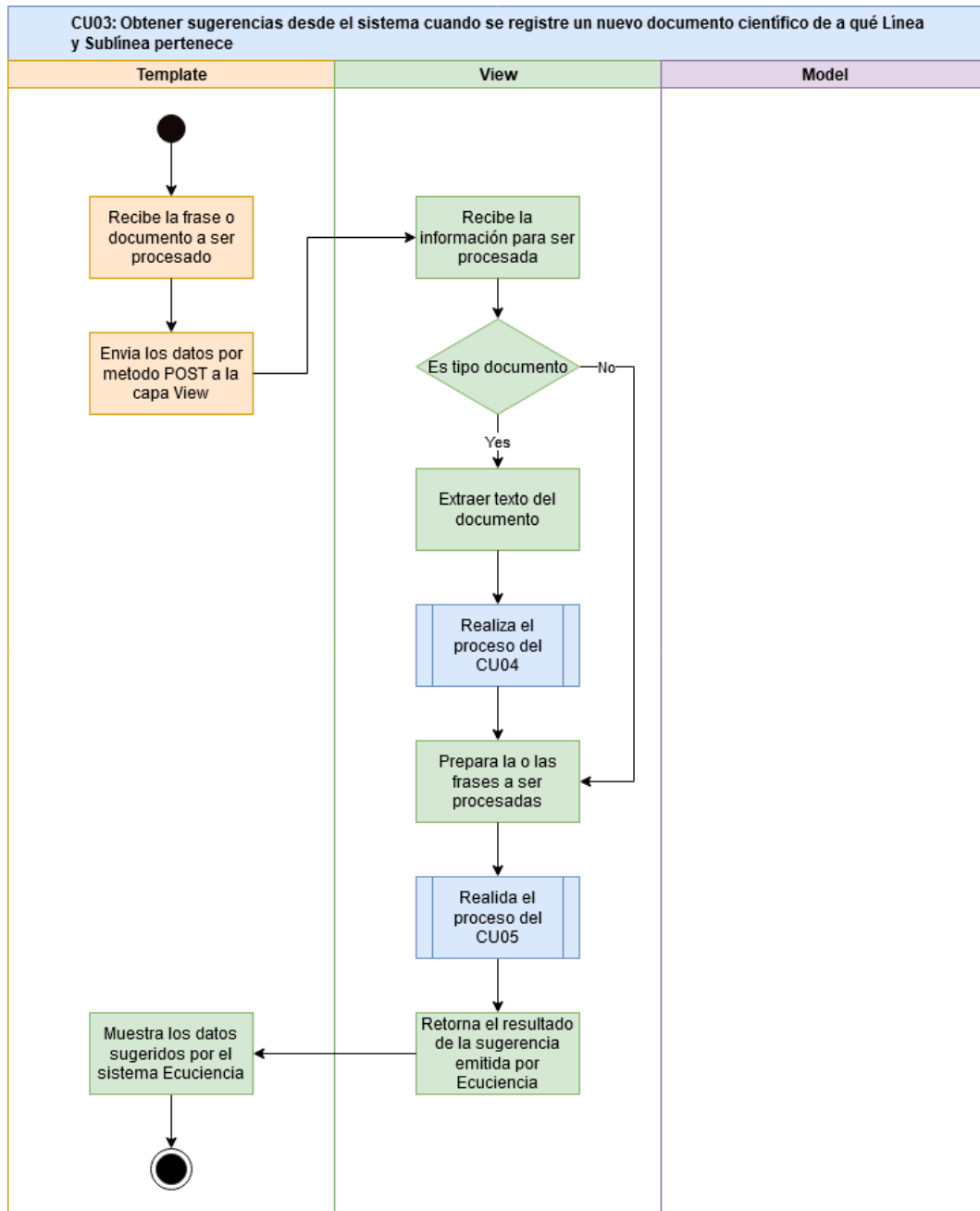


Figura 45: Diagrama de Actividades del Caso de Uso 03

Fuente: Investigador

En la figura 46 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 04.

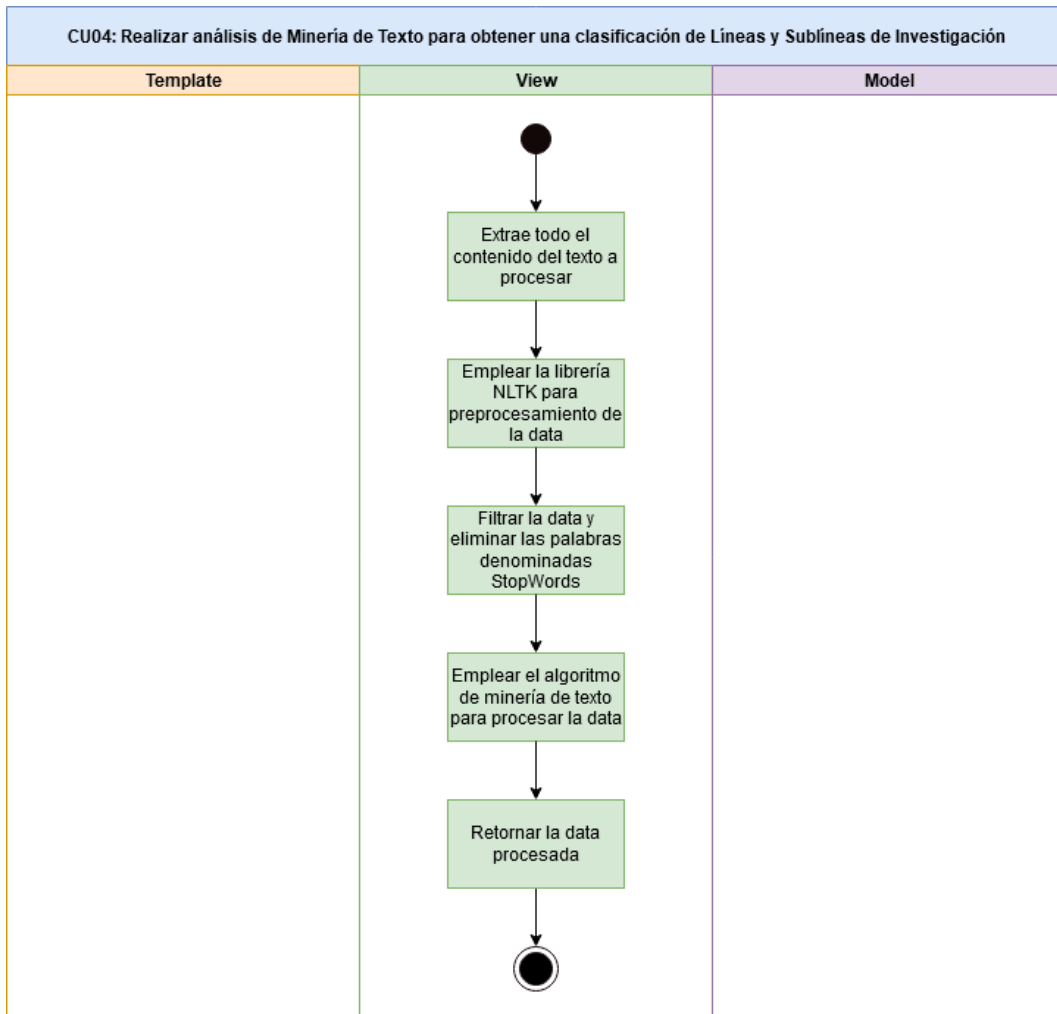


Figura 46: Diagrama de Actividades del Caso de Uso 04

Fuente: Investigador

En la figura 47 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 05.

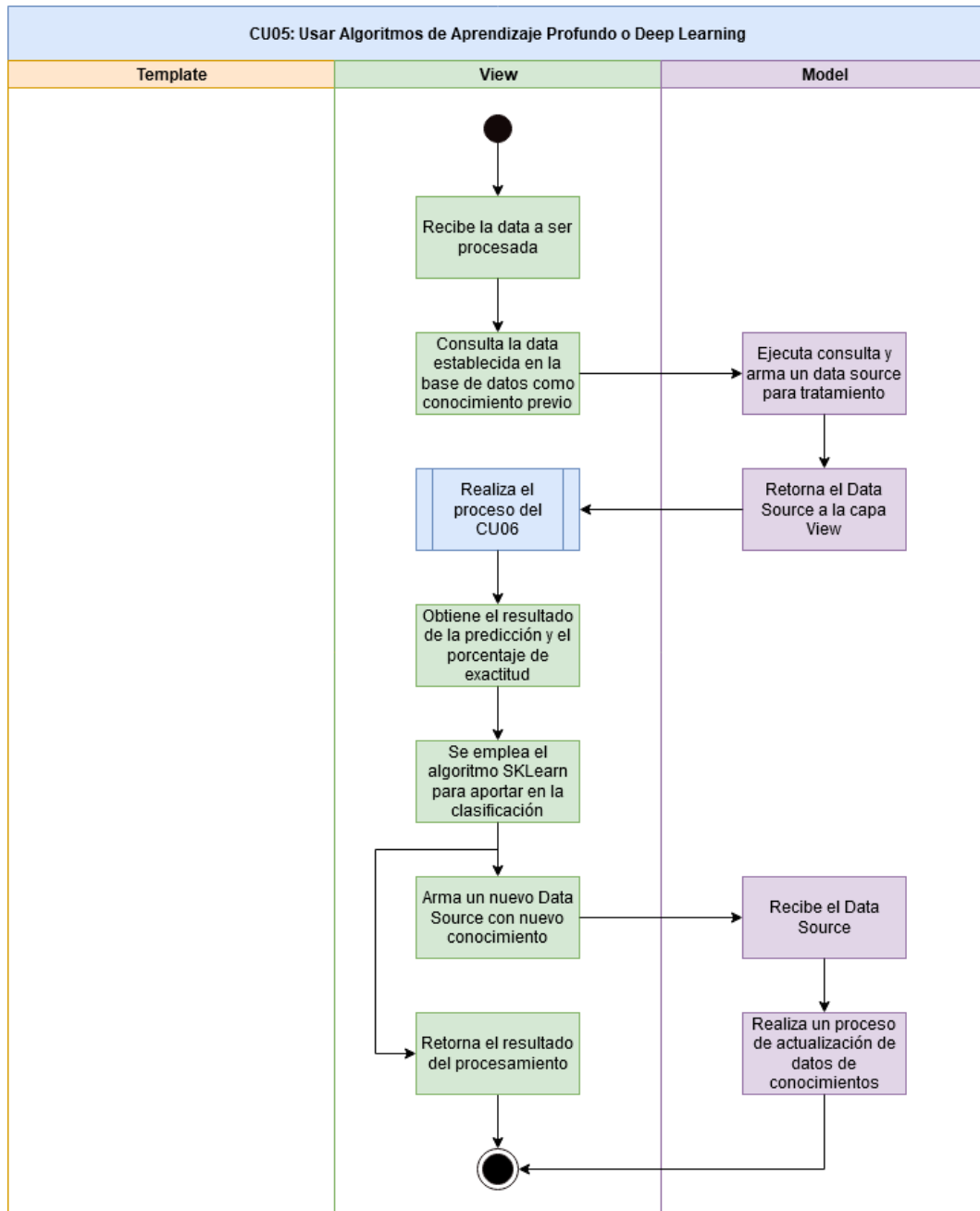


Figura 47: Diagrama de Actividades del Caso de Uso 05

Fuente: Investigador

En la figura 48 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 06.

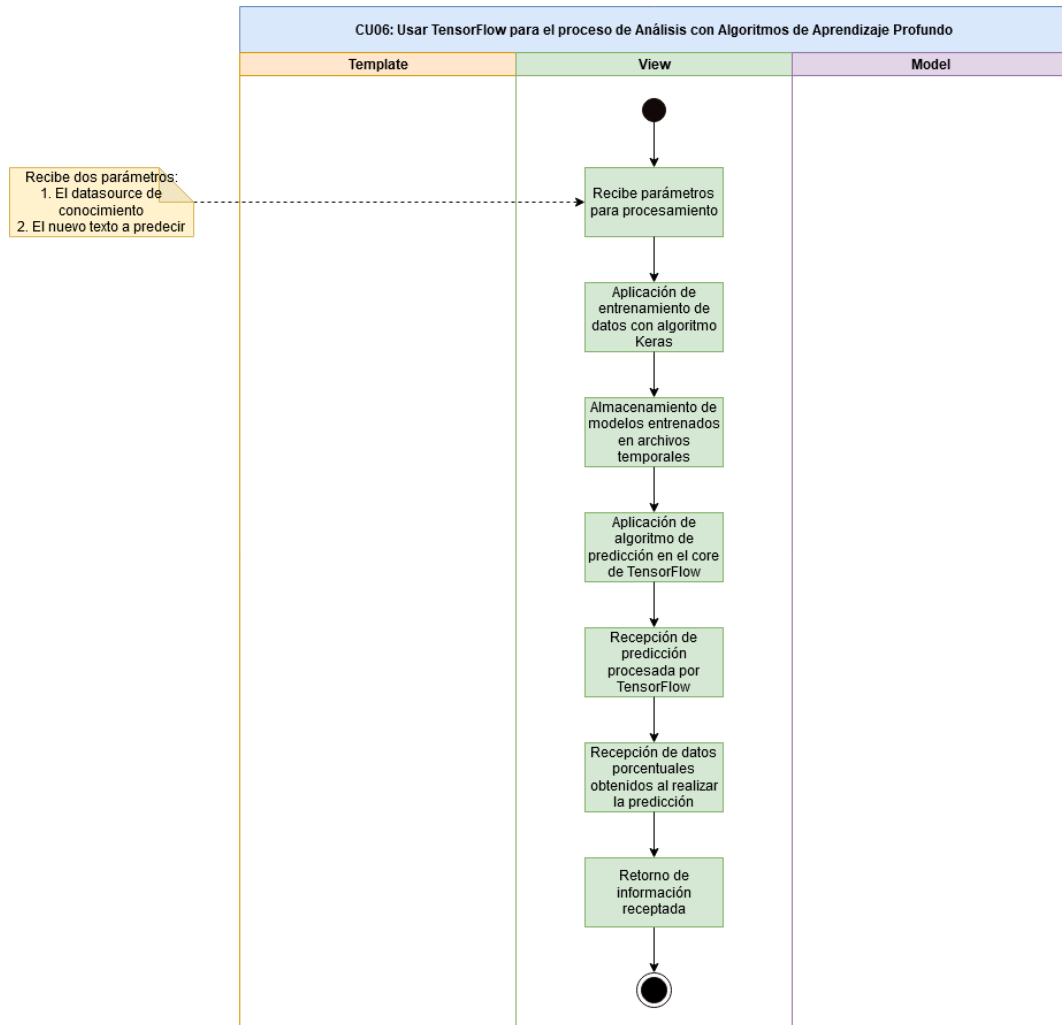


Figura 48: Diagrama de Actividades del Caso de Uso 06

Fuente: Investigador

En la figura 49 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 07.

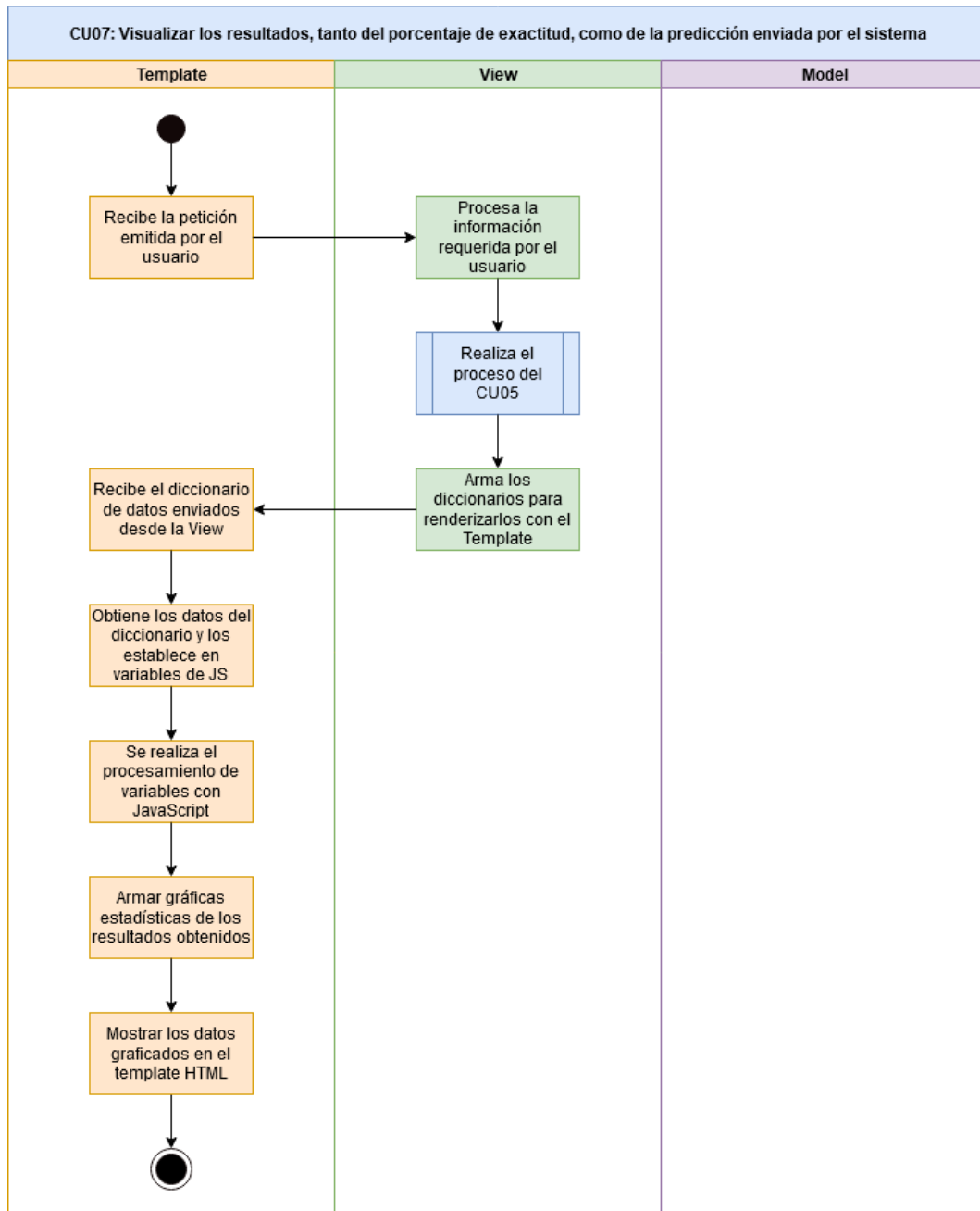


Figura 49: Diagrama de Actividades del Caso de Uso 07

Fuente: Investigador

En la figura 50 se puede apreciar el diagrama de Actividades correspondiente al Caso de Uso 07.

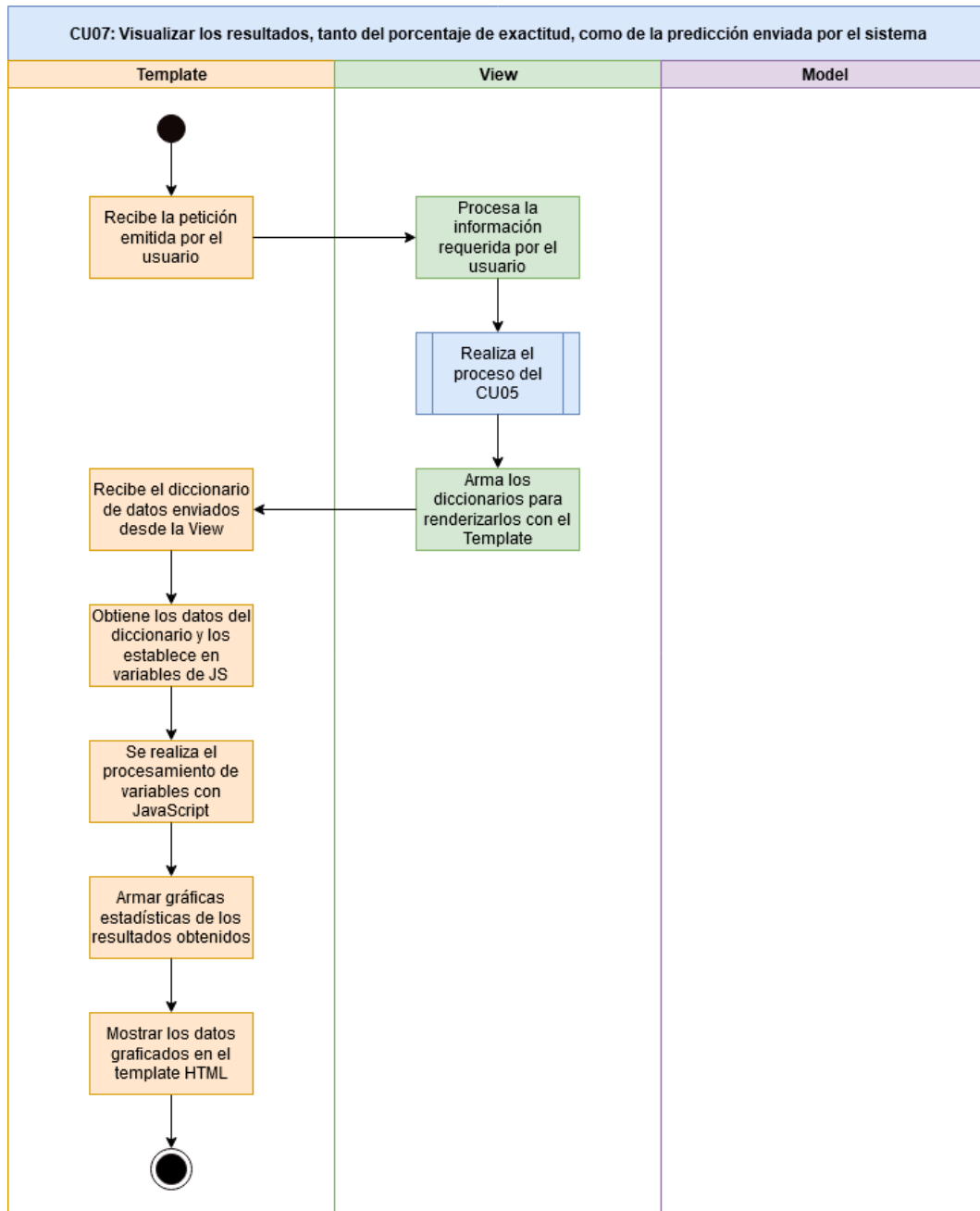


Figura 50: Diagrama de Actividades del Caso de Uso 07

Fuente: Investigador

Diagramas de Secuencia

En la figura 51 se puede apreciar el diagrama de Secuencia correspondiente al Caso de Uso 01 y 02.

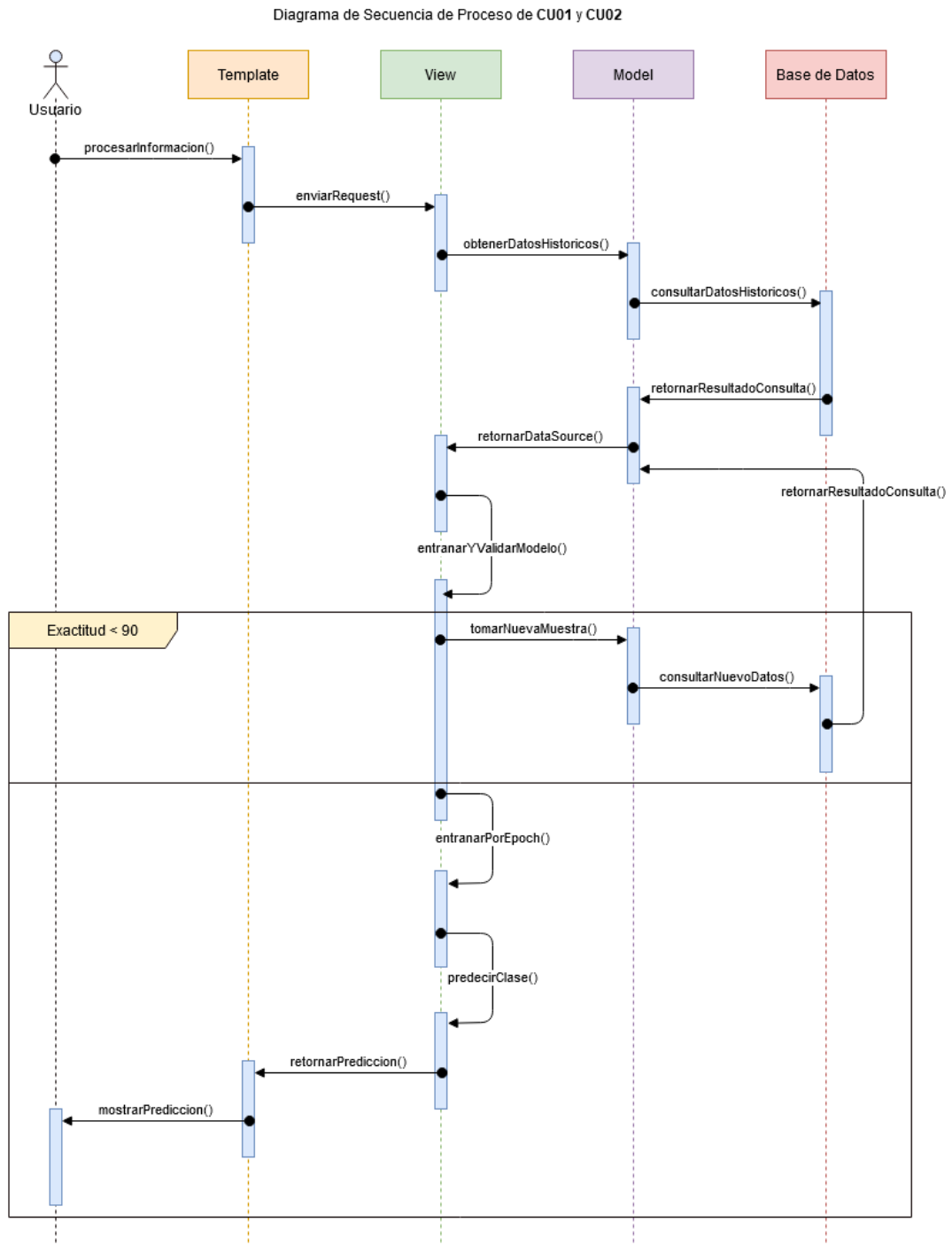


Figura 51: Diagrama de Secuencia del Caso de Uso 01 y 02

Fuente: Investigador

En la figura 52 se puede apreciar el diagrama de Secuencia correspondiente al Caso de Uso 03 y 04.

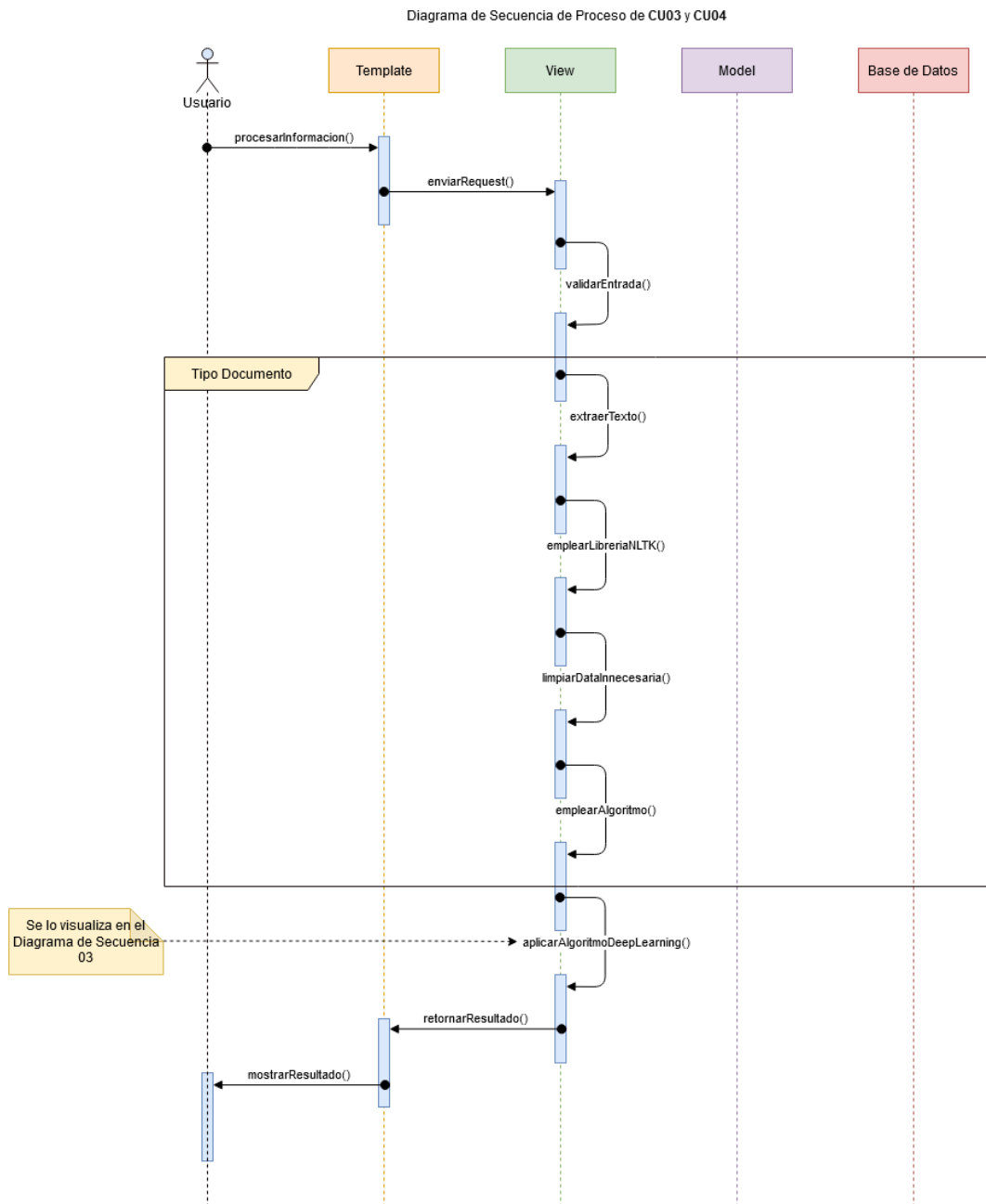


Figura 52: Diagrama de Secuencia del Caso de Uso 03 y 04

Fuente: Investigador

En la figura 53 se puede apreciar el diagrama de Secuencia correspondiente al Caso de Uso 05 y 06.

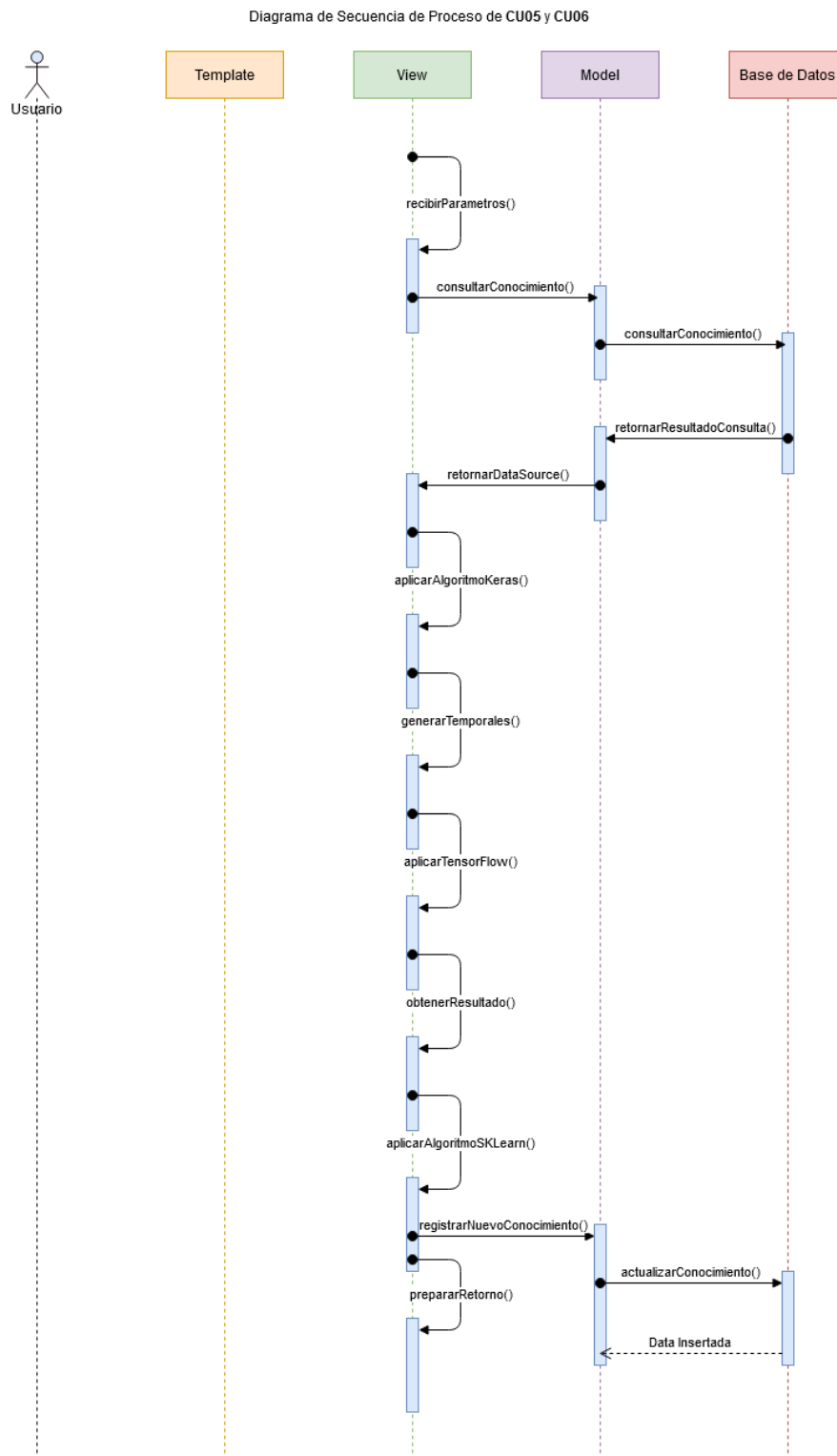


Figura 53: Diagrama de Secuencia del Caso de Uso 05 y 06

Fuente: Investigador

En la figura 54 se puede apreciar el diagrama de Secuencia correspondiente al Caso de Uso 07.

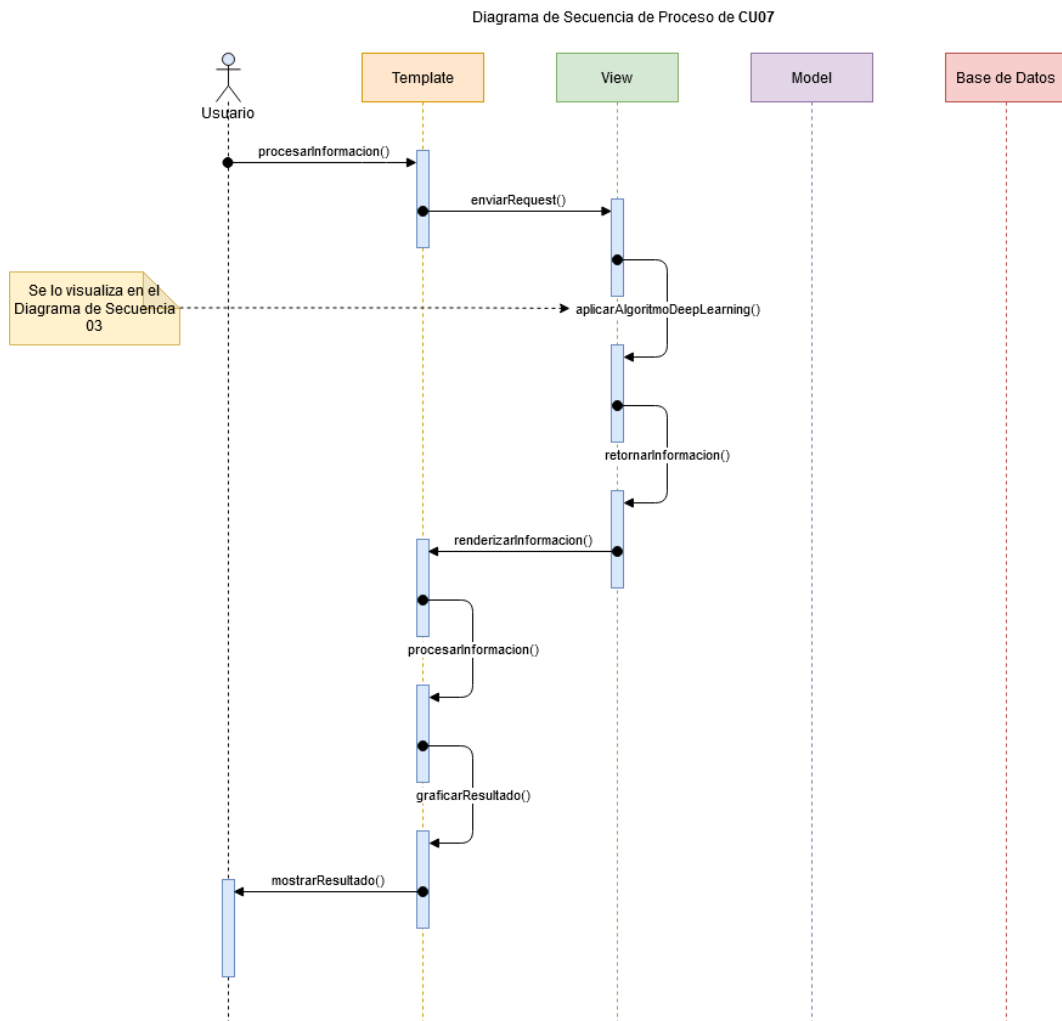


Figura 54: Diagrama de Secuencia del Caso de Uso 07

Fuente: Investigador

Diagramas de Entidad - Relación

En la figura 55 se puede apreciar el diagrama Entidad – Relación de las tablas que serán utilizadas para el desarrollo de la presente propuesta.

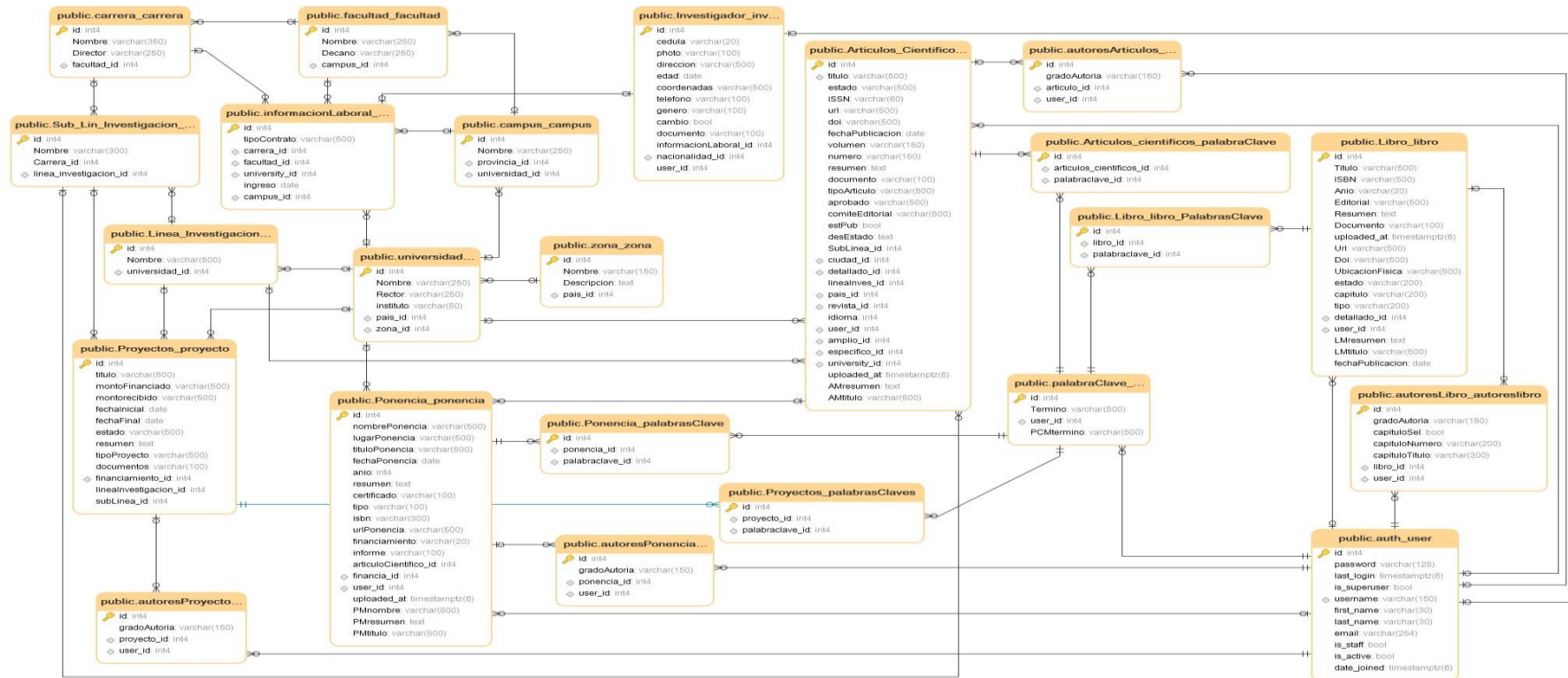


Figura 55: Diagrama Entidad - Relación

Fuente: Investigador

ANEXO III: Diseño de Maquetación de Interfaces.

En este anexo se muestra los resultados de la etapa de Diseño de desarrollo de software a través de la representación de los maquetados de interfaces.

Maquetados de Interfaces de Usuario

En la figura 56 se puede apreciar el maquetado de la interfaz de inicio.

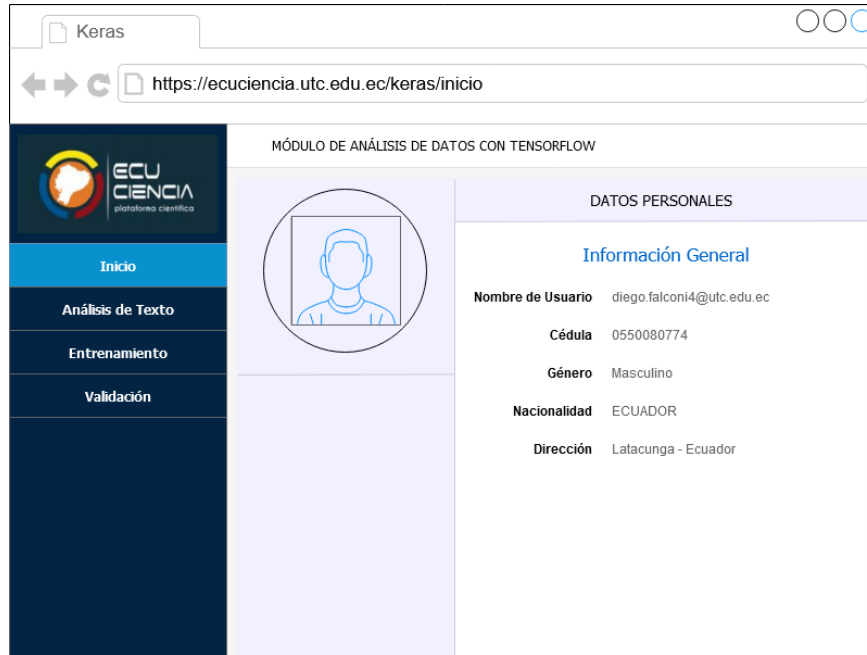


Figura 56: Maquetado de interfaz de inicio

Fuente: Investigador

En la figura 57 se puede apreciar el maquetado de la interfaz de análisis de texto.

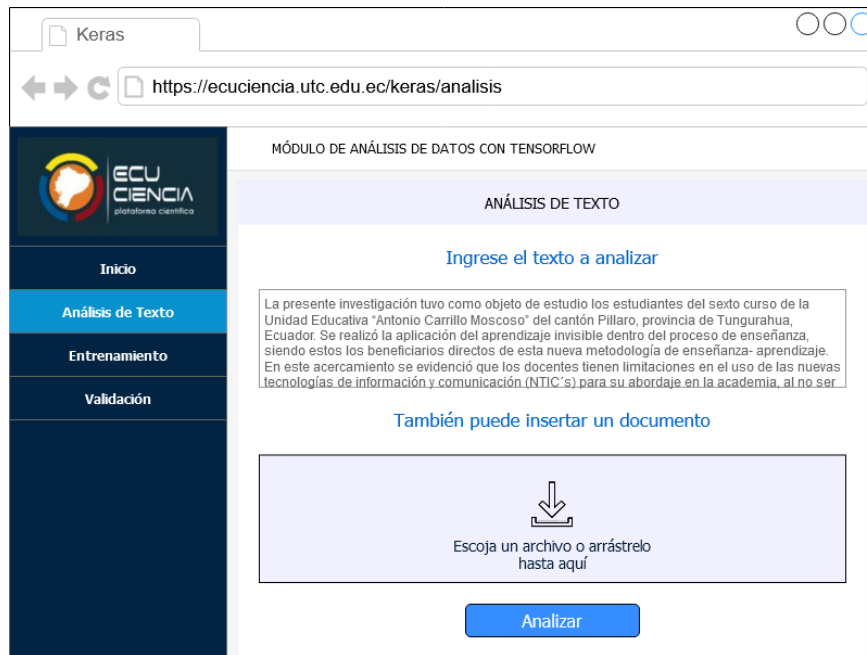


Figura 57: Maquetado de interfaz de análisis de texto

Fuente: Investigador

En la figura 58 se puede apreciar el maquetado de la interfaz de resultado de análisis.

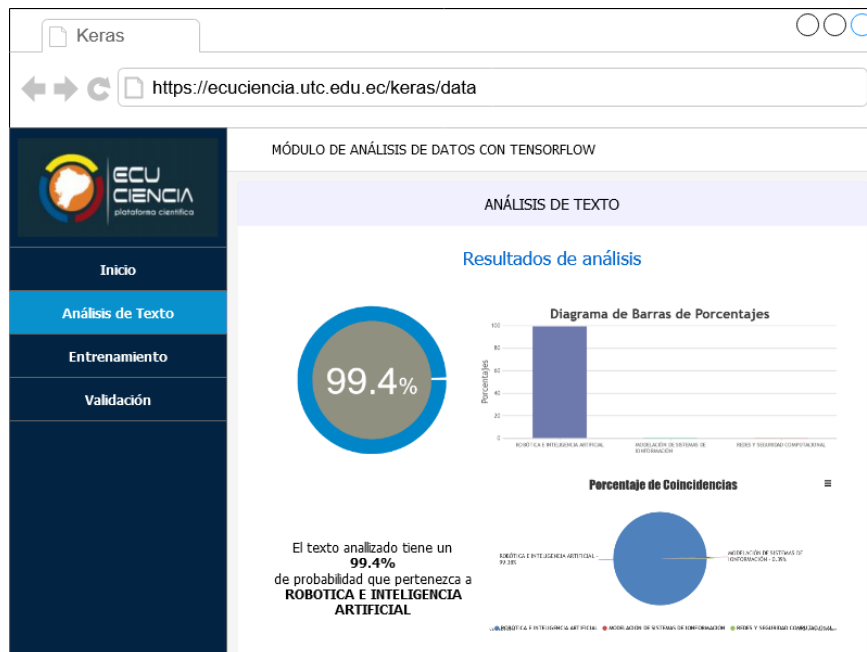


Figura 58: Maquetado de interfaz de resultado de análisis

Fuente: Investigador

En la figura 59 se puede apreciar el maquetado de la interfaz de entrenamiento de datos.

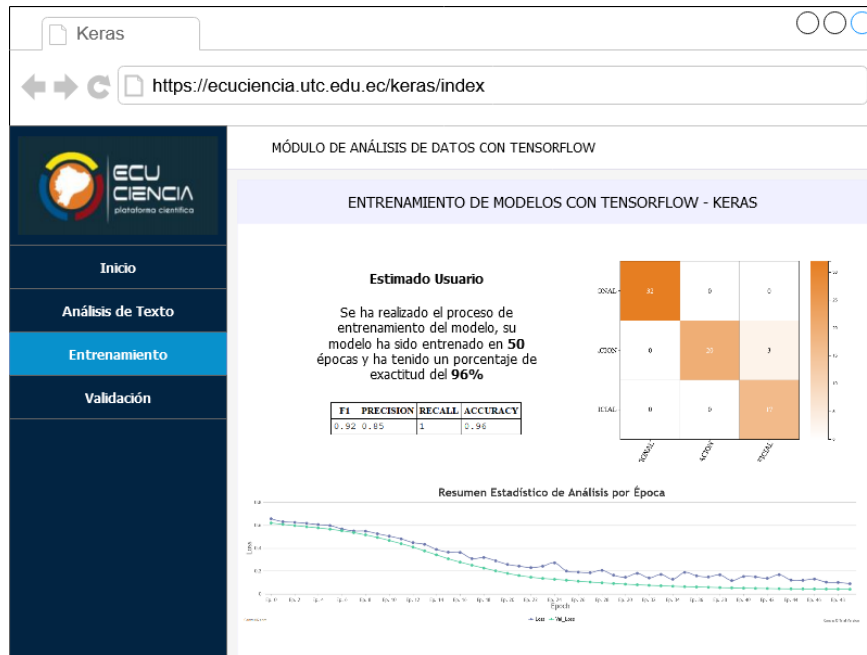


Figura 59: Maquetado de interfaz de entrenamiento de datos

Fuente: Investigador

En la figura 60 se puede apreciar el maquetado de la interfaz de validación.

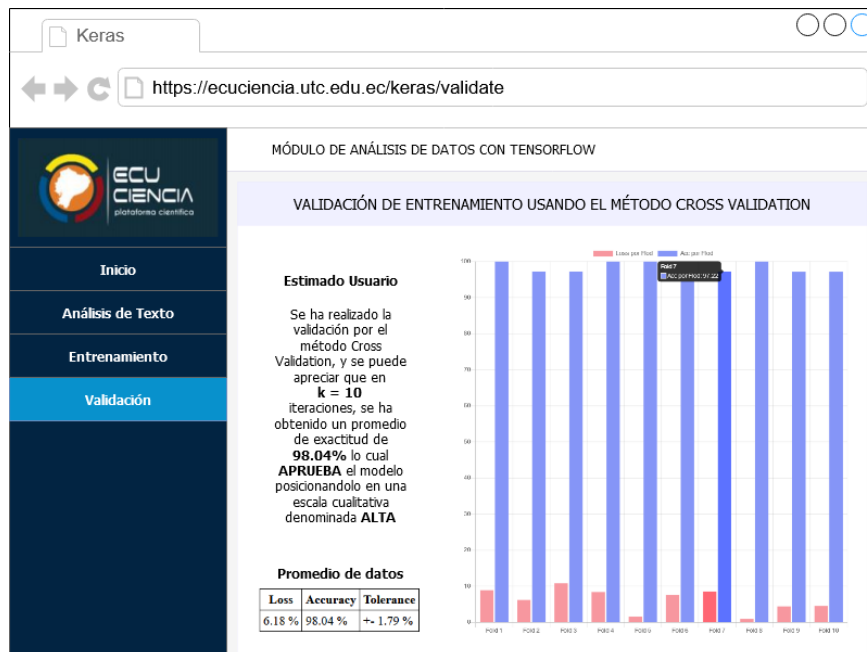


Figura 60: Maquetado de interfaz de validación de entrenamiento

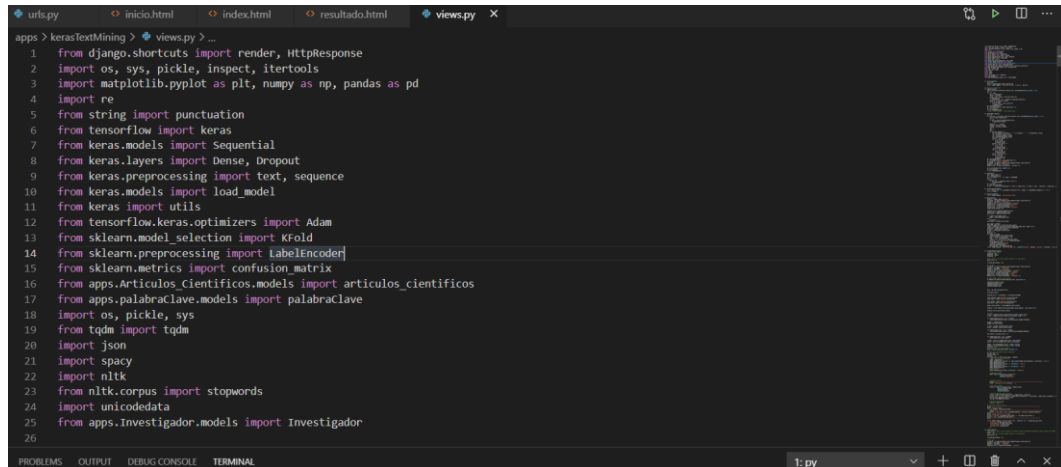
Fuente: Investigador

ANEXO IV: Código de programación.

En este anexo se muestra la parte de la codificación de la propuesta.

Código Python

En la figura 61 se puede apreciar las librerías importadas en la clase View.py del nuevo módulo.

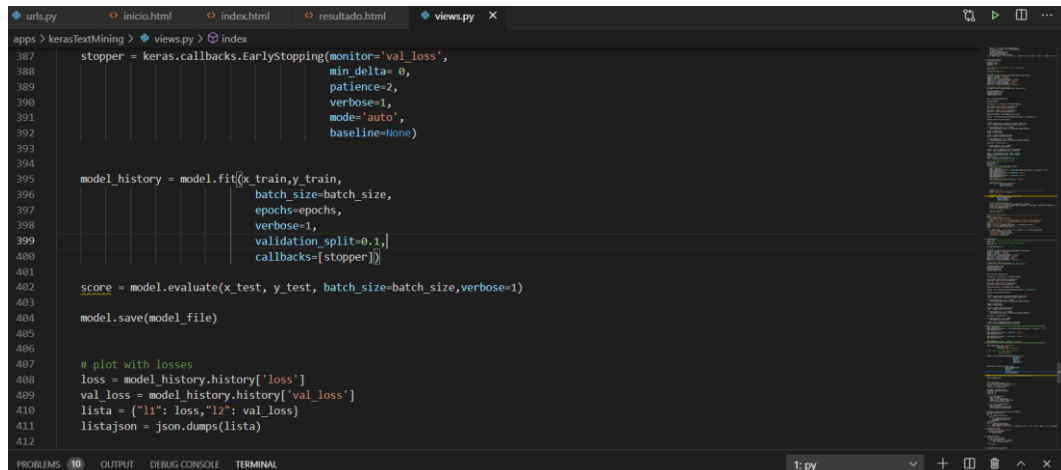


```
1 from django.shortcuts import render, HttpResponseRedirect
2 import os, sys, pickle, inspect, itertools
3 import matplotlib.pyplot as plt, numpy as np, pandas as pd
4 import re
5 from string import punctuation
6 from tensorflow import keras
7 from keras.models import Sequential
8 from keras.layers import Dense, Dropout
9 from keras.preprocessing import text, sequence
10 from keras.models import load_model
11 from keras import utils
12 from tensorflow.keras.optimizers import Adam
13 from sklearn.model_selection import KFold
14 from sklearn.preprocessing import LabelEncoder
15 from sklearn.metrics import confusion_matrix
16 from apps.Articulos Cientificos.models import articulos_cientificos
17 from apps.palabraClave.models import palabraClave
18 import os, pickle, sys
19 from tqdm import tqdm
20 import json
21 import spacy
22 import nltk
23 from nltk.corpus import stopwords
24 import unicodedata
25 from apps.Investigador.models import Investigador
26
```

Figura 61: Librerías usadas en la clase View.py

Fuente: Investigador

En la figura 62 se puede apreciar el llamado y validación a la librería Keras.

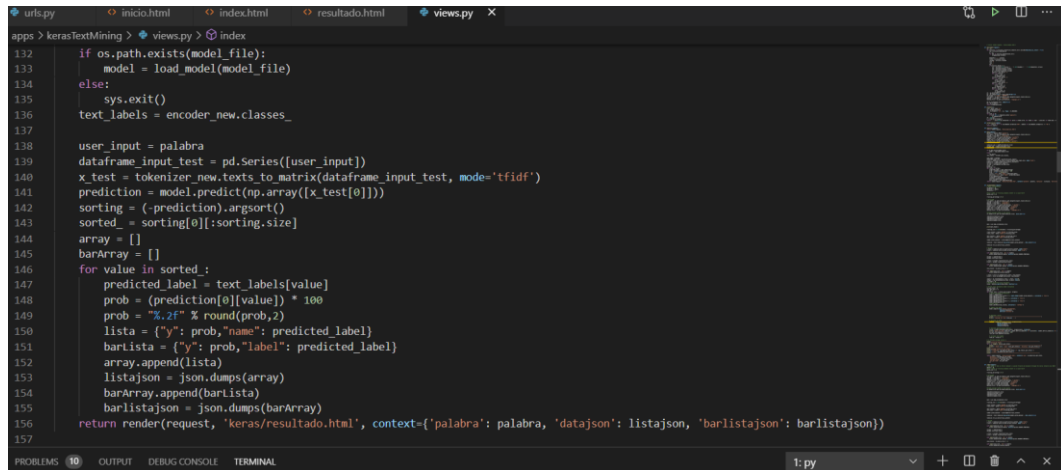


```
387 stopper = keras.callbacks.EarlyStopping(monitor='val_loss',
388                                       min_delta= 0,
389                                       patience=2,
390                                       verbose=1,
391                                       mode='auto',
392                                       baseline=None)
393
394
395 model_history = model.fit(x_train,y_train,
396                          batch_size=batch_size,
397                          epochs=epochs,
398                          verbose=1,
399                          validation_split=0.1,
400                          callbacks=[stopper])
401
402 score = model.evaluate(x_test, y_test, batch_size=batch_size,verbose=1)
403
404 model.save(model_file)
405
406
407 # plot with losses
408 loss = model_history.history['loss']
409 val_loss = model_history.history['val_loss']
410 lista = {"l1": loss,"l2": val_loss}
411 listajson = json.dumps(lista)
412
```

Figura 62: Llamado y validación de librería Keras

Fuente: Investigador

En la figura 63 se puede apreciar el código de predicción de la frase receptada desde el template.



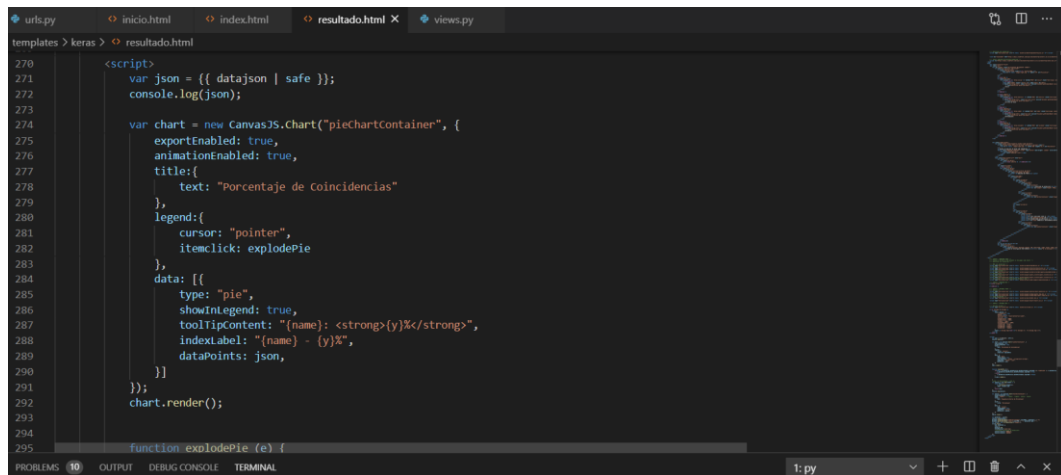
```
132 if os.path.exists(model_file):
133     model = load_model(model_file)
134 else:
135     sys.exit()
136 text_labels = encoder_new.classes_
137
138 user_input = palabra
139 dataframe_input_test = pd.Series([user_input])
140 x_test = tokenizer_new.texts_to_matrix(dataframe_input_test, mode='tfidf')
141 prediction = model.predict(np.array(x_test[0]))
142 sorting = (-prediction).argsort()
143 sorted_ = sorting[0]:sorting.size]
144 array = []
145 barArray = []
146 for value in sorted_:
147     predicted_label = text_labels[value]
148     prob = (prediction[0][value]) * 100
149     prob = "%.2f" % round(prob,2)
150     lista = {'y': prob, 'name': predicted_label}
151     barlista = {'y': prob, 'label': predicted_label}
152     array.append(lista)
153     listajson = json.dumps(array)
154     barArray.append(barlista)
155     barlistajson = json.dumps(barArray)
156 return render(request, 'keras/resultado.html', context={'palabra': palabra, 'datajson': listajson, 'barlistajson': barlistajson})
157
```

Figura 63: Código de predicción de datos

Fuente: Investigador

Código JavaScript

En la figura 64 se puede apreciar el código que captura los datos del diccionario renderizado desde la View.py al template.

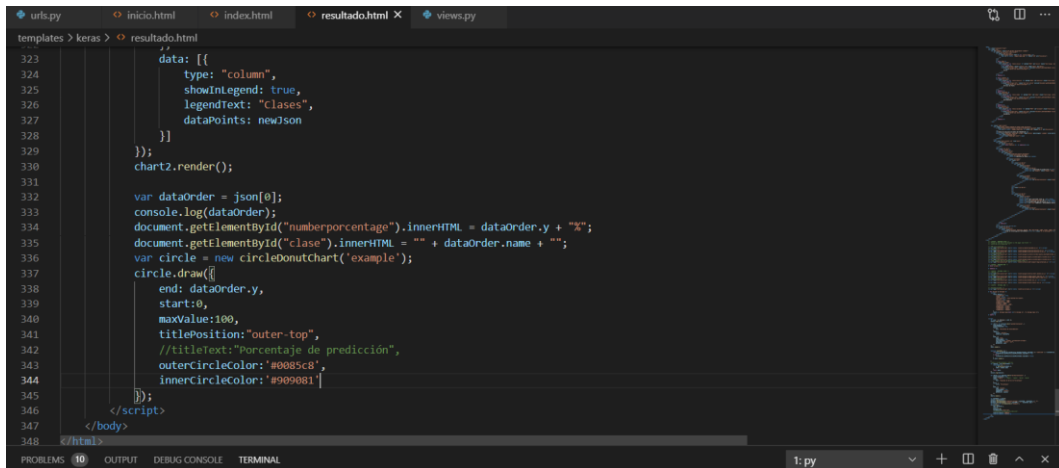


```
270 <script>
271     var json = {{ datajson | safe }};
272     console.log(json);
273
274     var chart = new CanvasJS.Chart("piechartcontainer", {
275         exportEnabled: true,
276         animationEnabled: true,
277         title: {
278             text: "Porcentaje de coincidencias"
279         },
280         legends: {
281             cursor: "pointer",
282             itemclick: explodePie
283         },
284         data: [{
285             type: "pie",
286             showLegend: true,
287             tooltipContent: "{name}: <strong>{y}</strong>",
288             indexLabel: "{name} - {y}%",
289             dataPoints: json,
290         }]
291     });
292     chart.render();
293
294
295     function explodePie (e) {
```

Figura 64: Captura de datos renderizados desde la View.py

Fuente: Investigador

En la figura 65 se puede apreciar el renderizado de las gráficas y las validaciones de los textos para el HTML.



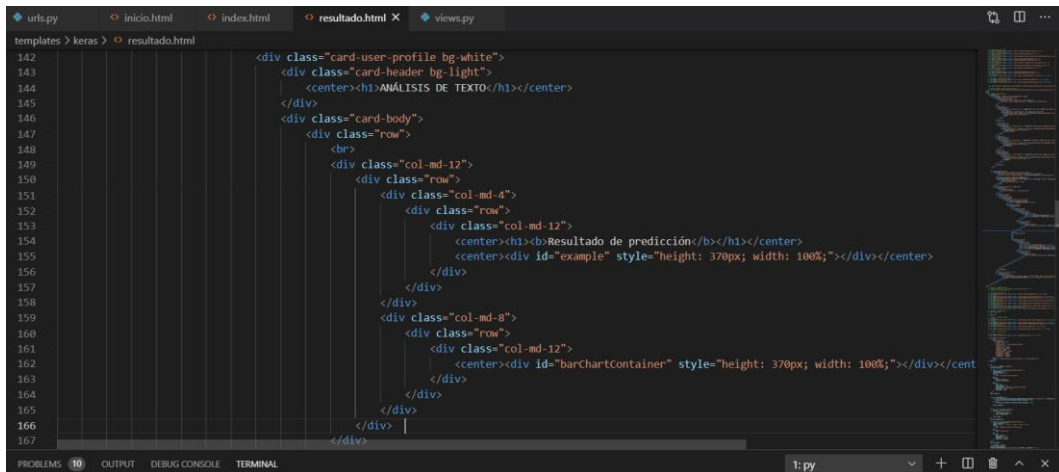
```
323     data: [{
324       type: "column",
325       showInLegend: true,
326       legendText: "Clases",
327       dataPoints: newJson
328     }]
329   });
330   chart2.render();
331
332   var dataOrder = json[0];
333   console.log(dataOrder);
334   document.getElementById("numberporcentaje").innerHTML = dataOrder.y + "%";
335   document.getElementById("clase").innerHTML = "" + dataOrder.name + "";
336   var circle = new circleDonutChart('example');
337   circle.draw({
338     end: dataOrder.y,
339     start: 0,
340     maxValue: 100,
341     titlePosition: "outer-top",
342     //titleText: "Porcentaje de predicción",
343     outerCircleColor: "#0085c8",
344     innerCircleColor: "#909081"
345   });
346 </script>
347 </body>
348 </html>
```

Figura 65: Renderizado de resultados desde JS a HTML

Fuente: Investigador

Código HTML

En la figura 66 se puede apreciar el código HTML reservado para los gráficos estadísticos renderizados por JS.

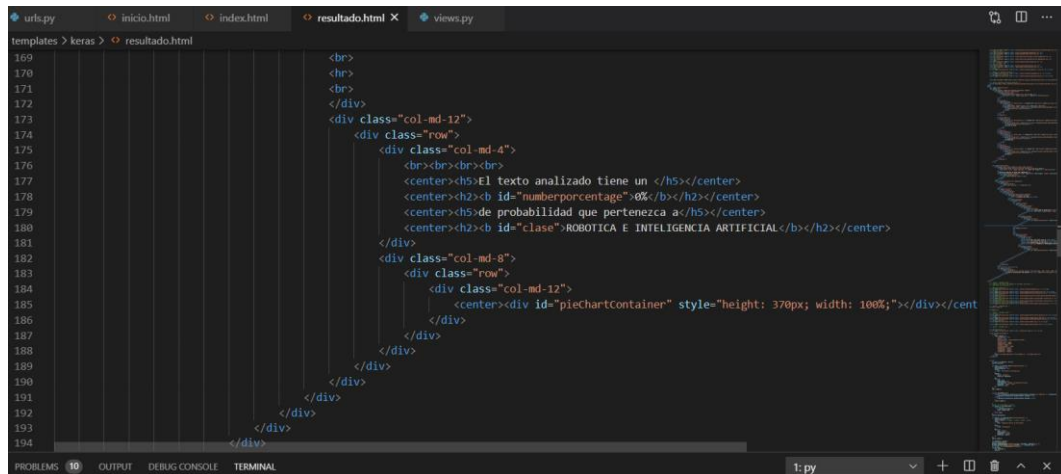


```
142 <div class="card-user-profile bg-white">
143 <div class="card-header bg-light">
144 <center><h1>ANÁLISIS DE TEXTO</h1></center>
145 </div>
146 <div class="card-body">
147 <div class="row">
148 <br>
149 <div class="col-md-12">
150 <div class="row">
151 <div class="col-md-4">
152 <div class="row">
153 <div class="col-md-12">
154 <center><h1>Resultado de predicción</h1></center>
155 <center><div id="example" style="height: 370px; width: 100%;"></div></center>
156 </div>
157 </div>
158 </div>
159 <div class="col-md-8">
160 <div class="row">
161 <div class="col-md-12">
162 | <center><div id="barChartContainer" style="height: 370px; width: 100%;"></div></cent
163 </div>
164 </div>
165 </div>
166 </div> |
167 </div>
```

Figura 66: Código HTML reservado para gráficos estadísticas

Fuente: Investigador

En la figura 67 se puede apreciar el código HTML reservado para la representación de los textos.

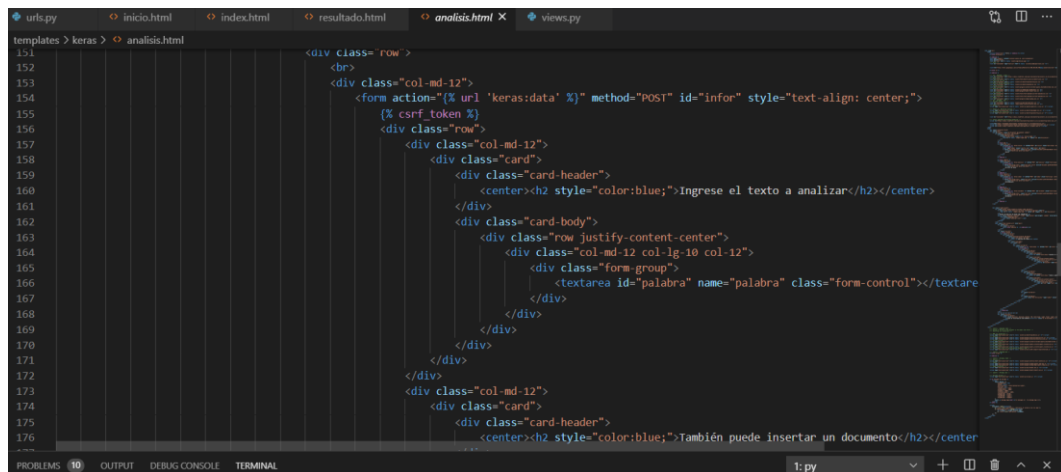


```
169 <br>
170 <hr>
171 <br>
172 </div>
173 <div class="col-md-12">
174 <div class="row">
175 <div class="col-md-4">
176 <br><br><br><br>
177 <center><h5>El texto analizado tiene un </h5></center>
178 <center><h2><b id="numberpercentage">0%</b></h2></center>
179 <center><h5>de probabilidad que pertenezca a</h5></center>
180 <center><h2><b id="clase">ROBOTICA E INTELIGENCIA ARTIFICIAL</b></h2></center>
181 </div>
182 <div class="col-md-8">
183 <div class="row">
184 <div class="col-md-12">
185 <center><div id="pieChartContainer" style="height: 370px; width: 100%;"></div></center>
186 </div>
187 </div>
188 </div>
189 </div>
190 </div>
191 </div>
192 </div>
193 </div>
194 </div>
```

Figura 67: Código HTML reservado para texto dinámico

Fuente: Investigador

En la figura 68 se puede apreciar el código HTML del Form de Análisis de texto.



```
151 <div class="row">
152 <br>
153 <div class="col-md-12">
154 <form action="{% url 'keras:data' %}" method="POST" id="infor" style="text-align: center;">
155 <div class="row">
156 <div class="col-md-12">
157 <div class="card">
158 <div class="card-header">
159 <center><h2 style="color:blue;">Ingrese el texto a analizar</h2></center>
160 </div>
161 <div class="card-body">
162 <div class="row justify-content-center">
163 <div class="col-md-12 col-lg-10 col-12">
164 <div class="form-group">
165 <input type="text" id="palabra" name="palabra" class="form-control"></div>
166 </div>
167 </div>
168 </div>
169 </div>
170 </div>
171 </div>
172 </div>
173 <div class="col-md-12">
174 <div class="card">
175 <div class="card-header">
176 <center><h2 style="color:blue;">También puede insertar un documento</h2></center>
```

Figura 68: Código HTML de tag Form de Análisis de Texto

Fuente: Investigador

ANEXO V: Captura de Interfaces Gráficas.

En este anexo se muestra los resultados de la etapa de Implementación de desarrollo de software a través de la representación de los interfaces graficas de usuario.

Interfaces de Usuario

En la figura 69 se puede apreciar la interfaz de inicio.

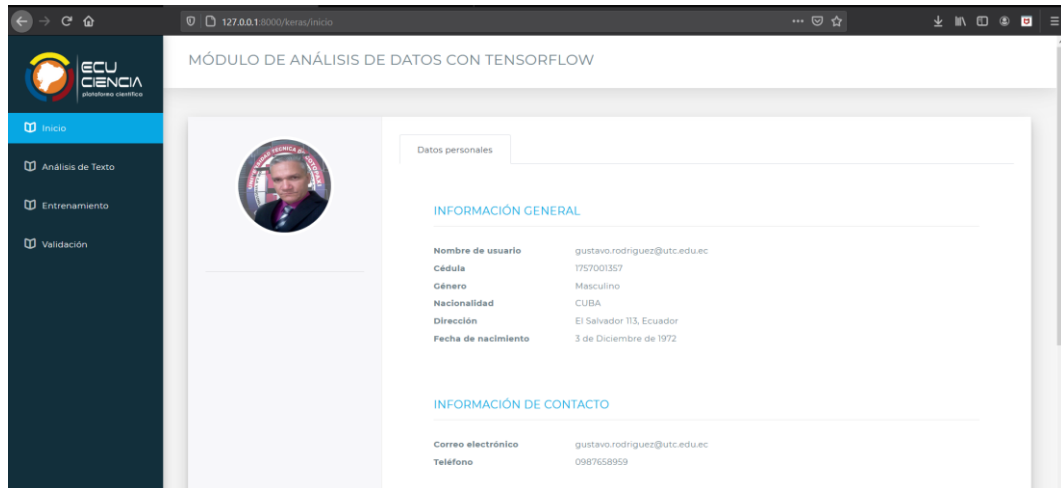


Figura 69: Interfaz de inicio

Fuente: Investigador

En la figura 70 se puede apreciar la interfaz de análisis de texto.



Figura 70: Interfaz de análisis de texto

Fuente: Investigador

En la figura 71 se puede apreciar la interfaz de resultado de análisis.

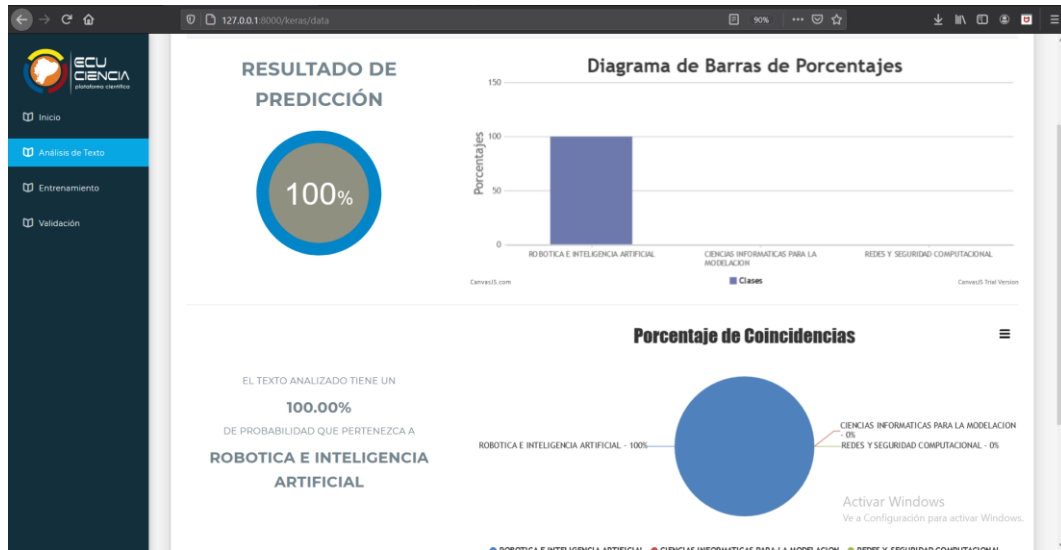


Figura 71: Interfaz de resultado de análisis

Fuente: Investigador

En la figura 72 se puede apreciar la interfaz de entrenamiento de datos.

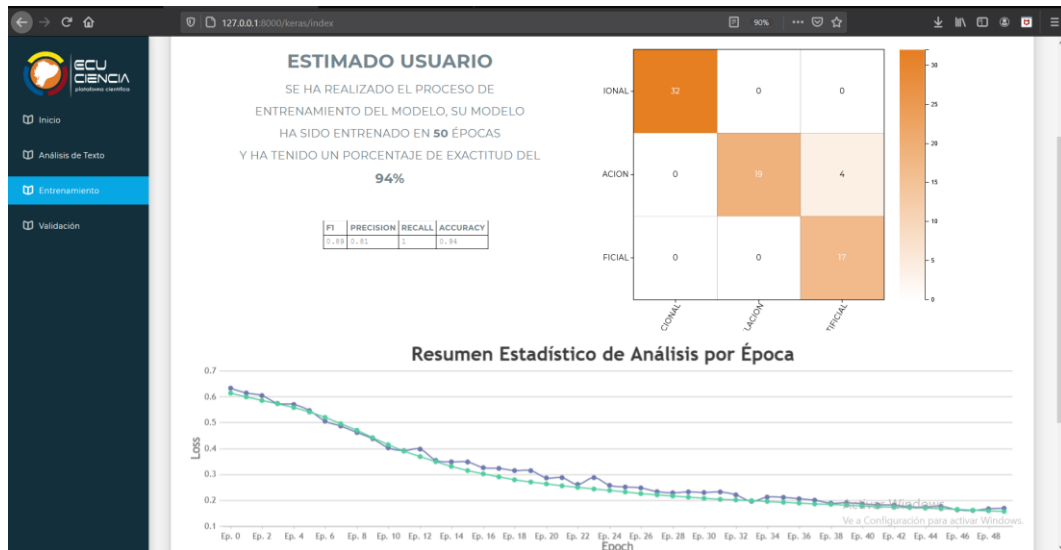


Figura 72: Interfaz de entrenamiento de datos

Fuente: Investigador

En la figura 73 se puede apreciar la interfaz de validación.

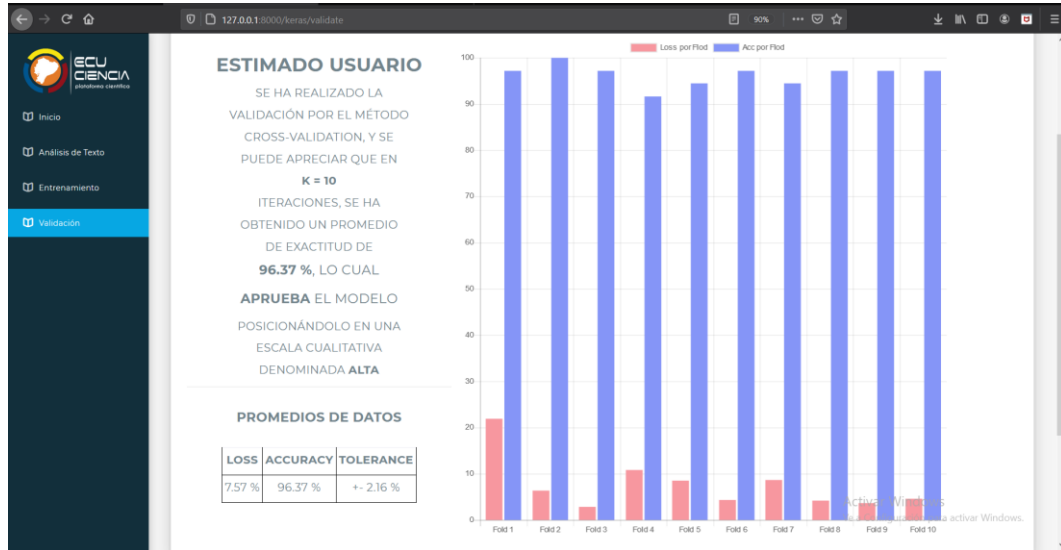


Figura 73: Interfaz de validación de entrenamiento

Fuente: Investigador

ANEXO VI: Matrices de Casos de Pruebas.

En este anexo se muestra los resultados de la etapa de Pruebas del desarrollo de software a través de la representación de los Casos de prueba y resultados validados.

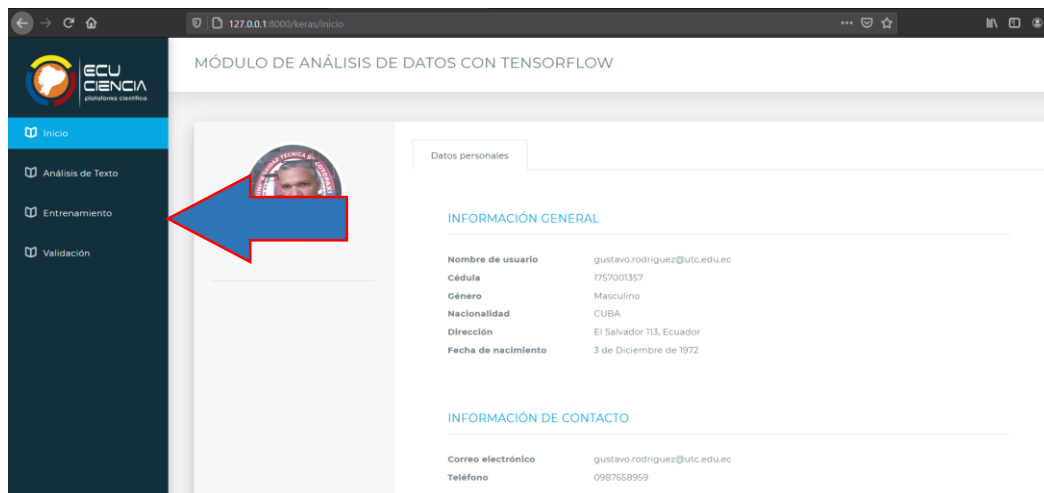
Casos de Pruebas

Tabla 25: Caso de Prueba 01.

CASO DE PRUEBA 01	
Identificador/es de Caso de Uso	CU01 y CU02
Nombre/s de Caso de Uso	<ul style="list-style-type: none">• Obtener información más acertada en el proceso de predicción de Línea y Sublínea de Investigación.• Mejorar el proceso de predicción al momento de establecer una Línea y Sublínea de Investigación.
Descripción de Prueba	Verificar si el valor de exactitud del entrenamiento del modelo, supera el 80% y se lo cataloga dentro de un nivel de aceptación alto.
Responsable	Ing. Diego Falconí.
Prerrequisito	
Ingresar al módulo de Análisis de Datos con TensorFlow.	
Descripción de Caso de Prueba	
Al ingresar en la opción “Entrenamiento” se deberá visualizar que el porcentaje de exactitud supere el 80% de predicción.	
Instrucciones de prueba	

CASO DE PRUEBA 01

1. Ingresar a la opción “Entrenamiento” y esperar a que el Algoritmo realice el proceso en Backend.



MÓDULO DE ANÁLISIS DE DATOS CON TENSORFLOW

Datos personales

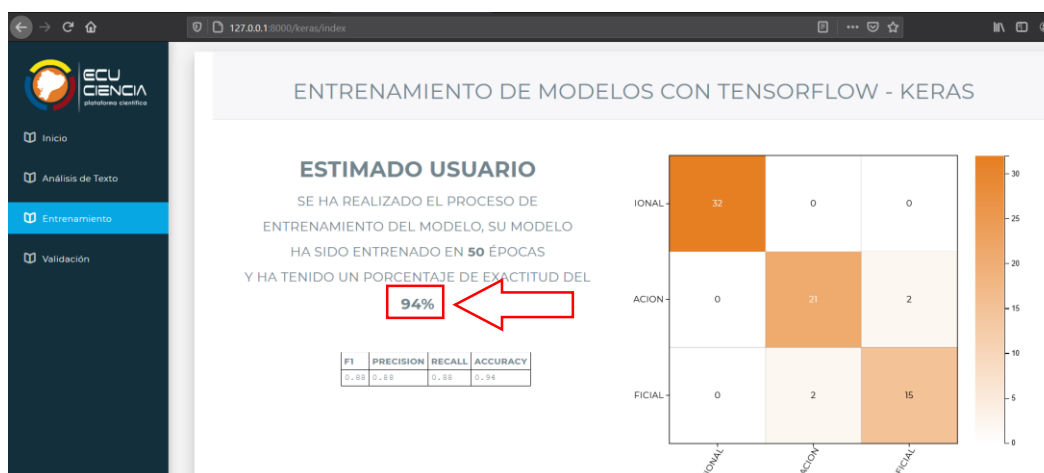
INFORMACIÓN GENERAL

Nombre de usuario: gustavo.rodriguez@utc.edu.ec
Cédula: 1757001357
Género: Masculino
Nacionalidad: CUBA
Dirección: El Salvador 113, Ecuador
Fecha de nacimiento: 3 de Diciembre de 1972

INFORMACIÓN DE CONTACTO

Correo electrónico: gustavo.rodriguez@utc.edu.ec
Teléfono: 0987658959

2. Verificar que el porcentaje de exactitud supere el 80%.



ENTRENAMIENTO DE MODELOS CON TENSORFLOW - KERAS

ESTIMADO USUARIO

SE HA REALIZADO EL PROCESO DE ENTRENAMIENTO DEL MODELO, SU MODELO HA SIDO ENTRENADO EN 50 ÉPOCAS Y HA TENIDO UN PORCENTAJE DE EXACTITUD DEL **94%**

F1	PRECISION	RECALL	ACCURACY
0.88	0.88	0.88	0.94

CONFUSION MATRIX:

	CONFUSION	LACIÓNY	TIPICAMENTE
CONFUSION	32	0	0
LACIÓNY	0	21	2
TIPICAMENTE	0	2	15

CASO DE PRUEBA 01

3. Validar el entrenamiento del modelo en la opción “Validación”.

MÓDULO DE ANÁLISIS DE DATOS CON TENSORFLOW

Datos personales

INFORMACIÓN GENERAL

Nombre de usuario: gustavo.rodriguez@utec.edu.ec
 Cédula: 1757001357
 Género: Masculino
 Nacionalidad: CUBA
 Dirección: El Salvador 113, Ecuador
 Fecha de nacimiento: 3 de Diciembre de 1972

INFORMACIÓN DE CONTACTO

Correo electrónico: gustavo.rodriguez@utec.edu.ec
 Teléfono: 0987658959

4. Verificar que el promedio del porcentaje de exactitud obtenida por el Método de validación “Cross-Validation”, supere el 80%.

ESTIMADO USUARIO

SE HA REALIZADO LA VALIDACIÓN POR EL MÉTODO CROSS-VALIDATION, Y SE PUEDE APRECIAR QUE EN **K = 10** ITERACIONES, SE HA OBTENIDO UN PROMEDIO DE EXACTITUD DE **96.92 %**, LO CUAL **APRUEBA** EL MODELO POSICIONÁNDOLO EN UNA ESCALA CUALITATIVA DENOMINADA **ALTA**.

PROMEDIOS DE DATOS

LOSS	ACCURACY	TOLERANCE
7.00 %	96.92 %	+/- 2.91 %

Bar chart showing Accuracy (Ac. por Fold) and Loss (Loss por Fold) for 10 folds. Accuracy values are consistently high, around 90-100%, while loss values are low, around 0-10%.

Respuesta esperada de la aplicación

Aprueba

El módulo debe presentar tanto en la interfaz de Entrenamiento como en la interfaz de Validación un valor porcentual superior a 80%.

Si

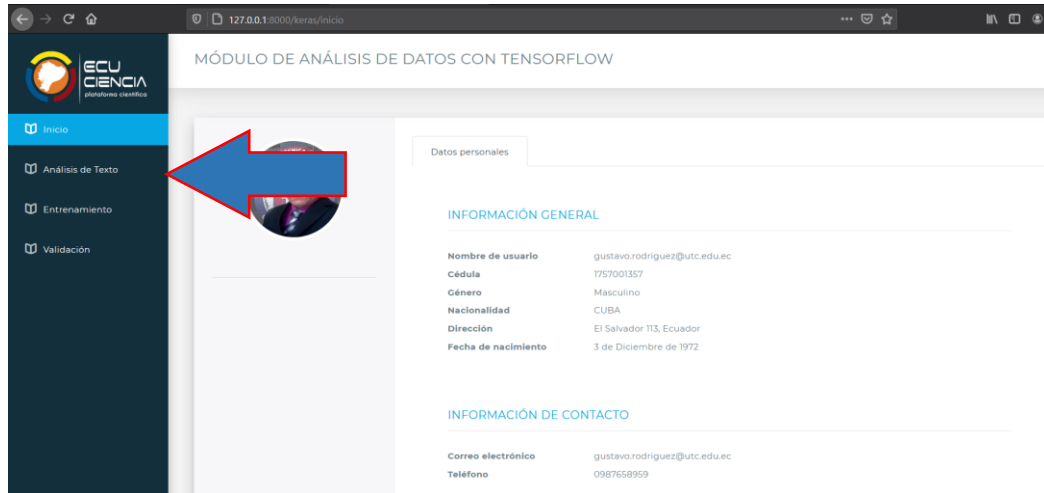
Elaborado por: Investigador

Tabla 26: Caso de Prueba 02.

CASO DE PRUEBA 02	
Identificador/es de Caso de Uso	CU03, CU04, CU05, y CU06
Nombre/s de Caso de Uso	<ul style="list-style-type: none"> • Obtener sugerencias desde el Sistema cuando se registre un nuevo documento científico de a qué Línea y Sublínea pertenece. • Realizar análisis de Minería de Texto para obtener una clasificación de Líneas y Sublíneas de Investigación. • Usar Algoritmos de Aprendizaje Profundo o Deep Learning. • Usar TensorFlow para el proceso de Análisis con Algoritmos de Aprendizaje Profundo.
Descripción de Prueba	Introducir una frase y generar una producción acertada y coherente.
Responsable	Ing. Diego Falconí.
Prerrequisito	
Realizar el proceso del Caso de Prueba 01.	
Descripción de Caso de Prueba	
Al ingresar en la opción “Análisis de Texto” se desplegará una interfaz con un formulario el mismo que tendrá dos controles principales, un input tipo Textarea en el cual se podrá escribir alguna frase en específico, y un input tipo File, el mismo que permitirá procesar archivos para análisis de texto.	
Instrucciones de prueba	

CASO DE PRUEBA 02

1. Ingresar a la opción “Análisis de Texto”.



2. Introducir una frase para el análisis, la frase será “El aprendizaje profundo o Deep Learning, es una de las aplicaciones más poderosas y de mayor crecimiento de la inteligencia artificial. Es un subcampo del aprendizaje automático que se utiliza para resolver problemas muy complejos que suelen involucrar grandes cantidades de datos.”.



CASO DE PRUEBA 02

3. Al pulsar en Analizar se podrá observar el resultado de la predicción, que para este caso debería ser “Robótica e Inteligencia Artificial”.



Respuesta esperada de la aplicación

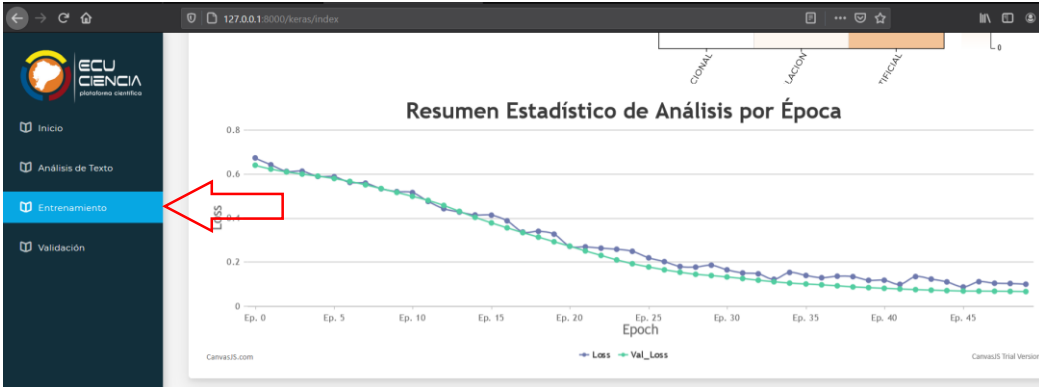
Aprueba

El módulo debe presentar como resultado de predicción al texto analizado, la clase Robótica e Inteligencia Artificial.

Si

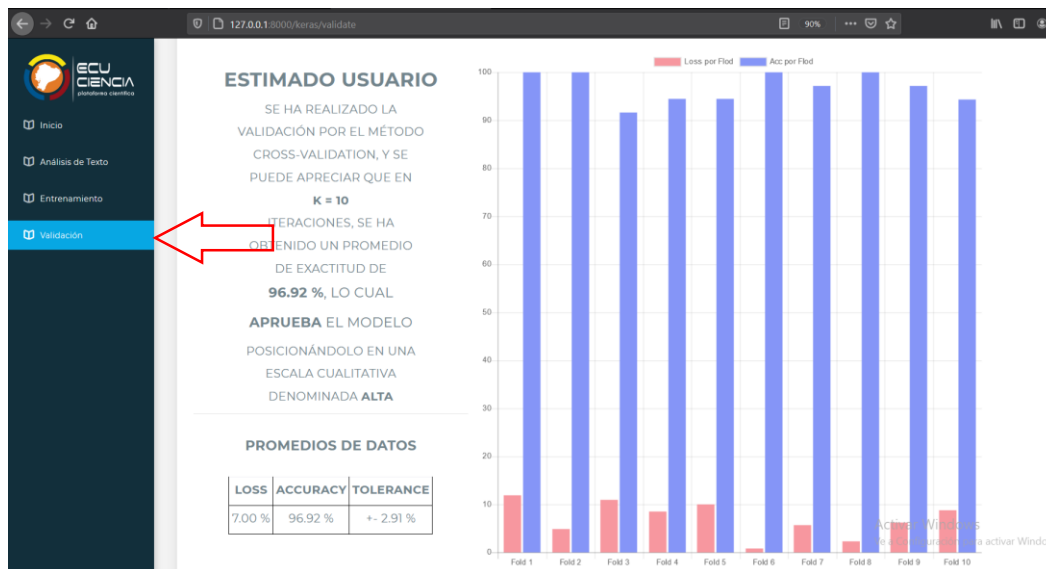
Elaborado por: Investigador

Tabla 27: Caso de prueba 03.

CASO DE PRUEBA 03	
Identificador/es de Caso de Uso	CU07
Nombre/s de Caso de Uso	<ul style="list-style-type: none"> Visualizar los resultados, tanto del porcentaje de exactitud, como de la predicción enviada por el sistema.
Descripción de Prueba	Se debe visualizar los datos en representaciones gráficas dinámicas.
Responsable	Ing. Diego Falconí.
Prerrequisito	
Realizar el proceso del Caso de Prueba 01 y 02.	
Descripción de Caso de Prueba	
El usuario deberá ser capaz de observar los resultados estadísticos dentro de gráficos dinámicos.	
Instrucciones de prueba	
1. Al realizar el proceso de “Entrenamiento”, el módulo debe representar gráficos estadísticos.	
	

CASO DE PRUEBA 03

2. Al realizar el proceso de “Validaciones”, el módulo debe representar gráficos estadísticos.



3. Al realizar el proceso de “Análisis de Texto”, el módulo debe representar gráficos estadísticos.



Respuesta esperada de la aplicación

Aprueba

El módulo debe presentar resultados en gráficos estadísticos.

Si

Elaborado por: Investigador

ANEXO VII: Consulta SQL y Resultado.

Ejecución de consulta a la base de datos de Ecuciencia para obtener información de documentos científicos. En la figura 74 se muestra la sentencia SELECT empleada para obtener la información.

```
SELECT ac.id, ac.resumen as content, sl."Nombre" as class
FROM public."Articulos_Cientificos_articulos_cientificos" ac
INNER JOIN public."Sub_Lin_Investigacion_sub_lin_investigacion" sl on sl.id = ac."SubLinea_id"
INNER JOIN public.carrera_carrera c on c.id = sl."Carrera_id"
WHERE c.id = 1
GROUP BY ac.id, sl."Nombre"
ORDER BY ac.id ASC
```

Figura 74: Sentencia SELECT de filtrado de datos de Ingeniería en Sistemas de Información

Fuente: Investigador

Al ejecutar la sentencia, se mostrará en una tabla el resultado de la consulta, se la puede apreciar en la figura 75.

id integer	content text	class character varying (300)
20	La presente...	DISEÑO, IMPLEMENTACIÓN Y...
50	Hoy en día l...	CIENCIAS INFORMÁTICAS PA...
51	Uno de los ...	ROBÓTICA E INTELIGENCIA A...
52	En el mund...	CIENCIAS INFORMÁTICAS PA...
100	Levels Of Si...	CIENCIAS INFORMÁTICAS PA...
108	El presente ...	DISEÑO, IMPLEMENTACIÓN Y...
112	La importa...	CIENCIAS INFORMÁTICAS PA...
120	En este trab...	CIENCIAS INFORMÁTICAS PA...
123	Desde hace...	CIENCIAS INFORMÁTICAS PA...
124	En los últim...	CIENCIAS INFORMÁTICAS PA...
125	En este artí...	CIENCIAS INFORMÁTICAS PA...
129	Dado que la...	CIENCIAS INFORMÁTICAS PA...
150	La investiga...	ROBÓTICA E INTELIGENCIA A...
153	En el pasad...	DISEÑO, IMPLEMENTACIÓN Y...
202	Los múltipl...	DISEÑO, IMPLEMENTACIÓN Y...
208	La ruleta te...	CIENCIAS INFORMÁTICAS PA...
209	La expresió...	CIENCIAS INFORMÁTICAS PA...
216	One of the ...	CIENCIAS INFORMÁTICAS PA...
233	El 14.VIII.01...	DISEÑO, IMPLEMENTACIÓN Y...
267	En la Socie...	ROBÓTICA E INTELIGENCIA A...

Figura 75: Resultado de consulta SELECT

Fuente: Investigador

ANEXO VIII: Logs resultantes del método de validación.

En la figura 76 se muestran los logs resultantes del análisis, en donde se aprecian los puntajes de cada iteración y el promedio general de todas las iteraciones.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Score for fold 25: loss of 0.00042407013825140893; accuracy of 100.0%
-----
Score per fold
-----
> Fold 1 - Loss: 0.1537824422121048 - Accuracy: 93.33333373069763%
-----
> Fold 2 - Loss: 0.14138759672641754 - Accuracy: 93.33333373069763%
-----
> Fold 3 - Loss: 0.08419086039066315 - Accuracy: 100.0%
-----
> Fold 4 - Loss: 0.14366339147090912 - Accuracy: 93.33333373069763%
-----
> Fold 5 - Loss: 0.08586997538805008 - Accuracy: 93.33333373069763%
-----
> Fold 6 - Loss: 0.10916611552238464 - Accuracy: 93.33333373069763%
-----
> Fold 7 - Loss: 0.014889397658407688 - Accuracy: 100.0%
-----
> Fold 8 - Loss: 0.023720042780041695 - Accuracy: 100.0%
-----
> Fold 9 - Loss: 0.0021540566813200712 - Accuracy: 100.0%
-----
> Fold 10 - Loss: 0.03214365988969803 - Accuracy: 100.0%
-----
> Fold 11 - Loss: 0.051696207374334335 - Accuracy: 92.85714030265808%
-----
> Fold 12 - Loss: 0.0018590402323752642 - Accuracy: 100.0%
-----
> Fold 13 - Loss: 0.001190961105749011 - Accuracy: 100.0%
-----
> Fold 14 - Loss: 0.17854560911655426 - Accuracy: 92.85714030265808%
-----
> Fold 15 - Loss: 0.002315213205292821 - Accuracy: 100.0%
-----
> Fold 16 - Loss: 0.10526729375123978 - Accuracy: 92.85714030265808%
-----
> Fold 17 - Loss: 0.023271696642041206 - Accuracy: 100.0%
-----
> Fold 18 - Loss: 0.003354474436491728 - Accuracy: 100.0%
-----
> Fold 19 - Loss: 0.011408503167331219 - Accuracy: 100.0%
-----
> Fold 20 - Loss: 0.006687835790216923 - Accuracy: 100.0%
-----
> Fold 21 - Loss: 0.0012561826733872294 - Accuracy: 100.0%
-----
> Fold 22 - Loss: 0.003643963485956192 - Accuracy: 100.0%
-----
> Fold 23 - Loss: 0.04287981241941452 - Accuracy: 100.0%
-----
> Fold 24 - Loss: 0.06121072173118591 - Accuracy: 92.85714030265808%
-----
> Fold 25 - Loss: 0.00042407013825140893 - Accuracy: 100.0%
-----
Average scores for all folds:
> Accuracy: 97.52380919456482 (+- 3.3046388712346317)
> Loss: 0.05143916495959275
```

Figura 76: Logs resultantes en las diferentes épocas de validación

Fuente: Investigador