



UNIVERSIDAD DE GUAYAQUIL

FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS

CARRERA DE INGENIERÍA EN SISTEMAS

COMPUTACIONALES

**DESARROLLO DE UN USER-BOT QUE PERMITA
RESPONDER PREGUNTAS DE STACK
OVERFLOW EN ESPAÑOL.**

PROYECTO DE TITULACIÓN

Previa a la obtención del Título de:

INGENIERO EN SISTEMAS COMPUTACIONALES

AUTORES:

Sánchez Piguave Alexander Sleiter
Almache Carranza Carlos Ignacio

TUTOR:

Ing. Miguel Angel Botto Tobar, Msc.

**GUAYAQUIL – ECUADOR
2020**

  	
REPOSITORIO NACIONAL EN CIENCIAS Y TECNOLOGÍAS	
FICHA DE REGISTRO DE TRABAJO DE TITULACIÓN	
TÍTULO: “ <i>DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL</i> ”	
AUTOR(ES): Sánchez Piguave Alexander Sleiter y Almache Carranza Carlos Ignacio	REVISOR(A):
INSTITUCIÓN: Universidad de Guayaquil	FACULTAD: Ciencias Matemáticas y Físicas
CARRERA: Ingeniería en Sistemas Computacionales	
FECHA DE PUBLICACIÓN:	N° DE PAGES: XXX
AREA TEMÁTICA:	
PALABRAS CLAVES: Bot, Stack Overflow, español, Telegram, Python	
<p>RESUMEN: El trabajo de investigación tuvo como finalidad desarrollar un USER-BOT que implemente algoritmos de machine learning como “Naive-Bayes” con la finalidad de responder preguntas introducidas por usuarios a través de la aplicación de mensajería instantánea Telegram, con información almacenada en la comunidad en línea Stack Overflow en español. “Stack Overflow” en español es una comunidad virtual donde miles de usuarios realizan publicaciones de preguntas, respuestas e información relacionada a los procesos de desarrollo de software en sus diferentes etapas. La búsqueda de información en la web acerca de temas de desarrollo de software puede tomar una cantidad considerable de tiempo por cuanto el usuario debe identificar, entre los miles de “post” existentes la información relevante requerida que se ajuste a su consulta. Un bot que ejecute algoritmos de machine learning que realice las tareas de explorar, clasificar, ordenar, seleccionar y presentar información que más relevancia tiene para el usuario es una ayuda considerable a las personas en su trabajo habitual. Para el desarrollo del trabajo se realizó un análisis de la estructura de las publicaciones de la comunidad Stack Overflow para ajustar un repositorio de datos acorde a ésta y se desarrollaron los procesos automáticos de extracción y procesamiento de datos para que esté disponible para el bot. Por otro lado, se desarrollaron los componentes en Python que interactúa con la información almacenada a través de algoritmos de “Naive-Bayes” para seleccionar la información relevante del repositorio. Finalmente, se configuró el bot en “Telegram” utilizando el “Bot Father” para interactuar con los usuarios. Los resultados obtenidos por el modelo fueron validados a través de la técnica de juicio de expertos en la cual 5 profesionales en el área de desarrollo de software verificaron que el USER-BOT presenta resultados útiles.</p> <p>Palabras clave: Stack Overflow, español, preguntas, bot, Python.</p>	
N° DE REGISTRO:	N° DE CLASIFICACIÓN:
DIRECCIÓN URL: (PROYECTO DE TITULACION EN LA WEB)	

ADJUNTO PDF	SI <input checked="" type="checkbox"/>	NO <input type="checkbox"/>
CONTACTO CON AUTOR(ES):	Teléfono:	Email:
CONTACTO DE LA INSTITUCIÓN	Nombre: Ab. Juan Chávez Atocha	
	Teléfono: 2307729	
	Email: juan.chaveza@ug.edu.ec	

APROBACIÓN DEL TUTOR

En mi calidad de Tutor(a) del Trabajo de Titulación, “DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL” elaborado por los Sres. Sánchez Piguave Alexander Sleiter y Almache Carranza Carlos Ignacio, **estudiantes no titulados** de la Carrera de Ingeniería en Sistemas Computacionales, Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil, previo a la obtención del Título de Ingeniero en Sistemas Computacionales, me permito declarar que luego de haber orientado, estudiado y revisado, la **apruebo** en todas sus partes.

Atentamente,

Ing. Miguel Angel Botto Tobar, Msc

TUTOR

DEDICATORIA

El presente trabajo de titulación se lo dedico a mis padres, hermanos y sobrinos quienes han sido mi pilar fundamental en el desarrollo de mi carrera profesional.

Al resto de mis familiares y amigos que de alguna forma me ayudaron a cumplir una de mis metas.

Alexander Sleiter Sánchez Piguave.

Dedico mi proyecto de titulación a Dios y a mis abuelitos que siempre han estado ahí con sus consejos, mis padres por ser la base principal en toda mi formación académica y formación personal.

Solo vencíéndote, vencerás.

Carlos Ignacio Almache Carranza

AGRADECIMIENTO

Agradezco en primer lugar a Dios, quien me ha dado la fuerza y la fe para culminar mis metas y objetivos. A mis padres, por el esfuerzo y el apoyo incondicional que me han brindado a lo largo de mi formación profesional y personal.

Alexander Sleiter Sánchez Piguave.

Más que mis abuelos, fueron las personas después de mis padres que más se preocupaban por mí. Sus canas son sinónimo de sabiduría. Me enseñaron muchas cosas vitales para la vida, y me encaminaron por el buen sendero.

Gracias abuelos.

Carlos Ignacio Almache Carranza.

TRIBUNAL PROYECTO DE TITULACIÓN

Ing. Fausto Cabrera Montes, M.Sc.
DECANO DE LA FACULTAD
CIENCIAS MATEMÁTICAS Y FÍSICAS

Ing. Gary Reyes Zambrano, Mgs.
DIRECTOR DE LA CARRERA DE
INGENIERÍA EN SISTEMAS
COMPUTACIONALES

Ing. Miguel Angel Botto Tobar, Msc
PROFESOR(A) TUTOR(A) DEL
PROYECTO
DE TITULACIÓN

Nombre y Apellidos
PROFESOR(A) REVISOR(A) DEL
PROYECTO
DE TITULACIÓN

Ab. Juan Chávez Atocha, Esp.
SECRETARIO

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este Proyecto de Titulación, me corresponden exclusivamente; y el patrimonio intelectual de la misma a la UNIVERSIDAD DE GUAYAQUIL”.

Sánchez Piguave Alexander Sleiter

Almache Carranza Carlos Ignacio



CESIÓN DE DERECHOS DE AUTOR

Ingeniero

Fausto Cabrera Montes, M.Sc.

DECANO DE LA FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS

Presente.

A través de este medio indico a usted que procedo a realizar la entrega de la cesión de derechos de autor en forma libre y voluntaria del trabajo de titulación “**Desarrollo de un USER-BOT que permita responder preguntas de Stack Overflow en español**”, realizado como requisito previo para la obtención del Título de Ingeniero(a) en Sistemas Computacionales de la Universidad de Guayaquil.

Guayaquil, Octubre de 2020.

Sánchez Piguave Alexander Sleiter
C.I. N° 0954179859

Almache Carranza Carlos Ignacio
C.I. N° 0953705787



UNIVERSIDAD DE GUAYAQUIL

FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

**DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE
STACK OVERFLOW EN ESPAÑOL**

Proyecto de Titulación que se presenta como requisito para optar por el título de
INGENIERO(A) EN SISTEMAS COMPUTACIONALES

Autor(a)(es): Sánchez Piguave Alexander Sleiter

C.I. 0954179859

Almache Carranza Carlos Ignacio

C.I. 0953705787

Tutor(a): Ing. Miguel Angel Botto Tobar, Msc

Guayaquil, Octubre de 2020

CERTIFICADO DE ACEPTACIÓN DEL TUTOR

En mi calidad de Tutor(a) del Proyecto de Titulación, nombrado por el Consejo Directivo de la Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil.

CERTIFICO:

Que he analizado el Proyecto de Titulación presentado por los estudiantes **Sánchez Piguave Alexander Sleiter y Almache Carranza Carlos Ignacio**, como requisito previo para optar por el Título de Ingeniero(a) en Sistemas Computacionales cuyo proyecto es:

DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL.

Considero aprobado el trabajo en su totalidad.

Presentado por:

Sánchez Piguave Alexander Sleiter

Cédula de identidad N° 0954179859

Almache Carranza Carlos Ignacio

Cédula de identidad N° 0953705787

Tutor(a): Ing. Miguel Angel Botto Tobar, Msc

Firma

Guayaquil, Octubre de 2020



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

AUTORIZACIÓN PARA PUBLICACIÓN DE PROYECTO DE TITULACIÓN EN FORMATO DIGITAL

1. Identificación del Proyecto de Titulación

Nombre del Estudiante: Sánchez Piguave Alexander Sleiter	
Dirección: Mapasingue Este, Coop. 14 de febrero, Mz:140, V:23	
Teléfono: 0991780043	Email: alexander.sanchezs@ug.edu.ec

Nombre del Estudiante: Almache Carranza Carlos Ignacio	
Dirección: Juan Montalvo, Coop. Pueblo y su reino, Mz:1888, V:6	
Teléfono: 0958623616	Email: carlos.almachec@ug.edu.ec

Facultad: Ciencias Matemáticas y Físicas
Carrera: Ingeniería en Sistemas Computacionales
Proyecto de Titulación al que opta: Desarrollo de un USER-BOT que permita responder preguntas de Stack Overflow en español.
Profesor(a) Tutor(a): Ing. Miguel Angel Botto Tobar, Msc.

Título del Proyecto de Titulación: Desarrollo de un USER-BOT que permita responder preguntas de Stack Overflow en español
--

Palabras Claves: Bot. Stack Overflow, español, Telegram, Python
--

2. Autorización de Publicación de Versión Electrónica del Proyecto de Titulación

A través de este medio autorizo a la Biblioteca de la Universidad de Guayaquil y a la Facultad de Ciencias Matemáticas y Físicas a publicar la versión electrónica de este Proyecto de Titulación.

Publicación Electrónica:

Inmediata	<input checked="" type="checkbox"/>	Después de 1 año	<input type="checkbox"/>
-----------	-------------------------------------	------------------	--------------------------

Firma Estudiante:

Sánchez Piguave Alexander Sleiter

Cédula de identidad N° 0954179859

Almache Carranza Carlos Ignacio

Cédula de identidad N° 0953705787

3. Forma de envío:

El texto del Proyecto de Titulación debe ser enviado en formato Word, como archivo .docx, .RTF o .Puf para PC. Las imágenes que la acompañen pueden ser: .gif, .jpg o .TIFF.

DVDROM

CDROM

ÍNDICE GENERAL

FICHA DE REGISTRO DE TRABAJO DE TITULACIÓN	II
APROBACIÓN DEL TUTOR.....	IV
DEDICATORIA.....	V
AGRADECIMIENTO	VI
TRIBUNAL PROYECTO DE TITULACIÓN	VII
DECLARACIÓN EXPRESA.....	VII
CESIÓN DE DERECHOS DE AUTOR	VIII
CERTIFICADO DE ACEPTACIÓN DEL TUTOR	X
AUTORIZACIÓN PARA PUBLICACIÓN DE PROYECTO DE TITULACIÓN EN FORMATO DIGITAL	XI
ÍNDICE GENERAL	XII
ÍNDICE DE TABLAS.....	XVI
ÍNDICE DE FIGURAS.....	XVII
ABREVIATURAS.....	XVIII
SIMBOLOGÍA	XIX
RESUMEN.....	XX
ABSTRACT.....	XXI
INTRODUCCIÓN	60
CAPÍTULO I.....	63

PLANTEAMIENTO DEL PROBLEMA	63
Descripción de la situación problemática	63
Ubicación del problema en un contexto.....	63
Situación conflicto nudos críticos	65
Delimitación del problema.....	66
Evaluación del Problema	66
Causas y consecuencias del problema	68
Formulación del problema	69
Objetivos del proyecto	69
Objetivo general.....	69
Objetivos específicos	69
Alcance del proyecto	70
Justificación e importancia	70
Limitaciones del estudio	71
CAPÍTULO II	72
MARCO TEÓRICO	72
Antecedentes del estudio.....	72
Fundamentación teórica.....	73
Revisiones sistemáticas	108
Hipótesis.....	120

Variables de la investigación.....	120
CAPÍTULO III.....	121
METODOLOGÍA DE LA INVESTIGACIÓN	121
Tipo de investigación	122
Diseño metodológico de la investigación.....	123
Procesamiento y análisis.....	132
Beneficiarios directos e indirectos del proyecto	133
Criterios de validación de la propuesta	133
CAPÍTULO IV	139
CONCLUSIONES Y RECOMENDACIONES.....	139
Conclusiones	139
Recomendaciones	140
Trabajos futuros.....	141
REFERENCIAS BIBLIOGRÁFICAS.....	142
ANEXOS.....	147
Anexo 1. Planificación de actividades del proyecto.....	148
Anexo 2. Fundamentación Legal	149
Anexo 3. Criterios éticos a utilizarse en el desarrollo del proyecto.....	153
Anexo 4. Validación de expertos.....	154
Anexo 5. Estudios Seleccionados	164

Anexo 6. Acta de entrega y recepción definitiva.....	166
Anexo 7. Artículo científico	167

ÍNDICE DE TABLAS

Tabla 1 Delimitación del problema.....	66
Tabla 2 Matriz de causas y consecuencias del problema.....	68
Tabla 3 Sub-preguntas y criterios.	114
Tabla 4 Selección de artículos.	118
Tabla 5 Resultados del mapeo sistemático	119
Tabla 6 de contingencia. Pregunta #5	134
Tabla 7 de contingencia. Pregunta #6	135
Tabla 8 de contingencia. Pregunta #7	135
Tabla 9 de contingencia. Pregunta #8	136
Tabla 10 Estudios Utilizados en el Mapeo Sistemático.....	164

ÍNDICE DE FIGURAS

Gráfico 1 Sistema Operativo.....	74
Gráfico 2 Ecosistemas de software.....	86
Gráfico 3 Modelo Conceptual para el desarrollo de ecosistemas de software.	87
Gráfico 4 Clasificación de la Inteligencia artificial.	103
Gráfico 5 Clasificación de machine learning.....	106
Gráfico 6 Etapas Mapeo Sistemático.....	109
Gráfico 7 Base de datos y la tabla creada.	124
Gráfico 8 Proceso de extracción con USER-BOT.....	125
Gráfico 9 Resultado.	127
Gráfico 10 Registro del bot.....	128
Gráfico 11 Conexión a Telegram.....	129
Gráfico 12 Proceso de consulta con USER-BOT.	129
Gráfico 13 Interfaz User-Bot.	131
Gráfico 14 Estructura del programa en zenodo	132
Gráfico 15 Dataset1	137
Gráfico 16 Valores Dataset1	137
Gráfico 17 Dataset2	138
Gráfico 18 Valores Dataset2.....	138

ABREVIATURAS

ABP	Aprendizaje Basado en Problemas
CC.MM.FF	Facultad de Ciencias Matemáticas y Físicas
EDT	Estructura de Desglose de Trabajo
FTP	Archivos de Transferencia
g.l.	Grados de Libertad
HTML	Lenguaje de Marca de salida de Hyper Texto
HTTP	Protocolo de transferencia de Hyper Texto
Ing.	Ingeniero
ISP	Proveedor de Servicio de Internet
M.Sc.	Máster
Mtra.	Maestra
UG	Universidad de Guayaquil
URL	Localizador de Fuente Uniforme
WWW	World Wide Web (Red Mundial)

SIMBOLOGÍA

s	Desviación estándar
e	Error
E	Espacio muestral
$E(Y)$	Esperanza matemática de la v.a. y
s	Estimador de la desviación estándar
e	Exponencial



**UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**DESARROLLO DE UN USER-BOT QUE PERMITA
RESPONDER PREGUNTAS DE STACK
OVERFLOW EN ESPAÑOL.**

Autor(a)(es): Sánchez Piguave Alexander Sleiter
C.I. N° 0954179859

Almache Carranza Carlos Ignacio
C.I. N° 0953705787

Tutor(a): Ing. Miguel Angel Botto Tobar, Msc.

RESUMEN

El trabajo de investigación tuvo como finalidad desarrollar un USER-BOT que implemente algoritmos de machine learning como “Naive-Bayes” con la finalidad de responder preguntas introducidas por usuarios a través de la aplicación de mensajería instantánea Telegram, con información almacenada en la comunidad en línea “Stack Overflow” en español. Stack Overflow en español es una comunidad virtual donde miles de usuarios realizan publicaciones de preguntas, respuestas e información relacionada a los procesos de desarrollo de software en sus diferentes etapas. La búsqueda de información en la web acerca de temas de desarrollo de software puede tomar una cantidad considerable de tiempo por cuanto el usuario debe identificar, entre los miles de “post” existentes la información relevante requerida que se ajuste a su consulta. Un robot que ejecute algoritmos de machine learning que realice las tareas de explorar, clasificar, ordenar, seleccionar y presentar información que más relevancia tiene para el usuario es una ayuda considerable a las personas en su trabajo habitual. Para el desarrollo del trabajo se realizó un análisis de la estructura de las publicaciones de la comunidad Stack Overflow para ajustar un repositorio de datos acorde a ésta y se desarrollaron los procesos automáticos de extracción y procesamiento de datos para que esté disponible para el robot. Por otro lado, se desarrollaron los componentes en Python que interactúa con la información almacenada a través de algoritmos de “Naive-Bayes” para seleccionar la información relevante del repositorio. Finalmente, se configuró el robot en “Telegram” utilizando el “Bot Father” para interactuar con los usuarios. Los resultados obtenidos por el modelo fueron validados a través de la técnica de juicio de expertos en la cual 5 profesionales en el área de desarrollo de software verificaron que el USER-BOT presenta resultados útiles.

Palabras clave: Stack Overflow, español, preguntas, bot, Python.



**UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

DEVELOPMENT OF A USER-BOT TO ANSWER QUESTIONS FROM STACK OVERFLOW IN SPANISH.

Autor(a)(es): Sánchez Piguave Alexander Sleiter
C.I. N° 0954179859

Almache Carranza Carlos Ignacio
C.I. N° 0953705787

Tutor(a): Ing. Miguel Angel Botto Tobar, Msc

ABSTRACT

The purpose of the research work was to develop a USER-BOT that implements machine learning algorithms such as "Naive-Bayes" in order to answer questions entered by users through the instant messaging application Telegram, with information stored in the community in "Stack Overflow" line in Spanish. "Stack Overflow" in Spanish is a virtual community where thousands of users post questions, answers, and information related to software development processes in its different stages. Searching for information on the web about software development issues can take a considerable amount of time because the user must identify, among the thousands of existing "posts", the relevant information required that fits her query. A bot that runs machine learning algorithms that performs the tasks of exploring, classifying, sorting, selecting and presenting information that is most relevant to the user is a considerable help to people in their daily work. For the development of the work, an analysis of the structure of the Stack Overflow community publications was carried out to adjust a data repository according to this and the automatic data extraction and processing processes were developed so that it is available to the bot. On the other hand, the components were developed in Python that interacts with the stored information through "Naive-Bayes" algorithms to select the relevant information from the repository. Finally, the bot was configured in "Telegram" using the "Bot Father" to interact with users. The results obtained by the model were validated through the expert judgment technique in which 5 professionals in the software development area verified that the USER-BOT presents useful results.

Key words: Stack Overflow, spanish, questions, bot, Python.

INTRODUCCIÓN

Durante el desarrollo de software en cualquier lenguaje de programación o plataforma tecnológica, es común que se presenten una serie de problemas técnicos en la compilación / ejecución del código creado por el profesional que causa que se consuma una gran cantidad de tiempo del programador tratando de corregir los errores o subsanar los problemas planteados. También suele ocurrir que los miembros del equipo de desarrollo desconozcan ciertos aspectos de las herramientas que deben utilizar durante el proceso lo cual puede ser compensado con la información a la cual pueda acceder para solventar la falta de información. Inclusive los programadores más experimentados pueden encontrarse en el proceso con algún inconveniente técnico que no le permita seguir avanzando en el desarrollo del sistema. Para corregir los errores o enmendar los problemas técnicos presentados, el programador habitualmente acude a la documentación del proveedor, si existiere, o a la información publicada que se ha puesto a disposición de la comunidad. La última opción sirve mucho, sin embargo, la información obtenida debe ser evaluada adecuadamente por cuanto los usuarios de internet podrían subir información que no necesariamente es acertada al problema planteado, causando pérdida de tiempo a los que usuarios que realizan la consulta.

En internet existe una serie de sitios web que permiten publicar información muy variada en cuanto a los temas que tratan y que están a disposición de las personas, como por ejemplo la enciclopedia virtual “Wikipedia” que permite a los usuarios agregar información de una amplia gama de temáticas. Sin embargo, la información registrada puede contener inconsistencias fundamentado o las referencias bibliográficas acordes a la realidad o carezcan de fundamento teórico – técnico lo que lo convierte en una información no útil para los intereses del programador que consulta.

Hace muchos años uno de los buscadores más populares de internet denominado Yahoo ha venido implementando una plataforma de “preguntas y respuestas” denominado “Yahoo respuestas” en la cual la comunidad aporta a la plataforma con respuestas a preguntas registradas previamente. Actualmente esta plataforma se encuentra disponible en 12 idiomas. En esta herramienta las preguntas se ordenan por categorías para poder mejorar el tiempo de consulta.

En el año 2008 se funda “Stack Overflow” que es la comunidad en línea más grande y confiable para cualquier persona que codifique para aprender, compartir sus conocimientos y desarrollar sus carreras. Más de 50 millones de visitantes únicos acceden a la plataforma cada mes para ayudar a resolver problemas de codificación, desarrollar nuevas habilidades y encontrar oportunidades de trabajo. Stack Overflow se asocia con empresas para ayudarles a comprender, contratar, involucrar y habilitar a los desarrolladores del mundo. En esta plataforma los desarrolladores pueden compartir sus experiencias, realizar preguntas a errores de codificación y sus soluciones, además de mejorar sus habilidades en distintos lenguajes de programación. Uno de los aspectos más

relevantes del sistema es lo que denomina “Reputación” cuyo término representa una mayor confianza a las respuestas en base a los votos de los usuarios.

El presente trabajo de titulación está constituido por 4 capítulos que se detallan a continuación:

Capítulo I: Se realiza el análisis de la problemática planteada en el estudio, los objetivos generales y específicos de la investigación, los alcances, la importancia y justificación del proyecto.

Capítulo II: Se detalla el marco teórico sobre el cual se fundamenta el estudio, así como el marco legal de la investigación, determinando la no violación de ninguna ley o reglamento del Ecuador.

Capítulo III: Se presenta la metodología utilizada en la investigación, la factibilidad del proyecto y los parámetros de evaluación de la propuesta.

Capítulo IV: Se presenta los resultados, conclusiones y recomendaciones del estudio.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

Descripción de la situación problemática

Ubicación del problema en un contexto

Durante el proceso de desarrollo de software se pueden presentar diferentes problemas tales como: error de código, error en el funcionamiento de la aplicación o resultados inesperados en los cálculos y muchos más eventos que pueden afectar la calidad del software que se está desarrollando. Los problemas en el desarrollo de software se pueden presentar en las distintas fases del proceso como son, fase de desarrollo, implementación y pruebas. Cuando los problemas en el proceso de desarrollo de software se presentan, los programadores, en base a su experiencia, los resuelven utilizando los conocimientos que han adquirido durante su carrera profesional. Sin embargo, en ciertos casos, el programador debe acudir a información adicional, ya sea la documentación de la herramienta que está utilizando, foros y portales web especializados. La búsqueda de información relacionada al proceso de desarrollo de software puede ser una tarea compleja y requiere la inversión de tiempo de investigación, además de, al existir una gran cantidad de información, el profesional deberá validar la que mejor se ajusta a sus necesidades.

Esto ha dado lugar a que se vayan creando comunidades en línea que contribuyen con información técnica del proceso de desarrollo de software, definiciones conceptuales, código, herramientas de programación, validación y puesta en marcha de sistemas, además de posibles

soluciones a diferentes problemas que se puede ir presentando durante el proceso del software. Estas comunidades digitales se encuentran en constante evolución y crecimiento por la gran ayuda que estas representan en el desarrollo de software por el acceso libre a la información proporcionada, ya que toda toda contribución de un usuario está licenciada bajo [Atribución-CompartirIgual de Creative Commons](#). Un sitio web que brinde información técnica confiable al equipo de desarrollo de software contribuye al desarrollo efectivo de los sistemas minimizando el tiempo de búsqueda de dicha información o de resolución de problemas presentados.

Stack Overflow ha puesto a disposición de la comunidad de habla hispana su versión del sitio en español ⁶⁵ con la finalidad de proveer los servicios antes mencionados a los países hispanohablantes, denominado “Stack Overflow en español”.

La comunidad en línea Stack Overflow en español es un popular sitio web que posee contenido relacionado a temas de desarrollo de software y tecnologías de la información y comunicaciones, aportadas por los usuarios de la comunidad a través de la formulación de preguntas y respuestas.

Situación conflicto nudos críticos

La consulta de información en comunidades en línea, como Stack Overflow en español, puede ser tan sencillo como acceder a un sitio web y digitar la consulta requerida o tan difícil como buscar información de un error específico entre muchas páginas web o varios posts publicados. Las comunidades en línea realizan ingentes esfuerzos para mantener funcionando adecuadamente sus sistemas y que la información presentada sea adecuada y completa.

El tiempo que los usuarios toman en la búsqueda de respuestas adecuadas para resolver sus interrogantes es alto por tener que consultar de manera “manual” al navegar por las diferentes publicaciones del sitio web. Este tiempo invertido podría afectar al normal desempeño de los profesionales que acuden al sitio en búsqueda de respuestas a diversos problemas tecnológicos.

Las comunidades en línea ofrecen a sus usuarios información relativa a todas las etapas del desarrollo de software que son aportadas directamente por sus colaboradores; Sin embargo, la información que registran debe cumplir ciertas características de precisión, calidad y exactitud, por lo que pasa por un proceso de validación a través de algún mecanismo que permita garantizar su calidad. Algunas comunidades en línea tienen procesos de calificación de la información (preguntas, respuestas, información técnica en general, código) que permite posicionarla entre las “más recomendadas”. Esta calificación la realizan los miembros de la comunidad lo que aporta de manera significativa acortando el tiempo de búsqueda.

Delimitación del problema

El estudio se limita a la investigación de la estructura y particularidades de la comunidad en línea “Stack Overflow en español” a través de un análisis de la organización de información en el sitio web, disponible en la web como una herramienta de consulta para profesionales del desarrollo de software. Así también, el desarrollo de un modelo “USER-BOT” que implemente algoritmos de machine learning para que presente información de la comunidad en línea que responda las preguntas que el usuario haya registrado en la aplicación de mensajería instantánea “Telegram” de manera adecuada, con información de la comunidad en línea Stack Overflow en español.

Tabla 1 Delimitación del problema

Delimitador	Descripción
Campo	Tecnología
Área	Desarrollo de software
Aspecto	Proceso de desarrollo de software
Tema	Desarrollo de un USER-BOT que permita responder preguntas de Stack Overflow en español.

Nota: Delimitación del problema.

Fuente: Elaboración propia.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

Evaluación del Problema

- **Delimitado:** El trabajo presentado esta delimitado al desarrollo de un modelo USER-BOT a través de aprendizaje automático que permita responder preguntas con la información de la comunidad en línea Stack Overflow en español. No está considerado en el presente trabajo otras comunidades en línea y, además, la interfaz de

comunicación entre el usuario y el bot será exclusivamente a través de la aplicación de mensajería instantánea “Telegram”.

- **Claro:** La comunidad en línea especializada en temas relacionados al desarrollo de software “Stack Overflow en español” ha tenido un crecimiento sostenido en el tiempo convirtiéndose en una herramienta utilizada por los usuarios relacionados a las actividades de desarrollo de software, ya sea de manera profesional o por interés de aprendizaje. Por lo anterior, un modelo USER-BOT que realice la consulta de manera automática a través de una interfaz amigable con el usuario tendría un impacto positivo en los miembros de la comunidad.
- **Concreto:** El estudio se concreta en el desarrollo de un modelo USER-BOT a través de aprendizaje automático que tenga la capacidad de interactuar con la información almacenada de Stack Overflow en español y, además, presentarla en una interfaz amigable a través de la aplicación de mensajería instantánea “Telegram”.
- **Relevante:** La información publicada en la comunidad en línea especializada en temas de desarrollo de software “Stack Overflow en español” ha permitido a millones de usuarios solventar dudas, superar problemas, completar programas, entre otras actividades. El uso de un modelo USER-BOT que automatice la búsqueda y selección de información de la comunidad en línea será una gran ayuda para sus usuarios que verán reducidos sus tiempos de consulta y obtendrán resultados adecuados. Lo anterior permitirá que los integrantes del equipo de desarrollo de software realicen sus actividades eficientemente.

- **Original:** El estudio plantea una alternativa amigable para responder preguntas relacionada al proceso de desarrollo de software en la comunidad Stack Overflow en español y que sea presentada en un dispositivo móvil donde se encuentre instalada la aplicación telegram, utilizando mecanismos de comunicación entre las diferentes tecnologías utilizadas para facilitar el acceso a información relevante.

Identifica los productos esperados:

- USER-BOT para interacción del usuario con la comunidad en línea denominada Stack Overflow.

Causas y consecuencias del problema

Tabla 2 Matriz de causas y consecuencias del problema.

Causas	Consecuencias
La información técnica o respuestas sobre problemas relacionados al proceso de desarrollo de software están dispersos en sitios web especializados.	Consumo de gran cantidad de tiempo en la búsqueda de información
La información registrada en las comunidades en línea no está validada por ningún organismo de control de contenido.	Las respuestas obtenidas no solucionan el problema planteado.
No existe un tiempo promedio de respuesta para las consultas registrada por los usuarios.	Perdida de interés del usuario hacia la comunidad en línea.
Existe información registrada en la comunidad en línea de lenguaje o versiones de software antiguas.	La información de las comunidades en línea especializadas en el proceso de desarrollo de software queda obsoleta.

Nota: Se detalla matriz de causas y consecuencias del problema.

Fuente: Elaboración propia.

Formulación del problema

¿El desarrollo de un modelo USER-BOT que responda preguntas con información obtenida de la comunidad en línea Stack Overflow en español permitirá a los desarrolladores de software acceder fácilmente a contenidos relacionados como apoyo a sus actividades profesionales?

Objetivos del proyecto

Objetivo general

Desarrollar un modelo USER-BOT mediante técnicas de aprendizaje automático para que responda preguntas de la comunidad en línea Stack Overflow en español.

Objetivos específicos

1. Diseñar un modelo de base de datos acorde a la estructura de la información publicada en la comunidad en línea Stack Overflow en español.
2. Recopilar la literatura relacionada a los temas de estudio para comprender los aspectos teóricos y prácticos relacionados con la presente investigación.
3. Desarrollar un modelo de aprendizaje automático que responda preguntas de la comunidad en línea Stack Overflow en español.
4. Analizar los resultados obtenidos del modelo USER-BOT que valide el correcto funcionamiento del modelo USER-BOT.

Alcance del proyecto

Al finalizar el desarrollo del proyecto de titulación se presentará como resultado los siguientes elementos.

1. Se realizará el diseño estructural de la comunidad en línea de Stack Overflow en español mediante el análisis del formato de almacenamiento de la información.
2. Se desarrolló un marco conceptual a través de la recopilación de la literatura relacionada al aprendizaje automático donde se resaltan los aspectos teóricos y prácticos.
3. Un modelo de aprendizaje automático que responde preguntas con información de la comunidad en línea Stack Overflow en español a través de la aplicación de mensajería instantánea Telegram.
4. Se realiza el análisis de los resultados obtenidos del modelo USER-BOT a través de consultas a expertos con la finalidad de validar el correcto funcionamiento del *bot*.

Justificación e importancia

El desarrollo de un modelo USER-BOT que implemente algoritmos de aprendizaje automático con la finalidad de obtener información de los posts registrados en la comunidad en línea Stack Overflow en español, para responder a cuestiones de consultas de usuarios, se convertiría en un apoyo para los usuarios de la comunidad. El modelo planteado apoya a las búsquedas de información que realizan los usuarios de la comunidad.

Limitaciones del estudio

Durante el desarrollo del presente trabajo de titulación existen posibles dificultades en limitar el alcance y la ejecución de los objetivos de la investigación los cuales se detallan a continuación:

- Corto tiempo asignado al desarrollo del proyecto de investigaciones.
- Datos obtenidos de la comunidad en línea Stack Overflow en español corresponde a la fecha 22/07/2020.
- Por el tiempo limitado no se pudo contactar con usuarios de la comunidad en línea Stack Overflow en español.
- La construcción del dataset se limitó a las preguntas no contestadas de la comunidad en línea Stack Overflow en español con etiqueta MySQLi.
- Para el desarrollo de las pruebas del USER-BOT, el mismo se encuentra en funcionamiento en un servidor local.

CAPÍTULO II

MARCO TEÓRICO

Antecedentes del estudio

En el trabajo realizado por Martínez y otros, denominado “Análisis de calidad y recuperación de información en foros de discusión” mencionan que “herramientas colaborativas como los blogs, wikis y foros de discusión hacen posible el reúso de conocimiento disponible en la Web” (Martínez, y otros, 2018). En el mencionado trabajo, los autores realizan la clasificación de contenido de los mencionados sitios web a través de diferentes métodos de clasificación, entre los cuales destaca algoritmos de aprendizaje automático. Sin embargo, el trabajo citado no posee una interfaz de comunicación con el usuario final y se limita a obtener las sugerencias de los hilos de discusión que forman las preguntas y respuesta de los sitios web organizados y categorizados de acuerdo con las reglas y métricas establecidas por los autores.

En el trabajo de Mora (2018) sobre el “Desarrollo de un sistema de Recuperación de Información para listas de preguntas frecuentes” realiza el desarrollo de un sistema de Recuperación de Información (RI) que haga la consulta en las listas de preguntas frecuentes (FAQ) que los administradores de sitios web mantienen para no saturar los sistemas. La herramienta planteada busca tomar la pregunta del usuario y, aplicando técnicas de procesamiento de lenguaje natural, busca si dicha pregunta se asemeja a otra realizada previamente. Este proyecto no abarca en el contenido de algún sitio web o comunidad en especial sino a las listas de preguntas frecuentes mantenidas por diversos sitios.

Fundamentación teórica

Según Wolf, Ruiz, Bergero y Meza determinan que un Sistema Operativo es “el principal programa que se ejecuta en toda computadora de propósito general” (Wolf, Ruiz, Bergero, & Meza, 2015). Así también que existen sistemas muy simples como también otros de muy alta complejidad. El sistema operativo es el único programa que interactúa directamente con el hardware de la computadora. Así también, Stalling lo define como “el programa que controla la ejecución de aplicaciones y programas y que actúa como interfaz entre las aplicaciones y el hardware del computador” (Stalling, 2005).

Stalling determina tres aspectos básicos de todos los sistemas operativos que son:

- **Facilidad de uso:** el sistema operativo facilita el uso del computador
- **Eficiencia:** Permite que los recursos de un sistema computacional se puedan utilizar de una manera eficiente.
- **Capacidad para evolucionar:** El sistema operativo está construido con la finalidad de que pueda desarrollarse, probar e introducir nuevas funcionalidades sin interferir en los servicios que brinda.

Servicios que presta un sistema operativo

Stalling clasifica los servicios que presta un sistema operativo en seis grandes grupos que son (Stalling, 2005):

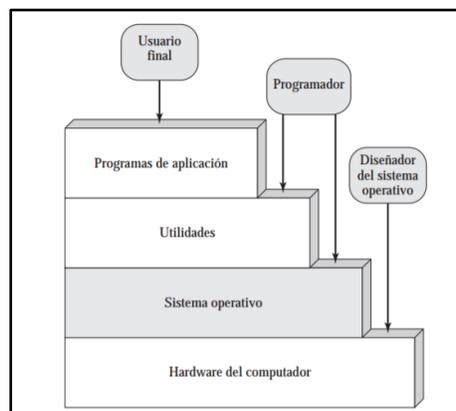
- **Desarrollo de programas:** Proporciona utilidades y servicios para asistir al programador para la creación de los programas.
- **Ejecución de programas:** Realiza las labores de planificación, carga del programa en

memoria principal e inicializar los dispositivos de entrada -salida y ficheros relacionados.

- **Acceso a dispositivos de entrada – salida:** Proporciona una interfaz uniforme para que los programas puedan utilizar los dispositivos de entrada salida en sus instrucciones.
- **Acceso controlado a los ficheros:** Proporciona los mecanismos para el acceso a los dispositivos de almacenamiento.
- **Acceso al sistema:** Controla los accesos de cada uno de los usuarios y el acceso a cada uno de los recursos y datos.
- **Detección y respuesta de errores:** Detecta los errores que se pueden presentar tanto de Hardware como de Software y determina las acciones a seguir para eliminar la condición de error que permita que el sistema continúe funcionando.

En el Grafico 1 se puede observar que el sistema operativo es la base sobre la cual se ejecutan los demás programas de aplicación y, además, realiza las tareas de intermediario para que el usuario y/o los programas puedan utilizar el hardware computacional.

Gráfico 1 Sistema Operativo



Nota: Niveles y vistas de un Sistema Operativo.

Fuente: Sistemas operativos.

Autor: Stalling William.

Ingeniería de software

Según Sommerville la Ingeniería de software es “es una disciplina de la ingeniería cuya meta es el desarrollo costeable de sistemas de software. Éste es abstracto e intangible. No está restringido por materiales, o gobernado por leyes físicas o por procesos de manufactura. De alguna forma, esto simplifica la ingeniería del software al no existir limitaciones físicas del potencial del software” (Sommerville, 2015). Entonces, podemos inferir que la meta de la ingeniería de software es desarrollar programas informáticos de máxima calidad que cumplan con las necesidades identificadas de la organización a la cual se sirve dentro del presupuesto económico y de tiempos planificado.

Además, Sommerville menciona, que “no hay un enfoque «ideal» a la ingeniería del software. La amplia diversidad de tipos de sistemas y organizaciones que usan estos sistemas significa que necesitamos una diversidad de enfoques al desarrollo de software. Sin embargo, las nociones fundamentales de procesos y la organización del sistema son la base de todas estas técnicas, y éstas son la esencia de la ingeniería del software” (Sommerville, 2015). Se puede inferir entonces que no existe un modelo de desarrollo único para todos los tipos de proyectos informáticos sino más bien que depende de la complejidad, alcance y las características específicas de la aplicación que se requiere desarrollar. La necesidad de cada sistema es distinta y, así mismo, las estrategias también varían de acuerdo con el equipo de trabajo de acuerdo con la metodología aplicada.

Actividades del proceso de software

El desarrollo de software es un conjunto de actividades íntimamente relacionadas que se ejecutan de manera ordenada y planificada. Estas actividades están agrupadas en fases o etapas, las cuales

Sommerville las define de la siguiente manera:

Especificación del software: “La especificación del software o ingeniería de requerimientos es el proceso de comprensión y definición de qué servicios se requieren del sistema y de identificación de las restricciones de funcionamiento y desarrollo de este” (Sommerville, 2015). En esta primera fase el equipo de desarrollo realiza las acciones iniciales del proyecto con la finalidad de determinar la viabilidad técnica, determinar los requerimientos del cliente (funcionalidad esperada por el cliente), la especificación y validación de dichos requerimientos. Es importante esta primera fase por cuanto se define lo que se espera del sistema.

Dentro de esta primera fase de desarrollo de software se agrupan las siguientes acciones:

- 1. Estudio de viabilidad.** Corresponde al estudio técnico que determina si las necesidades de los usuarios pueden ser solventadas a través del desarrollo del software con la utilización de herramientas existentes.
- 2. Obtención y análisis de requerimientos.** Conjunto de actividades orientadas a recopilar información sobre de las necesidades de funcionalidad que los clientes esperan del sistema, también denominados “requerimientos”.
- 3. Especificación de requerimientos.** Se traduce la información recopilada durante la etapa previa y su resultado es un documento donde se definen los requerimientos.
- 4. Validación de requerimientos.** En esta etapa se comprueba la veracidad, consistencia y completitud de los requerimientos. Inevitablemente se descubren errores en el documento de requerimientos.

Diseño e implementación de software

De acuerdo con Sommerville, se puede definir la fase de diseño e implementación de software como “el proceso de convertir una especificación del sistema en un sistema ejecutable” (Sommerville, 2015, pág. 71). En esta fase, los requerimientos que se encuentran determinados en el documento de definición realizado previamente se traducen en programas computacionales utilizando un lenguaje de programación, con la finalidad de proveer las funcionalidades esperadas por los usuarios. Es necesario que las funcionalidades desarrolladas cubran los requerimientos identificados. “Un diseño de software es una descripción de la estructura del software que se va a implementar, los datos que son parte del sistema, las interfaces entre los componentes del sistema y, algunas veces, los algoritmos utilizados” (Sommerville, Ingeniería del Software, 2015, pág. 71). Esta definición implica los modelos que se debe realizar para representar cada uno de los componentes del sistema, sus relaciones y lógica de acción.

El desarrollo de software es una actividad posterior a las actividades de diseño y tiene que ver con que los programadores escriban las instrucciones computacionales en un lenguaje de programación previamente seleccionado con la finalidad de que el sistema realice las funciones para las cuales está siendo creado. Este desarrollo debe seguir un conjunto de reglas y normas que permitan escribir adecuadamente el código lo cual redundará en una facilidad de lectura y mantenimiento del sistema. De acuerdo con (Sommerville, 2015) “la programación es una actividad personal y no existe un proceso general que se siga comúnmente”. El éxito de esta actividad depende completamente de las habilidades y conocimientos del programador, así como su capacidad de ir resolviendo problemas de codificación que pudieran presentarse.

Validación del software

Sommerville menciona que “la validación del software o, de forma más general, la verificación y validación se utiliza para mostrar que el sistema se ajusta a su especificación y que cumple las expectativas del usuario que lo comprará” (Sommerville, Ingeniería del Software, 2015). Durante esta fase se realiza una comparación de las funciones desarrolladas en el sistema y los requerimientos determinados en la primera fase para verificar que todos los aspectos han sido considerados y, además, que cada operación se realice correctamente.

Evolución del software

Los negocios se encuentran en constante evolución y las necesidades de sistemas que se determinaron al inicio del proyecto pueden cambiar o aumentar con el pasar del tiempo, es por ello que los sistemas deben tener la capacidad de ir evolucionando en conjunto con el negocio. La evolución del software es necesaria a través de ajustes o la implementación de nuevas funciones al sistema.

Modelo del proceso de software

Según (Sommerville, Ingeniería del Software, 2015) el modelo de los procesos de software “es una representación abstracta de un proceso del software. Cada modelo de proceso representa un proceso desde una perspectiva particular, y así proporciona sólo información parcial sobre ese proceso”. Se pueden definir ciertos modelos generales los cuales son abstracciones que se pueden utilizar para explicar diferentes enfoques para el desarrollo de software. Estos modelos generales nos proporcionan una perspectiva de alto nivel sobre los pasos a seguir y el marco de trabajo para el desarrollo de software.

Los modelos del proceso de software generales son:

- Modelo en cascada
- Desarrollo evolutivo
- Ingeniería de software basada en componentes

Iteración de Procesos

Los cambios son inevitables en el desarrollo de proyectos de software grandes, por diferentes causales pueden cambiar los requerimientos del sistema, ya sea por factores internos o externos de la institución que requiere el sistema. Si cambia el estilo administrativo, el entorno legal, las estrategias empresariales, las herramientas tecnológicas, los diseños, etc., harán que los requerimientos del sistema se modifiquen. Por lo anterior el proceso del software “no es un proceso único; más bien, las actividades del proceso se repiten regularmente conforme el sistema se rehace en respuesta a peticiones de cambios”. (Sommerville, Ingeniería del Software, 2015, pág. 66).

Según Sommerville (2015), “el desarrollo iterativo es fundamental para el desarrollo de software y se pueden presentar en dos modelos que son”:

1. **Entrega incremental.** La especificación, el diseño y la implementación del software se dividen en una serie de incrementos, los cuales se desarrollan por turnos;
2. **Desarrollo en espiral.** El desarrollo del sistema gira en espiral hacia fuera, empezando con un esbozo inicial y terminando con el desarrollo final del mismo.

Aspectos sociales de la ingeniería de software

La ingeniería de software es “es una disciplina de la ingeniería cuya meta es el desarrollo costeable

de sistemas de software” (Sommerville, Ingeniería del Software, 2015) a través de la ejecución de una serie de actividades ordenadas e interrelacionadas con la finalidad de desarrollar sistemas funcionales. Una de las etapas del proceso de desarrollo es la ingeniería de requerimientos en la cual se trata de entender el con precisión el problema que se requiere resolver. De acuerdo con Donoso y Vega, “múltiples factores afectan este proceso, destacando el aspecto social y emocional de la persona, que no son considerados comúnmente por técnicas de educación tradicionales” (Donoso & Vega, 2017).

De acuerdo con Donoso y Vega, la ingeniería de requerimientos “es un proceso intensivo de transferencia de información entre quienes son «poseedores del conocimiento» desde el lado del cliente-usuario y el equipo de desarrollo. Este conocimiento está incrustado casi en su mayoría en los stakeholders y es labor del especialista en requerimientos poder abstraer este conocimiento apoyado por procesos cognitivos, emocionales y motivacionales” (Donoso & Vega, 2017). Los autores determinan una lista de “factores sociales y humanos provenientes desde analistas y clientes que pueden afectar el proceso de educación y, en contra parte, otorgar una base para su consideración al momento de efectuar actividades de ingeniería de requerimientos” (Donoso & Vega, 2017).

De acuerdo con Donoso y Vega, los factores humanos y sociales que tienen mayor incidencia en el proceso de desarrollo de software son la ambigüedad, comunicación y entendimiento.

Ambigüedad: Se puede definir como “malentendidos durante las entrevistas provocando especificaciones poco claras e incompletas producto del uso del lenguaje natural. Desde el lado

del cliente se manifiesta en el sentido de que este último se expresa utilizando un lenguaje lleno de jergas asociadas a su dominio provocando la interpretación múltiple de los analistas o entrega muy poca o demasiada información que se aleja de los límites de procesamiento del analista durante una entrevista (Donoso & Vega, 2017).

Comunicación: La comunicación es sin duda el elemento intrínseco del proceso de educación, sin un adecuado entendimiento de las partes existe un alto índice de rechazo del producto final dados los vacíos y malentendidos durante el proceso de análisis. Los principales factores que afectan la comunicación se producen desde el lado del cliente y abarcan distintos ámbitos:

- Comunicación infrecuente producto de las barreras organizacionales
- Falta de habilidades de presentación, habilidades de escritura, falta en la organización de ideas y organización general de la información.
- Respuesta tardía con base en el canal de comunicación utilizado aumentando los tiempos que tomará el desarrollo del producto.

Entendimiento: El entendimiento mutuo está estrechamente ligado con la ambigüedad, conocimiento tácito y comunicación que producen generalmente malentendidos. Esto se debe a que la ingeniería de requerimientos es un proceso en donde participan individuos de distintas disciplinas abarcando un amplio conjunto de conceptos propios a cada una de ellas. El foco central del entendimiento es la negociación la cual es fundamental para definir los límites del sistema, establecer prioridades en los requerimientos, identificar aquellas personas relevantes con la

autoridad y poder en la toma de decisiones, y finalmente discutir los cambios que se producen sobre los requerimientos. El especialista en requerimientos debe poseer un alto grado de comunicación interpersonal que le permita negociar en situaciones críticas y, a su vez, tratar con las diversas personalidades de los stakeholders enfocado en la resolución de conflictos.

Ámbito cognitivo, emocional y motivacional

De acuerdo con Donoso y Vega, la “dimensión social es un área muy poco explorada dentro de la Ingeniería de Software, esto se debe a que los ingenieros se enfocan más bien en aspectos técnicos por sobre los sociales” (Donoso & Vega, 2017). Los autores definen los tres componentes:

- **Cognitivo:** los individuos hacen uso de sus facultades mentales para producir la interacción analistas y stakeholders, lo que permite percibir la información de su entorno, interpretarla, representarla para el uso correcto en la toma de decisiones.
- **Emocional:** permite reconocer las «emociones» de los stakeholders derivadas de lo que supone representará el sistema para su entorno o trabajo. El especialista en requerimientos debe elegir cuidadosamente el método y forma en que expone sus inquietudes con todos los involucrados.
- **Motivacional:** es fundamental para predecir, detectar y evitar emociones reacias durante el proceso de educación; el ingeniero debe actuar como un pseudo-sicólogo promoviendo la participación del proceso en un ambiente agradable, de confianza, franqueza y empatía.

Ecosistemas Software

Un ecosistema de software “se forma a partir del sistema de relaciones entre entidades diferentes con el objetivo de compartir segmentos del mercado o aumentar la eficiencia y la eficacia en los procesos productivos que desarrollan. Está soportado por la definición de arquitecturas y plataformas comunes que facilitan la integración de soluciones y componentes, el intercambio de información, recursos, artefactos y activos en general. Estas relaciones deben estar soportadas por relaciones económico-financieras, convenios de trabajo o modelos de desarrollo que promuevan el intercambio libre de datos y códigos” (Piñero & Pérez, 2015).

De acuerdo con Recena, los ecosistemas de software son “el conjunto de soluciones de software que habilitan, automatizan y soportan cierta necesidad de negocio” (Recena, Muñiz, Gómez, & Beas, 2012). Otros autores han determinado de forma general que los ecosistemas de software “están orientados a crear alianzas entre diferentes organizaciones, enfoque a la externalización de las relaciones de las organizaciones, potenciando la productividad y compartir sectores de mercado” (Recena, Muñiz, Gómez, & Beas, 2012). Así también, Sosa menciona que “la base para el desarrollo de los ecosistemas de software está en el establecimiento de sólidas arquitecturas empresariales que se apoyen en la integración de sus componentes” (Sosa, Pérez, García, Peñaherrera, & Piñero, 2016).

De acuerdo con Sosa citando a Piñero (2105) un ecosistema de software “se forma a partir del sistema de relaciones entre entidades diferentes con el objetivo de compartir segmentos del mercado o aumentar la eficiencia y la eficacia en los procesos productivos que desarrollan” (Sosa,

Pérez, García, Peñaherrera, & Piñero, 2016). De acuerdo con los mencionados autores, los componentes del ecosistema “está soportado por la definición de arquitecturas y plataformas comunes que facilitan la integración de soluciones y componentes, el intercambio de información, recursos, artefactos y activos en general”. Las relaciones mencionadas deben “estar soportadas por relaciones económico-financieras, convenios de trabajo o modelos de desarrollo que promuevan el intercambio libre de datos y códigos” (Sosa, Pérez, García, Peñaherrera, & Piñero, 2016).

Así también, García menciona que “el término ecosistema tecnológico se utiliza para definir un conjunto de componentes software que se relacionan entre sí mediante flujos de información en un medio físico que proporciona el soporte para dichos flujos” (García, 2013). Haciendo una analogía con los ecosistemas naturales, define tres componentes que los ecosistemas de software poseen:

1. Cada componente se relaciona con otros componentes. Si un componente es totalmente independiente, no forma parte del ecosistema.
2. La estabilidad de los ecosistemas depende de su diversidad. Un ecosistema tecnológico es más estable cuantas más posibilidades y opciones ofrezca, pero manteniendo una armonía, unidad, seguridad y coherencia entre sí.
3. El ecosistema tecnológico debe crecer de manera controlada. La evolución del sistema sin un fin concreto, una necesidad, puede llevar a tener un ecosistema insostenible que no cumple con la finalidad para la que se desarrolló.

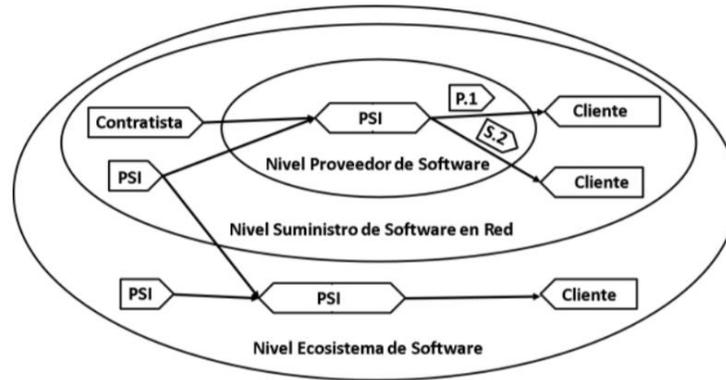
Resalta el autor que “una de las características fundamentales de un ecosistema tecnológico es la capacidad del sistema para soportar los cambios y adaptarse a las necesidades de los usuarios a lo largo del tiempo. Para que esto ocurra los componentes software que forman el ecosistema deben ser capaces de adaptarse a los cambios” (García, 2013).

Para determinar los orígenes del término (Castaño, 2016) menciona:

La primera definición de ECOS fue hecha Messerschmitt y Szypersky en el 2005 y lo definieron como una “colección de productos de software que presentan un determinado grado de relaciones simbióticas”. Frantz entiende por un ecosistema de software el conjunto de entidades de negocios que actúan como una unidad e interactúan para compartir un mercado común de aplicaciones informáticas y servicios bajo un sistema ordenado de interrelaciones entre ellas.

Así también, citando a Jensen y colaboradores, un ECOS es “un conjunto de negocios que funcionan como una unidad e interactúan con un mercado común para el software y los servicios, junto con las relaciones entre ellos” (Castaño, 2016). Además, Bosch lo define como “un sistema donde un conjunto de soluciones de software habilita, soportan y automatizan las actividades y transacciones hechas por los actores en el ecosistema, así como las organizaciones que proveen estas soluciones” (Bosch & Bosch-Sijtsema, 2010). El Gráfico 2 muestra las perspectivas o niveles de un ecosistema de software.

Gráfico 2 Ecosistemas de software



Nota: Perspectivas o niveles de un ecosistema de software.

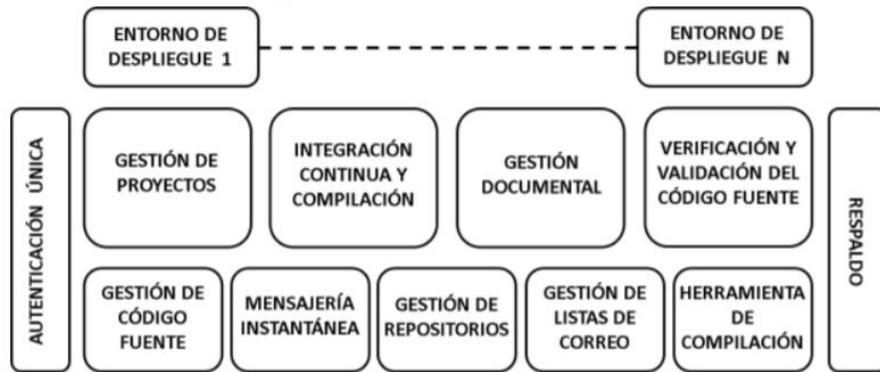
Fuente: A research agenda for software ecosystems.

Autor: Jansen, S.; Finkelstein, A., et al.

De los ecosistemas de software surge un concepto y problemática adicional para los desarrolladores de software por cuanto no se debe pensar en sistemas individuales o independientes que cumplen sus funciones de acuerdo con las necesidades identificadas, sino que se debe considerar la interacción que va a tener con los demás componentes del ecosistema. De esta manera se presentan los “Ecosistemas de Desarrollo de Software (ECODS)” los que son definidos como “un espacio de trabajo en el que conviven una serie de herramientas que, acompañadas de unas buenas prácticas, permiten a un equipo de desarrollo modelar una metodología de trabajo” (Recena S. , 2012).

En el gráfico 3 se representa el modelo conceptual para el desarrollo de ecosistemas de software presentado por (Recena S. , 2012).

Gráfico 3 Modelo Conceptual para el desarrollo de ecosistemas de software.



Nota: Gráfico sobre el desarrollo de ecosistemas de software.

Fuente: Ecosistemas de desarrollo software. Líneas de automatización.

Autor: Recena, Soto.

De acuerdo con Bosh, citado por (Parejo, Fernández, & Ruíz-Cortés, 2010) “un ecosistema de desarrollo de software se constituye a través de una plataforma software, un grupo de desarrolladores pertenecientes al responsable de dicha plataforma, un grupo de desarrolladores externos y una comunidad de expertos del dominio que componen soluciones relevantes para satisfacer las necesidades de una comunidad de usuarios”. La plataforma base del ecosistema podría ser un sistema operativo o una aplicación extensible. Una característica fundamental de un ecosistema software es el uso de un modelo de componentes para habilitar la distribución de desarrollos sobre la plataforma como por ejemplo AppStore o AndroidMarket.

Investigación de campo

La investigación de campo se refiere al proceso sistemático de recopilación de información en el lugar donde se presentan los eventos objeto del estudio. En la presente investigación se realizó el análisis de la información contenida en la comunidad en línea “Stack Overflow en español” con la

finalidad de identificar las características de las publicaciones, componentes del sitio, estructura de las preguntas y respuestas, y demás parámetros considerados para el desarrollo del modelo USER-BOT.

Investigación bibliográfica

Comprende en la recopilación, selección, análisis y síntesis de la información técnica relacionada al tema de investigación. Se realizó la búsqueda bibliográfica en libros, revistas indexadas especializadas en tecnología, y demás activos de información que permitió a los autores desarrollar una base teórica sólida para sustentar el proyecto. La ejecución de esta etapa comprende en el fundamento teórico-práctico sobre el cual se desarrolla el proyecto ya que se toma en consideración los resultados de investigaciones previas, documentos académicos y textos publicados con los más altos estándares científicos.

Comunidades en línea

J.C.R. Licklider y Robert Taylor en 1978 reflexionaban en la posibilidad de establecer comunidades en ambientes informáticos. Decían que “en la mayoría de los campos estarán constituidas de miembros separados geográficamente, a veces agrupados en pequeñas sociedades y a veces trabajando individualmente. Serán comunidades no basadas en una locación común, pero en intereses comunes” (Siles, 2005).

Así mismo, Siles citando a Sandy Stone (1991) propone otro concepto sugerente de comunidades informáticas al definir las como “incontrovertiblemente espacios sociales en los cuales las personas se encuentran todavía cara a cara, pero bajo nuevas definiciones tanto de "encontrarse" como de

"cara" [...] [son] puntos de paso para colecciones de creencias y prácticas en común que unían a personas que estaban separadas físicamente (Siles, 2005).

Siles, citando el popular libro “The Virtual Community” de Howard Rheingold, menciona que este determinó por primera vez el término de “comunidad virtual” definiéndolo a grupos que (Siles, 2005):

Reúnen individuos instalados en los cuatro rincones de la tierra pero que, en su gran mayoría, guardan a pesar de toda una inserción local. Desarrollan conversaciones igual de ricas intelectual y emocionalmente que las de la vida real. Es un mundo de intercambio entre iguales. En definitiva, la Red puede permitir reorganizar el vínculo social en debilidad, redinamizar el debate público y, más ampliamente, la vida democrática. Este libro propone entonces uno de los mitos fundadores de Internet

En consecuencia, se puede definir una comunidad en línea (también denominada comunidad virtual por el uso del internet para comunicarse) como “el conjunto de individuos organizados alrededor de intereses y afinidades en común, que interactúan en línea a partir de intercambios de texto y de las diferentes formas de comunicación digital e hipertextual (imágenes, vídeos, sonidos y animaciones, todos ligados entre sí) (Siles, 2005).

Campos, Laferrière y Harrasim (2001), citados por Siles, han categorizado las comunidades en línea en tres grupos (Siles, 2005):

- **Comunidades de interés**, en las cuales los individuos comparten intereses, discusiones y necesidades en común, y son grupos de intercambio generalmente más informales;
- **Comunidades de aprendizaje**, constituidas con fines educativos y según distintos niveles de enseñanza;

- **Comunidades de práctica**, en el sentido enunciado por Etienne Wenger (1999), es decir, espacios donde el aprendizaje social y la identidad de los individuos se construyen gracias a prácticas de participación e intercambios de significados. Son generalmente asociadas a espacios profesionales u organizacionales.

En la actualidad existen una infinidad de comunidades en línea que se ajustan a la categorización presentada previamente, en forma de redes sociales, foros de discusiones, comunidades de preguntas y respuestas. Estas últimas han tenido gran avance por cuanto buscan promover el conocimiento de los usuarios.

Las comunidades de preguntas y respuestas están orientadas a que los miembros planteen preguntas, en el ámbito del tema que trata la comunidad, y los demás participantes puedan responder aquella pregunta, en base a sus conocimientos y experiencias. Lo anterior permite la búsqueda de una solución concreta a un problema determinado. En las comunidades orientadas al desarrollo de software, en este tipo de comunidades es común encontrar preguntas y respuestas que incluyan segmentos de código para contextualizar la pregunta realizada. Devada, sitio web especializado en tecnología de la información y, específicamente de comunidades virtuales menciona que “las mejores comunidades de desarrolladores fomentan la creación y conservación de bases de conocimiento bajo demanda a través de la gamificación, incluidas insignias, logros de estado de perfil y tablas de clasificación”(Devada, 2018).

Stack Overflow

Stack Overflow es uno de los sitios web de StackExchange, que es una red compuesta por varios

sitios web con preguntas y respuestas gratuitas. Fue creado por Jeff Atwood utilizado por una comunidad de usuarios entre los cuales se destacan desarrolladores de software, en la cual otros miembros de la comunidad pueden encontrar soluciones a problemas de programación en diferentes lenguajes, entre otros temas concernientes al proceso de desarrollo de software.

De manera general, para hacer uso de la información de la comunidad Stack Overflow debe:

1. Suscribe al sitio web.
2. Publica su pregunta.
3. Recibe las respuestas.

Las respuestas son publicadas por los miembros de la comunidad con las mismas experiencias que encontraron solución al problema planteado o que tienen conocimiento del tema. Adicionalmente, todos los usuarios pueden votar por las preguntas y por sus respuestas; Cuando se vota por una pregunta, el usuario puede calificarlas como más relevante o menos relevante; por otra parte, cuando se vota por las respuestas, éstas pueden ser más acertadas o menos acertadas.

Reputación

Stack Overflow posee un aspecto interesante, la reputación, este término es destacado de acuerdo con la cantidad de votos que poseen las preguntas y las respuestas, a mayor cantidad de votos relevantes o aciertos, mayor es la reputación en el sitio web. De hecho, el número de votos es un indicador para muchos aspectos, entre estos tenemos:

- Confianza de los usuarios.
- Habilidades de comunicación.
- Calidad y relevancia en preguntas y respuestas.

- Manejo de los temas.
- Nivel de experiencia y participación.
- Base de datos

Stack overflow en español es una de las comunidades equivalentes de Stack Overflow, por ende, también pertenece a la red de comunidades Q&A llamada “Stack Exchange”. SO-ES está inspirada en SO, y la diferencia radica en que está dirigida a usuarios de países hispanos, en cuanto a la interfaz, sistema de reputación, funcionalidades, etc. son iguales a los de su homólogo en inglés. De acuerdo con datos publicados por “Stack Exchange” en SO-ES existen alrededor de 276138 usuarios registrados dentro de la comunidad.

Programación orientada a objetos

De acuerdo con Flores, la programación orientada a objetos (POO), se define como “un paradigma que permite realizar una abstracción de la realidad, que se puede implementar en una aplicación de software, que permite el desarrollo de aplicaciones robustas” (Flórez, 2012). La programación orientada a objetos (POO), “está constituida por objetos con ciertas características y métodos, los cuales pertenecen a una clase para poder satisfacer necesidades específicas que brinden funcionalidad al programa” (Flórez, 2012).

Web services

(Gil, 2015) menciona que un “web service es un sistema de comunicación entre diferentes servidores, a través de la red, basado en mensajes que cumplen un estándar (SOAP) basado en XML”, así también se menciona que es “es un sistema de software diseñado para soportar

interoperabilidad máquina a máquina a través de una red de interacción. Tiene una interfaz descrita en un formato procesable-máquina (específicamente WSDL)”. Ampliando los conceptos anteriores, también menciona que “es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios Web para intercambiar datos en redes de ordenadores como Internet”. Lo anterior ofrece a los usuarios compartir información de manera adecuada y fácilmente.

(Gil, 2015) adicionalmente resalta algunas ventajas de los “web services”, las cuales son:

- Aportan interoperabilidad entre aplicaciones de software, independientemente de sus propiedades o de las plataformas sobre las que se instalen.
- Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido.
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.
- Independencia del lenguaje de programación: El servidor y el cliente no necesitan estar escritos en el mismo lenguaje.
- Independencia del modo de transporte: SOAP puede funcionar sobre múltiples protocolos de transporte, como, por ejemplo: http, HTTPS, BEEP, IIOP, SMTP o FTP.

(Mitra, 2016) menciona que la interfaz de usuario puede definirse como todo lo que ve y con lo cual interactúa. En Android todos los elementos gráficos son construidos utilizando objetos “vistas” y “grupos de vistas”. Las vistas son objetos que se dibujan en la pantalla, mientras que el grupo de vistas es un contenedor invisible que contiene objetos de vista para definir un diseño particular. Así también, (Azzola, 2014) menciona además que “las aplicaciones de Android son muy diferentes entre sí porque intentan abordar las diferentes necesidades de los usuarios. Hay aplicaciones simples con una interfaz de usuario muy sencilla que solo tiene una vista y hay otras aplicaciones mucho más complejas con una navegación muy estructurada y múltiples vistas”.

Una interfaz estructurada de manera correcta que permita a los usuarios una navegación de tal manera que los usuarios tengan una experiencia adecuada es el primer paso en el camino para que el sistema sea aceptado y utilizado frecuentemente.

Base de datos

Las definiciones de base de datos son en la actualidad muy ampliamente aceptadas y, en su mayoría, refieren que es un conjunto de datos interrelacionados, almacenados en un dispositivo de almacenamiento, sin redundancias innecesarias, de forma independiente de los programas que acceden a ellos y que garantizan la integridad y seguridad necesarios. Así mismo, un sistema de gestión de base de datos (SGDB) consiste en la colección de datos almacenados y el conjunto de programas que acceden a ellos. Los sistemas de bases de datos son desarrollados para gestionar una gran cantidad de información garantizando el acceso adecuado a cada uno de los usuarios del sistema proveyendo los mecanismos necesarios para la manipulación de los datos almacenados.

Existe una gran cantidad de sistemas de gestión de base de datos en el mercado que las podemos clasificar en dos grandes grupos: los sistemas de gestión de base de datos de uso libre (Open Source) y aquellos por los cuales se paga un valor por el uso del sistema. Sin embargo, todos los mencionados sistemas ofrecen herramientas acordes que permiten gestionar la información eficientemente. Algunos sistemas más conocidos en la actualidad son:

1. Microsoft SQLServer
2. MySql
3. PostgreSql
4. Oracle DataBase
5. SQLite
6. MongoDB
7. Cassandra

Sistemas de gestión de base de datos no relacionales

Una base de datos no relacional es aquella que no requiere de estructuras de datos fijas como tablas, pero que sin embargo no garantiza las características ACID (acrónimo de Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad) y que escala muy bien horizontalmente. Este tipo de base de datos se utiliza en entornos distribuidos que deben estar siempre disponibles y operativos y que gestionan un gran volumen de datos. Entre los sistemas de gestión de base de datos no relacionales podemos encontrar a MongoDB, Redis,

Cassandra, entre otros de menos uso (Báez, y otros, 2016).

MongoDB

MongoDB es un sistema de gestión de base de datos multiplataforma orientado a documentos, de esquema libre. El esquema libre significa que cada entrada o registro puede tener un esquema de datos diferente, con atributos o “columnas” que no tienen por qué repetirse de un registro a otro. Está escrito en C++, lo que les confiere cierta cercanía a los recursos de hardware de la máquina, de modo que es bastante rápido a la hora de ejecutar sus tareas. Además, está licenciado como GNU AGPL 3.0, de modo que se trata de un software de licencia libre. Funciona en sistemas operativos Windows, Linux, OS X y Solaris (Paramio, 2020).

De acuerdo con la página oficial de MongoDB, este sistema de gestión de base de datos corresponde a “una base de datos documental, lo que significa que almacena datos en forma de documentos tipo JSON”. Según sus creadores, esta es la forma más natural de concebir los datos; frente al tradicional modelo de filas y columnas, esta es mucho más expresiva y potente (MongoDB, 2020). MongoDB es una base de datos NoSQL distribuida, basada en documentos y de uso general que ha sido diseñada para desarrolladores de aplicaciones modernas y para la era de la nube” (MongoDB, 2020).

Adicionalmente, (Paramio, 2020) al respecto del modo de almacenamiento de datos, menciona:

En MongoDB, cada registro o conjunto de datos se denomina **documento**. Los documentos se pueden agrupar en **colecciones**, las cuales se podría decir que son el equivalente a las tablas en una base de datos relacional (sólo que las colecciones pueden almacenar documentos con muy diferentes formatos, en lugar de estar sometidos a un esquema fijo). Se pueden crear índices para

algunos atributos de los documentos, de modo que MongoDB mantendrá una estructura interna eficiente para el acceso a la información por los contenidos de estos atributos

Algunas de las ventajas de MongoDB mencionadas por (Paramio, 2020) son:

- Ofrece mejores resultados al momento de manejar grandes cantidades de documentos o bases de datos en modo lectura.
- El coste de implementación es muy bajo en comparación con otras soluciones relacionales y su rendimiento podría ser bastante mayor.
- Permite agregar y editar cualquier información nueva a los documentos por separado.
- Realiza muchas operaciones por segundo reduciendo los tiempos para el usuario.
- Puede ser escalable y permite, mediante el sistema de sharding, distribuir información por diferentes clústeres de máquinas.

Así también, el autor resalta algunas desventajas, que son:

- No es capaz de realizar transacciones.
- Carece de Joins para realizar consultas complejas.

Consultas en MongoDB

Las bases de datos relacionales tienen un lenguaje propio para realizar consultas denominado “Structured Query Language” o Lenguaje de Consulta Estructurado, más conocido como SQL. Pero en MongoDB no existe la posibilidad de utilizar SQL. En este caso, y al igual que con los otros tipos de consultas, las búsquedas se realizarán con Javascript. Algunas de las funciones más utilizadas para consultar datos en MongoDB son:

Find: Permite buscar documentos en una colección. Si no se le pasa ningún parámetro, se buscarán

todos los elementos en la colección. Aunque se podría pensar que esta consulta devolvería todos los resultados, esto no es así. Para evitar devolver demasiados documentos y sobrecargar el servidor, MongoDB utiliza los cursores.

Filtros: Para añadir un filtro a la función `find` se debe pasar un documento JSON como parámetro de la consulta.

Cursores: es una conexión con el servidor que permanece abierta, y que permite iterar sobre los resultados de una consulta, de forma que estos se devuelvan poco a poco.

Proyecciones: Una proyección se utiliza para devolver un conjunto determinado de campos de un documento.

findOne: Nos devuelve un solo documento.

findAndModify: Permite buscar un documento para modificar alguno de sus campos y luego devolverlo para procesarlo en la aplicación.

MySQL

MySQL es un sistema gestor de base de datos relacional que posee la licencia de uso libre por lo cual dicha herramienta puede ser descargada del sitio web y utilizada en los proyectos de software que lo requieran. Es uno de los sistemas gestores de base de datos cuyo uso se ha popularizado especialmente en aplicaciones de internet por su facilidad de uso, gran cantidad de documentación y un extenso número de desarrolladores que publican sus productos a través de la red de redes. Para el uso de este sistema no existen costos por licencia relacionados al uso ni mantenimiento de esta. Garantiza la seguridad, integridad y confidencialidad de los datos almacenados, además posee mecanismos de escalabilidad a las organizaciones que deciden utilizarla.

MySQL posee una licencia GNU (General Public Licence) que es una licencia de derecho de autor que garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software. Su propósito es doble: declarar que el software cubierto por esta licencia es libre, y protegerlo (mediante una práctica conocida como copyleft) de intentos de apropiación que restrinjan esas libertades a nuevos usuarios cada vez que la obra es distribuida, modificada o ampliada (Mysql, 2014).

En el sitio web de la empresa propietaria menciona algunas características de MySQL, entre las cuales se puede resaltar:

- Velocidad y robustez.
- Multiproceso.
- Multiplataforma.
- Sistema de contraseñas y privilegios muy flexible y seguro.
- Encriptación.
- Los clientes usan TCP o UNIX Socket para conectarse al servidor.
- Soporte de mensajes de error en distintos idiomas.
- Documentación técnica completa.

Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de base de datos relacional, desarrollado por la empresa Microsoft. El lenguaje de desarrollo utilizado es Transact-SQL, una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos, crear tablas y definir relaciones entre ellas. Es uno de los sistemas más utilizados a nivel mundial por sus características

de calidad y seguridad que posee, además del soporte que brinda la empresa dueña del producto.

Python

Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 cuyo nombre está inspirado en el grupo de cómicos ingleses “Monty Python”. Es un lenguaje similar a Perl, pero con una sintaxis muy limpia y que favorece un código legible. Se trata de un lenguaje interpretado o de script, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos (González R. , 2016). La ventaja de los lenguajes compilados es que son más rápidos en su ejecución, sin embargo, los lenguajes interpretados son más flexibles y portables. Python posee muchas de las características de los lenguajes compilados lo cual lo hace algunos autores lo definen como semi interpretado.

El tipado dinámico se refiere a que no es necesario declarar inicialmente los tipos de datos de las variables que van a ser utilizados en el programa sino que dicho tipo se determinará al momento de ejecución, de acuerdo al tipo de valor que se le asigne y puede cambiar durante la ejecución del programa, al momento de que se asigne un valor diferente (González R. , 2016). La característica de fuertemente tipado se refiere a que “no se permite tratar a una variable como si fuera de un tipo distinto al que tiene, es necesario convertir de forma explícita dicha variable al nuevo tipo previamente” (pág. 8).

El intérprete de Python está disponible en una amplia gama de plataformas computacionales como Unix, Linux, Windows, Solaris, Mac Os, entre otras lo que lo hace popular por cuanto, si no se utiliza librerías específicas de cada plataforma, el programa desarrollado en este lenguaje se podría ejecutar en todas las plataformas mencionados sin cambios significativos (González R. , 2016).

IDE para Python

Para ejecutar Código Python se puede utilizar el intérprete en el cual se puede escribir líneas de código y obtener una respuesta del intérprete por cada línea escrita en modo de sesión interactiva; o bien escribir el código del programa en un archivo de texto y ejecutarlo posteriormente. Los IDE (Entorno de desarrollo integrado) son herramientas de software que le permiten al programador acceder a una serie de funciones que ayudan al desarrollo de aplicaciones de manera más sencilla. Existen algunos IDE's populares actualmente, entre los cuales (G., 2016) destaca:

Pycharm IDE: IDE desarrollado por la empresa “JetBrains” y es una herramienta profesional que viene en dos modalidades: una edición Free y otra muy completa privada orientada a empresas de desarrollo de software

PyDev IDE: Es un IDE para Python libre de costo y se puede encontrar una larga lista de poderosas características para programar de manera eficiente y, básicamente, se trata de un plugin de código abierto que se ejecuta en Eclipse. Algunas de las características más importantes son “Integración con Django, completa el código de manera automática, soporte multi lenguaje, plantillas de código, análisis de código, marcado de errores y mucho más”

Sublime Text 3 IDE: es uno de los más livianos y potentes IDE's en la actualidad. Con el uso de Plugins, Sublime Text 3 puede utilizarse como un IDE completo. La interfaz de usuario es muy rápida y fácil de configurar. Contiene muchos paquetes para darle diferentes características. Anaconda es un complemento que convierte Sublime Text 3 en un excelente IDE Python.

Wing IDE: Es una herramienta comercial y está orientada para desarrolladores profesionales. Fue lanzado hace casi 20 años y es un producto muy maduro, con una infinidad de herramientas y

características para programar en Python. Es soportado por Windows, OS X y Linux.

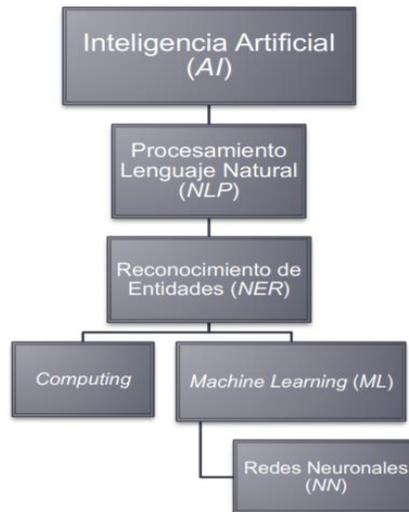
Vim IDE: Es uno de los editores más avanzados y populares dentro de la comunidad de desarrolladores Python. Es de código abierto y se encuentra disponible gratuitamente bajo licencia GPL.

Inteligencia artificial

De acuerdo con Leyva-Vázquez y Smarandache, existen dos enfoques hacia la inteligencia artificial. El denominado “inteligencia artificial general aspira a crear la capacidad de resolver tareas generales en las máquinas, tales como pensar y actuar semejante a la mente humana (Leyva & Smarandache, 2018). El otro, denominado “inteligencia artificial estrecha o IA débil pretende crear la capacidad en las máquinas de realizar tareas específicas, extremadamente bien, incluso superando a los humanos. Este enfoque está teniendo incontables éxitos y aplicaciones prácticas” (Leyva & Smarandache, 2018). Además, “la inteligencia artificial utiliza modelos inteligentes para predecir, clasificar y agrupar datos de manera que se obtenga información relevante para la toma de decisiones y para desarrollar flujos de trabajo rápidos, óptimos y prácticos para las personas” (Barrientos & Mamani, 2019).

Así también, Leyva y Smarandache mencionan que “la inteligencia artificial no es un campo monolítico, sino que está dividido en varias ramas tales como: el aprendizaje automático, el procesamiento de lenguaje natural, los sistemas expertos, la visión por computadoras, el reconocimiento automático del habla, la planificación y la robótica” (Leyva & Smarandache, 2018). En el Gráfico 4 se presenta una clasificación general de la Inteligencia artificial definido por (Gallardo, 2019).

Gráfico 4 Clasificación de la Inteligencia artificial.



Nota: Representación de la clasificación general de la Inteligencia artificial.

Elaboración: Gallardo, Eduardo.

Un elemento importante en la Inteligencia Artificial es el aprendizaje automático. El aprendizaje automático es una rama de la Inteligencia Artificial que tiene como objetivo lograr que las computadoras aprendan. Existen 5 paradigmas fundamentales del aprendizaje automático:

- Algoritmos evolutivos,
- Conexionismo y redes neuronales,
- Simbolismo,
- Redes bayesianas,
- Razonamiento por analogía

El aprendizaje automático tiene como propósito dotar a las computadoras de la capacidad de aprender sin haber sido explícitamente programadas para ello. Sin embargo, presenta limitaciones en cuanto a la necesidad de realizar la selección de variables a procesar y en cómo escalan en

presencia de grandes volúmenes de datos tales como imágenes y videos.

Procesamiento del lenguaje natural

El procesamiento del lenguaje natural (NLP, por sus siglas en inglés) consiste en desarrollar aplicaciones y servicios que puedan comprender los idiomas humanos. Ejemplos prácticos de procesamiento del lenguaje natural (NLP) son el reconocimiento de voz, la traducción del habla, la comprensión de oraciones completas, la comprensión de sinónimos de palabras coincidentes y la escritura de oraciones y párrafos completos gramaticalmente correctos.

Implementaciones del NLP

Estas son algunas de las implementaciones exitosas del procesamiento de lenguaje natural (NLP):

- **Motores de búsqueda** como Google, Yahoo, etc. El motor de búsqueda de Google entiende que eres un técnico, por lo que te muestra resultados relacionados a ti.
- **Los feeds de los sitios web sociales** como la fuente de noticias de Facebook. El algoritmo de feed de noticias entiende tus intereses mediante el procesamiento de lenguaje natural y muestra anuncios y publicaciones relacionadas con mayor probabilidad que otras publicaciones.
- **Motores de voz** como Apple Siri.
- **Filtros de spam** como filtros de spam de Google. No se trata solo del **filtro de spam** habitual, ahora los filtros de spam entienden qué hay dentro del contenido del correo electrónico y ver si es un correo no deseado o no.

Librerías NLP

Hay muchas librerías de NLP de código abierto y éstas son algunas de ellas:

- Natural language toolkit (NLTK).
- Apache OpenNLP.
- Stanford NLP suite.
- Gate NLP library.

Natural Language Toolkit (NLTK) es la biblioteca más popular para el procesamiento del lenguaje natural (NLP) que fue escrita en Python y tiene una gran comunidad detrás. NLTK también es muy fácil de aprender; en realidad, es la biblioteca de procesamiento de lenguaje natural (NLP) más fácil que usará.

Machine learning

De acuerdo con Russo, entre otros, “machine learning es un área de la inteligencia artificial que engloba un conjunto de técnicas que hacen posible el aprendizaje automático a través del entrenamiento con grandes volúmenes de datos” (Russo, y otros, 2018). En la actualidad existen modelos que utilizan esta técnica logrando excelentes resultados. Además, “La construcción de modelos de Machine Learning requiere adaptaciones propias debido a la naturaleza de los datos o a la problemática a la que se aplica” (Russo, y otros, 2018).

De acuerdo con Russo, entre otros, se pueden encontrar tres grandes grupos de algoritmos “Machine Learning”, que son:

1. **Algoritmos supervisados:** Estos algoritmos utilizan un “conjunto de datos de entrenamiento etiquetados (preclasificados), los cuales procesan para realizar predicciones

sobre los mismos, corrigiéndolas cuando son incorrectas. El proceso de entrenamiento continúa hasta que el modelo alcanza un nivel deseado de precisión” (Russo, y otros, 2018).

2. **Algoritmos semi-supervisados:** Estos algoritmos combinan tanto datos etiquetados como no etiquetados para generar una función deseada o clasificador. Este tipo de modelos deben aprender las estructuras para organizar los datos, así como también realizar predicciones (Russo, y otros, 2018).
3. **Algoritmos no supervisados:** El conjunto de datos no se encuentra etiquetado y no se tiene un resultado conocido. Por ello deben deducir las estructuras presentes en los datos de entrada, lo puede conseguir a través de un proceso matemático para reducir la redundancia sistemáticamente u organizando los datos por similitud (Russo, y otros, 2018).

En el Gráfico 5 se aprecia la clasificación de machine learning.

Gráfico 5 Clasificación de machine learning.



Nota: Clasificación de machine learning.

Fuente: Inteligencia artificial

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Clasificador Naive Bayes

Corresponde a los algoritmos de machine learning supervisados por clasificación. El clasificador de Naive Bayes es utilizado para predecir eventos futuros mediante operaciones probabilísticas, este método evalúa la probabilidad de cada una de las clases, así como sus variables independientes. El algoritmo NB está compuesta de un nodo padre y n nodos hijos, donde el nodo principal representa a la variable de clase (variable dependiente) y sus hijos hacen referencia a las variables independientes. Pereira y otros, citando a Witten, mencionan que “El objetivo de la clasificación en el aprendizaje automático es entrenar un método determinado a partir de un conjunto de datos para construir un modelo que sea capaz de predecir uno de los valores nominales que pertenecen al dominio de un atributo llamado clase” (Pereira-Toledo, López-Cabrera, & Quintero-Domínguez, 2017).

Por otra parte, Salamanca y Velásquez indican que Naive Bayes es un “clasificador estadístico supervisado. Está basado en el teorema de bayes en la teoría de hipótesis de independencia entre las variables X_1, \dots, X_p .”(Salamanca & Velásquez, 2018).

Revisiones sistemáticas

Mapeo sistemático

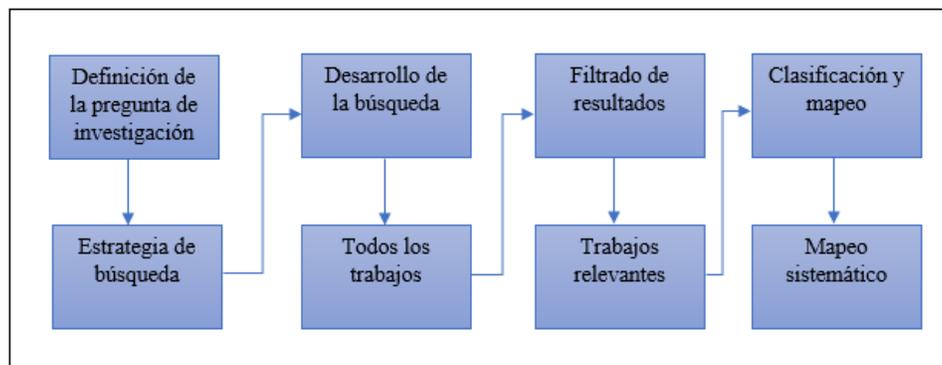
De acuerdo con Celdrán, Mazón, Ivars y Vera el mapeo sistemático “es un método que se centra en la clasificación de la producción científica asociada a un determinado tema, para lo que categoriza y contabiliza los resultados o estudios primarios relacionados.” (Celdrán, Mazón, Ivars, & Vera, 2018). El resultado de la información se combina para dar respuesta a la pregunta de investigación, para ello se realiza una estrategia o protocolo fundamentado. En el Gráfico 6 se muestra el proceso sistemático de revisión de la producción científica compuesto de una serie de etapas. El mapeo sistemático es una técnica de utilidad para temas de investigación muy amplios. De acuerdo a (Botto, 2014), las principales diferencias entre un mapeo sistemático y una revisión son:

- Los mapeos son generalmente manejados por preguntas de investigación extensas y a menudo responden múltiples preguntas de investigación.
- Los términos de búsqueda para mapeos sistemáticos suelen ser enfocados a un nivel más alto que las revisiones sistemáticas y retornan un gran número de estudios. Para un mapeo, sin embargo, es menos problemático trabajar con abundantes resultados.
- El proceso de extracción de datos para los mapeos sistemáticos es además mucho más amplio que la extracción de datos para revisiones sistemáticas y puede ser definido como un estado de clasificación o categorización.

- El propósito de este estado es clasificar los estudios con suficiente detalle para responder las preguntas de investigación más amplias y para identificar estudios para las subsecuentes revisiones.
- El estado de análisis para un estudio de mapeo sistemático consiste en resumir los datos de acuerdo a responder la pregunta de investigación. No es necesario incluir técnicas de análisis a profundidad tales como el meta-análisis y la síntesis narrativa, sino totales y resúmenes. Representaciones gráficas de distribución de estudios por tipos de clasificación pueden ser un tipo efectivo de mecanismo de presentación de resultados.
- La diseminación de los resultados de un mapeo puede ser más limitada que de una revisión sistemática.

El punto de partida del proceso de mapeo sistemáticos es la definición de la pregunta de investigación, que guían la estrategia de búsqueda que permitirá identificar los trabajos más relevantes.

Gráfico 6 Etapas Mapeo Sistemático



Nota: Datos propios de la investigación.
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El mapeo sistemático del presente estudio se encuentra estructurado en cuatro fases:

1. **Búsqueda (Search):** Durante la fase de búsqueda se formuló la pregunta de investigación y el modelo de búsqueda para delimitar la misma a estudios que guarden relación con el presente trabajo de investigación.
2. **Evaluación:** Se definen criterios de inclusión como de exclusión para delimitar la información considerada en el estudio con la finalidad de identificar los estudios relevantes que se ajusten a la presente investigación, a la vez que permite reducir la cantidad de estudios encontrados como resultado de la fase de búsqueda.
3. **Síntesis:** Se realiza a través de un método cuantitativo, es decir, se le asignara un valor numérico al criterio.
4. **Análisis:** en esta fase se realizará la interpretación de los resultados obtenido en la fase de síntesis y con ello se determinará los estudios relevantes.

1. Búsqueda

Pregunta de investigación

El área de investigación corresponde a machine learning y específicamente al análisis y ejecución de algoritmos de aprendizaje automático aplicados a datos de tipo textual que contiene Stack Overflow en español.

La pregunta de investigación que centró la realización del presente trabajo de titulación fue:

RQ1: ¿Qué modelos de aprendizaje automático han utilizado los investigadores para entrenamiento de los datos en comunidades en línea?

Búsqueda, identificación y selección de estudios

Para la identificación de estudios relacionados a la pregunta de investigación se ha considerado como fuente de búsqueda la plataforma “Google Scholar” o Google académico que indexa una gran cantidad de documentos científicos de una inmensa variedad de artículos publicados en revistas científicas, congresos y bases de datos científicas.

Cadena de búsqueda

Para la búsqueda de artículos científicos necesarios para el análisis, luego de probar diversas combinaciones de palabras y analizar los diferentes resultados obtenidos en la consulta respectiva, se registró la siguiente cadena de consulta:

(modelo* or técnica* or método* o algoritmo*) and ((machine learning) or (aprendizaje automático)) and bot * and texto* and (comunidad and línea)

2. Evaluación

Criterios de inclusión: los criterios de inclusión corresponden a las características que deben cumplir los artículos consultados para ser considerados como parte del proceso de análisis. Para que un artículo sea considerado como parte del estudio debe:

1. Estudios que presenten técnicas, modelos o algoritmos de aprendizaje automático orientados al tratamiento de texto.
2. Trabajos de investigación que presenten resultados prácticos o teóricos sobre técnicas de modelos de aprendizaje automático aplicados a conjunto de texto.
3. Artículos publicados en el año 2015 o posterior.

Criterios de exclusión: los criterios de exclusión son aquellos parámetros que, al cumplirse uno de ellos, causa que no sea considerado dentro del presente trabajo. Para excluir los artículos consultados, basta que se cumpla una de las siguientes condiciones:

- Al artículo corresponde a temas introductorios y libros, capítulos de libros.
- Investigaciones repetidas que provengan de diferentes fuentes.
- Artículos con una cantidad de páginas inferior a cinco.
- El artículo no contiene en su título las palabras clave de la cadena de búsqueda.
- El artículo no menciona en su resumen las palabras clave determinadas en la cadena de búsqueda.
- No es de acceso gratuito.
- No es un artículo de revista indexada.
- Artículo publicado antes del año 2015

Aseguramiento de la calidad

Para garantizar la calidad de los artículos seleccionados, se deberá realizar las siguientes interrogantes:

- ¿El artículo presenta técnicas, modelos o algoritmos de aprendizaje automático?
- ¿El artículo se encuentra publicado en una revista científica, congreso o conferencia académica?
- ¿Existen citas de otros autores que hagan referencia al artículo seleccionado?

Las interrogantes planteadas permiten identificar los estudios relevantes que se encuentran acorde con el tema de investigación propuesto.

3. Síntesis

Extracción de datos

Para la extracción de datos se ha procedido a determinar varias sub-preguntas de investigación que se consideraron como criterios lo cual permite una mejor búsqueda y organización de la información. El siguiente cuadro presenta los criterios de cada sub-pregunta.

Tabla 3 Sub-preguntas y criterios.

Pregunta de Investigación	Criterio	Opciones
Q1: ¿Cómo se clasifican los algoritmos de aprendizaje automático?	C1: Supervisado	Clasificación Regresión
	C2: No supervisado	Clústering Asociación
	C3: Por refuerzo	Reducción de dimensionalidad Proceso de decisión Sistemas de premio
Q2: ¿Cuáles son los tipos de soporte informático utilizado en la investigación?	C4: Tipos de soporte	Automatizado
		Semi-Automatizado
		Manual
Q3: ¿Qué tipos de estructuras de datos han utilizado?	C5: Artefactos usados	<i>Dataset</i>
		Archivos de texto plano

Nota: Datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

Los criterios que trata la sub-pregunta 1 (Q1: ¿Cómo se clasifican los algoritmos de aprendizaje automático?), se definen como:

C1: Supervisado: Se utilizó este criterio para clasificar aquellos estudios que utilizan técnicas de aprendizaje automático de tipo supervisados utilizando un conjunto de datos previamente etiquetados para realizar predicciones. El posible valor de este criterio es:

- a) **Clasificación:** cuando el estudio se refiere a un problema en el cual los datos de entrada deben clasificarse en determinadas categorías.
- b) **Regresión:** cuando el estudio se refiere a un problema en el cuál la salida es un número real, como por ejemplo un peso o un precio.

C2: No supervisado: Cuando el estudio se refiere a modelos que no conocen las categorías de los datos, es decir, el modelo aprende únicamente en base a los datos de entrada, sin ninguna supervisión.

- a) **Clustering:** cuando el estudio se refiere a problemas que pretenden determinar agrupaciones de los datos de entrada.
- b) **Asociación:** cuando el estudio trate acerca problemas que tratan de encontrar similitudes o asociaciones entre los valores de los datos.
- c) **Reducción de la dimensionalidad:** el objetivo de este tipo de algoritmo es convertir un *dataset* de una dimensión extensa a una de menor dimensión.

C3: Por refuerzo: Si el artículo se basa en algoritmos por refuerzo en cuyo problema se tiene un agente que interactúa con un entorno, de forma que recibe refuerzos positivos o negativos de cada interacción y aprende cuál decisión es la correcta.

- a) **Proceso de decisión:** forma idealizada matemáticamente del problema de aprendizaje por refuerzo, para el cual se podría encontrar un enunciado teórico preciso que pueda describirla.

b) Sistema de premio: son procesos con valores que juzgan que tan positivo es estar en un estado.

Los criterios para la sub-pregunta 2 (Q2: ¿Cuáles son los tipos de soporte informático utilizado en la investigación?) se definen como:

C4: Tipos de soporte: se refiere al tipo de automatización de los procesos que presenta en la investigación. Se puede clasificar en:

- c) **Automatizado:** Si el investigador utilizó herramientas de software o programas de aplicación para la obtención y tratamiento de los datos.
- d) **Semi-automatizado:** Si el investigador utilizó para ciertas etapas de la obtención y tratamiento de datos herramientas y/o programas computacionales.
- e) **Manual:** Si la obtención y tratamiento de datos se realizó sin apoyo de herramientas informáticas.

Con respecto a la tercera sub-pregunta Q3: ¿Qué tipos de estructuras de datos han utilizado? se refiere a qué tipo de origen de datos ha utilizado el investigador.

C5: Artefactos. Se refiere al tipo de elementos o estructuras se utilizó para la recopilación de datos. Los valores pueden ser:

- a) **Dataset:** Conjunto de datos organizados dentro de una estructura lógica establecida como una tabla de base de datos.

b) **Archivo plano:** Documentos con datos textuales sin un formato o estructura determinada.

Método de Síntesis

El método de síntesis utilizado es cuantitativo y cualitativo. Para el método cuantitativo se consideró los siguientes parámetros:

1. Cantidad total de estudios clasificados en cada uno de los criterios planteados en la Tabla 3.
2. Cantidad total de estudios agrupados por año.

Para el análisis cualitativo se realizó la inclusión de acuerdo con los criterios determinados en los párrafos anteriores del aseguramiento de calidad y aplicados por los autores de la presente investigación.

Conducción

El primer paso consistió en adaptar la cadena de búsqueda a la sintaxis definida por el motor de búsqueda de Google Académico. Luego, se aplicaron los filtros por año y área y se ordenaron de acuerdo con el nivel de ocurrencia de las palabras claves tanto en el título, como en el resumen.

Posteriormente, en la cadena de consulta arroja aproximadamente 400 artículos para el análisis. Una vez aplicados los criterios de inclusión, se logró obtener un total de 21 artículos elegibles para someterlos al proceso de análisis de datos, tal como se muestra en la Tabla 4.

Tabla 4 Selección de artículos.

Fuente	Estudios potenciales	Porcentaje	Estudios seleccionados	Porcentaje
Búsqueda automática				
Google Scholar	400	100%	21	5,25%
Total	400	100%	21	5,25%

Nota: Datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

Resultados

Los resultados del mapeo sistemático se encuentran en la Tabla 5 donde se puede visualizar la cantidad de estudios obtenidos por cada sub-pregunta de investigación. En el anexo “Tablas y referencias” se encuentra la tabla detallada.

Para el criterio C1 (Supervisado), el mapeo sistemático arroja que el 71.4% de los autores implementaron sus investigaciones con algoritmos supervisados para el tratamiento de texto lo cual nos indica una tendencia es la preferida para este tipo de datos. (Priego, Anaya, & Reyes, 2018) menciona que los algoritmos supervisados dieron un mejor resultado para el análisis de texto acerca de la clasificación de noticias de desastres naturales. Así también, (Katiyar, Kumar, & Walia, 2020) menciona que Las técnicas de aprendizaje automático supervisado son capaces de aprender el proceso humano para clasificar con base en las características alimentadas, y los parámetros que se les pueden asignar.

Para el criterio C2 (No supervisado) se encontró que 9.5% de los estudios revisados mencionan algoritmos no supervisados como una opción para el tratamiento de texto.

Para el criterio C3 (De refuerzo) se encontró que el 19% de los estudios analizados utilizaban dicho algoritmo, en muchas ocasiones combinado con otros. Así, (Talavera & Luna, 2020) menciona que una metodología de aprendizaje automático muy conveniente es el aprendizaje por refuerzo o Reinforcement Learning (RL), donde un agente aprende el comportamiento a través de interacciones de prueba y error en un entorno dinámico.

Acerca del criterio C4 (Tipos de soporte), el 81% de los trabajos revisados utilizaron un soporte automático lo que representa que utilizaron aplicaciones para ejecutar cada una de las fases del estudio. Un pequeño porcentaje del 4.8% utiliza mecanismos manuales de procesamiento para obtener los resultados planteados en sus investigaciones.

El criterio C5 (Artefactos) demuestra que el 71% de los artículos investigados utilizan para la obtención de datos el mecanismo de “Dataset” y el restante 28.6% realiza manipulación de archivos de texto plano.

Tabla 5 Resultados del mapeo sistemático

Criterio	Valor	Resultados	
		N° de estudios	Porcentaje %
C1: Supervisado	Supervisado	15	71.4%
C2: No supervisado	No supervisado	2	9.5%
C3: Por refuerzo	Por refuerzo	4	19%
C4: Tipo de soporte	Automatizado	17	81%
	Semi automatizado	3	14.3%
	Manual	1	4.8%

C5: Artefactos	Dataset	15	71.4%
	Archivo de texto	6	28.6%

Nota: Datos propios de la investigación.
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

Hipótesis

¿El modelo USER-BOT mediante técnicas de algoritmos de aprendizaje automático responde preguntas en SO-ES correctamente?

Variables de la investigación

Independiente: Algoritmos de aprendizaje automático.

Se define los algoritmos de recuperación, procesamiento, análisis y presentación de datos que provee las técnicas de machine learning.

Dependiente: Responder preguntas de Stack Overflow a través de un USER-BOT correctamente.

Corrección: Si la respuesta emitida corresponde al tema planteado.

Tiempo: Si las respuestas emitidas fueron entregadas en el tiempo adecuado

Utilidad: Si las respuestas presentadas son de utilidad para el usuario.

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

Modalidad de la investigación

Durante el desarrollo de la investigación se realizaron las actividades técnicas concernientes a la elaboración del modelo de machine learning que implementa el USER-BOT en el sistema de mensajería instantánea “Telegram”. El desarrollo del modelo de machine learning comprende una serie de actividades técnicas íntimamente relacionadas que da como resultado un sistema que interactúa con el usuario y presenta información de Stack Overflow en español. El desarrollo del *bot* se realizó siguiendo una metodología de desarrollo de software incremental donde el equipo de trabajo va completando la funcionalidad del sistema en partes o incrementos hasta culminar con un sistema completamente funcional. Cada incremento de la aplicación puede tener las siguientes actividades:

- Obtención de requerimientos
- Análisis y Diseño
- Desarrollo
- Verificación y validación

Al finalizar esta etapa se presenta el sistema completamente funcional, acorde con las expectativas de los usuarios y, en este caso, acorde a las características que los investigadores definieron al inicio del estudio.

Una vez desarrollado el modelo machine learning a través del USER-BOT que presenta los resultados en la aplicación de mensajería instantánea “Telegram”, se realizó una fase experimental en la cual realizaron actividades orientadas que usuarios reales puedan interactuar con la comunidad en línea “Stack Overflow en español” a través de USER-BOT con la finalidad de evaluar los resultados obtenidos lo que permitió mejorar el modelo de búsqueda y los algoritmos utilizados para obtener respuestas acertadas en el menor tiempo posible. Para realizar la validación de los resultados obtenidos por el USER-BOT, se ha solicitado la colaboración de profesionales de la ciudad de Guayaquil que laboran en el área de desarrollo de software en diferentes instituciones públicas y privadas para que realicen una serie de pruebas que permitan determinar la validez de la propuesta lo que se denomina la técnica de juicio de expertos.

Tipo de investigación

Investigación cuasi experimental

El presente trabajo de titulación fue desarrollado con un tipo de investigación cuasi experimental la cual, según Manterola y Otzen (2015) se caracteriza “especialmente por la ausencia de asignación aleatoria. El objetivo de este manuscrito es reportar los principios básicos, tipos, características y estrategias para la conducción de ECE.” (pág. 1).

Es una deducción de los objetos experimentales, en los cuales la asignación del objeto de investigación no es aleatoria, aunque el factor de exposición es controlado por el investigador, se contrasta hipótesis causales para lograr los objetivos.

Diseño metodológico de la investigación

La investigación se ha realizado en etapas claramente definidas que ha tenido por objeto desarrollar los componentes del modelo de aprendizaje automático a través de un USER-BOT.

A continuación, se presentan los componentes desarrollados para concluir con éxito el proyecto de titulación.

Preparación del ambiente

Las herramientas que se utilizaron para el presente estudio son:

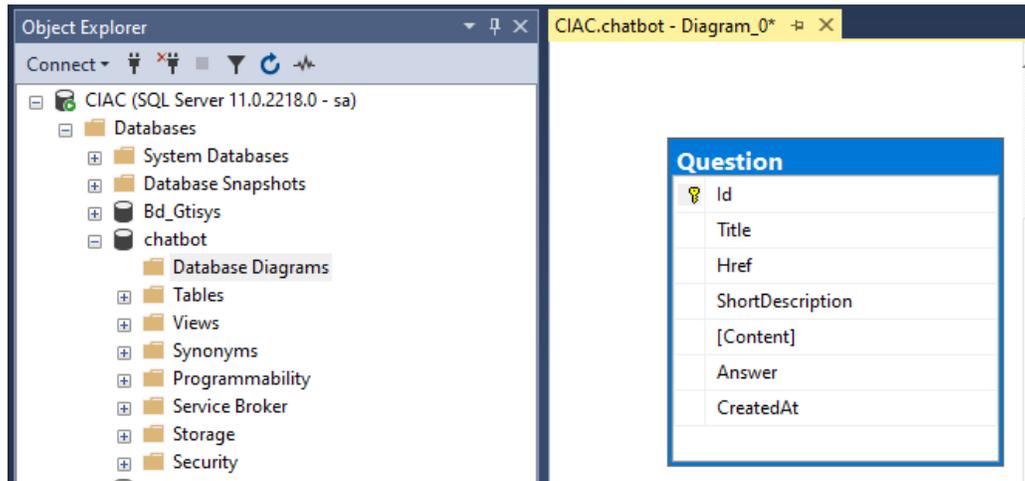
1. Estructura de datos.
2. Python 3.8.
3. Telegram.

Creación de estructuras de datos

Se realizó el diseño de la base de datos tomando como referencia la data de la red de comunidades Stack Exchange. Tomando como etiqueta la data que maneja el gestor de base MySQLi. Como resultado a la recolección de datos obtenida se procedió a crear la base de datos llamada “CHATBOT” y con una tabla llamada “QUESTION”.

En el Gráfico 7 se muestra la ilustración de la base de datos y la tabla creada.

Gráfico 7 Base de datos y la tabla creada.



Nota: Ilustración de la base de datos y la tabla creada.

Fuente: Investigación

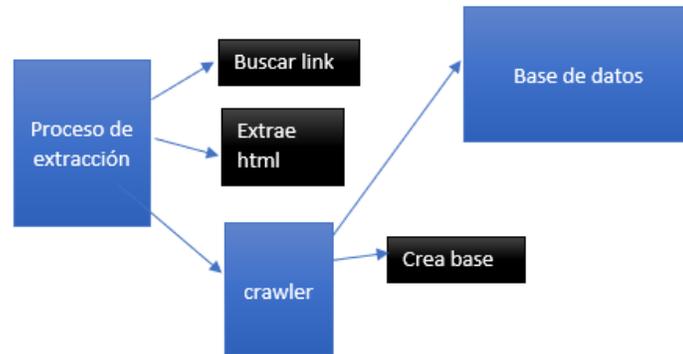
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Extracción de datos

Es el proceso a través del cual el USER-BOT recopila la información de preguntas y respuestas que contiene la comunidad en línea Stack Overflow en español, realiza un proceso de depuración y clasificación de acuerdo con los algoritmos implementados y realiza el almacenamiento en la base de datos anteriormente definida.

A continuación, en el Gráfico 8 se presenta el proceso de extracción con el USER-BOT.

Gráfico 8 Proceso de extracción con USER-BOT.



Nota: Representación del Proceso de extracción con USER-BOT.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

Para el proceso de extracción:

- Se busca el enlace de la página, el cual es: <https://es.stackoverflow.com>.
- Se procedió a extraer la información en html con el siguiente filtro:
/questions/tagged/mysqli?page={page}&sort=votes&pagesize=50.

En el *crawler* realiza un proceso a través del cual se crea la base de datos automáticamente en el caso de al momento de ejecutarse el *bot*, en caso de que no exista previamente.

Por último, se almacena la información obtenida de la extracción en la base de datos.

Preprocesamiento de información

Para este proceso se tomó en consideración el análisis de contenido ya que, al trabajar con un conjunto de datos, está la posibilidad de encontrarnos con datos irrelevantes que perjudiquen el buen funcionamiento del USER-BOT.

Este proceso comienza su función al momento que el usuario envía su pregunta al USER-BOT.

A continuación, se indica los pasos de limpieza que se emplean:

1. Transformación de las letras mayúsculas a minúsculas.
2. Tokenización.
3. Eliminación de palabras irrelevantes.
4. Lematización.

El primer paso se utiliza para establecer una estandarización, el método que empleamos es “**lower()**” convierte todos los caracteres que estén en mayúsculas a minúsculas.

En el segundo paso se emplea “**tokenización**” para realizar la separación de cadenas de textos largas a cadenas de textos cortas.

La utilización del tercer paso ayuda a disminuir el tiempo de aprendizaje del USER-BOT dando como resultado que la respuesta dada por el *bot* sea más precisa. Para esto se utilizó la función “**stopwords**” que elimina del origen de datos las palabras “conectoras” que se repiten con frecuencia en un idioma.

Por último, se utilizó “**lematización**” que no ayuda a simplificar aún más las palabras enviadas por el usuario dando como resultado que el USER-BOT sea más preciso al momento de dar la respuesta.

Gráfico 9 Resultado.

```
message received ¿como hago una consulta en mysql?  
Pregunta dividida: ['como', 'hago', 'una', 'consulta', 'en', 'mysql']  
Palabras vacías: ['hago', 'consulta', 'mysql']  
Lematización: ['hag', 'consult', 'mysql']
```

Nota: Demostración del proceso de limpieza.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

En el Gráfico 9 se toma como ejemplo la pregunta: “¿Cómo hago una consulta en MySQLi?” dando como resultado lo mostrado en dicho gráfico.

Aprendizaje Automático

Durante esta fase se procedió a la construcción del modelo de aprendizaje automático usando el algoritmo de Bayes que nos ayuda con el aprendizaje automático que tiene como base una técnica de clasificación estadística llamada “**Teorema de bayes**” usando como ayuda “**Pandas**” librería destinada al análisis de datos, ayuda a proporcionar una estructura de dato flexible que permite realizar un trabajo más eficiente, trabaja en conjunto con “**Numpy**” librería que ayuda a un potente tipo matricial a Python.

Por ser un algoritmo

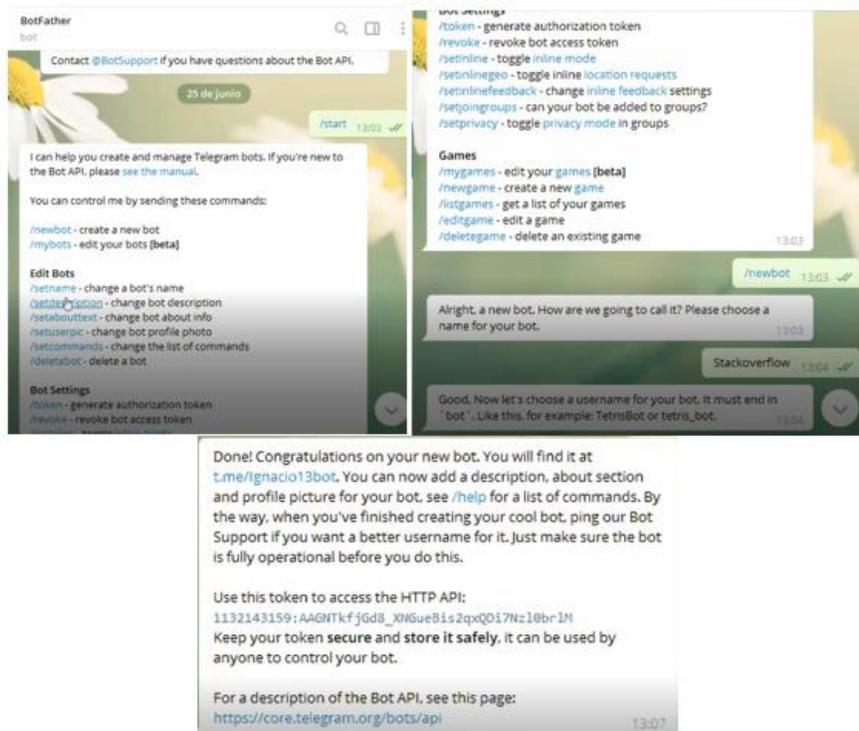
Se utilizo el teorema de Bayes por cuanto proporciona una manera diferente de calcular la probabilidad de la hipótesis estando basada en su probabilidad previa, las probabilidades de observar diversos datos dada la hipótesis y los datos observados en sí.

Registro y configuración del bot

Para la configuración del *bot* se utilizó el *bot* “*BotFather*” que se encuentra disponible en el API de Telegram que, como su nombre lo indica es el “papá” de todos los *bot* y se utiliza para poder configurar cualquier *bot* que se quiera implementar.

En el Gráfico 10 se visualiza el proceso para el registro del *bot* dentro de “*BotFather*” donde, una vez registrado, genera un código de acceso o “token” para utilizarlo en el código fuente del Bot con la finalidad de que Telegram “reconozca” el *bot* y pueda ser utilizado por los usuarios finales.

Gráfico 10 Registro del bot.



Nota: Proceso para el registro del bot dentro de “BotFather”.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

En el Gráfico 11 se muestra el código donde crea una instancia y se realiza la conexión entre el USER-BOT con Telegram.

Gráfico 11 Conexión a Telegram.

```
def train_bot(file):  
    logger.debug('creating bot')  
    telegram_token = '1132143159:AAGNTkfjGd8_XNGueBis2qxQDi7Nz10br1M'  
    updater = Updater(  
        token = telegram_token,  
        use_context = True  
    )
```

Nota: Conexión entre el USER-BOT con Telegram.

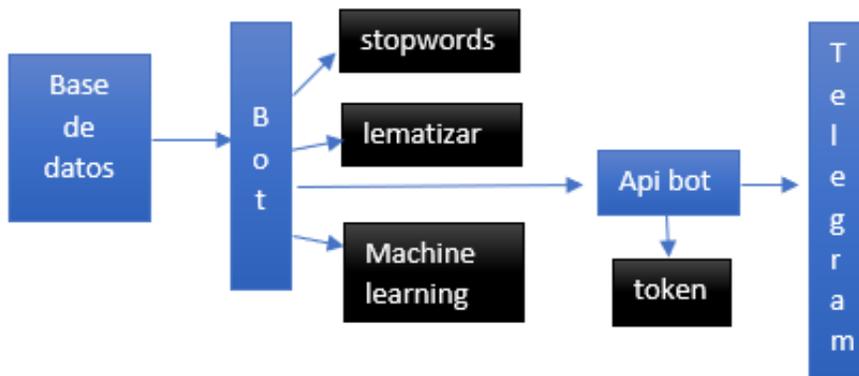
Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Consulta

A continuación, en el Gráfico 12 se muestra el proceso de consulta.

Gráfico 12 Proceso de consulta con USER-BOT.



Nota: Proceso de consulta con USER-BOT.

Fuente: Investigación

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

La Base de datos es el repositorio donde se almacena la información que el *bot* obtuvo de la comunidad en línea Stack Overflow en español mientras tanto el *bot* es el conjunto de programas que realizan toda la lógica de recopilación, análisis, ordenamiento, almacenamiento de la información y la interacción con el usuario.

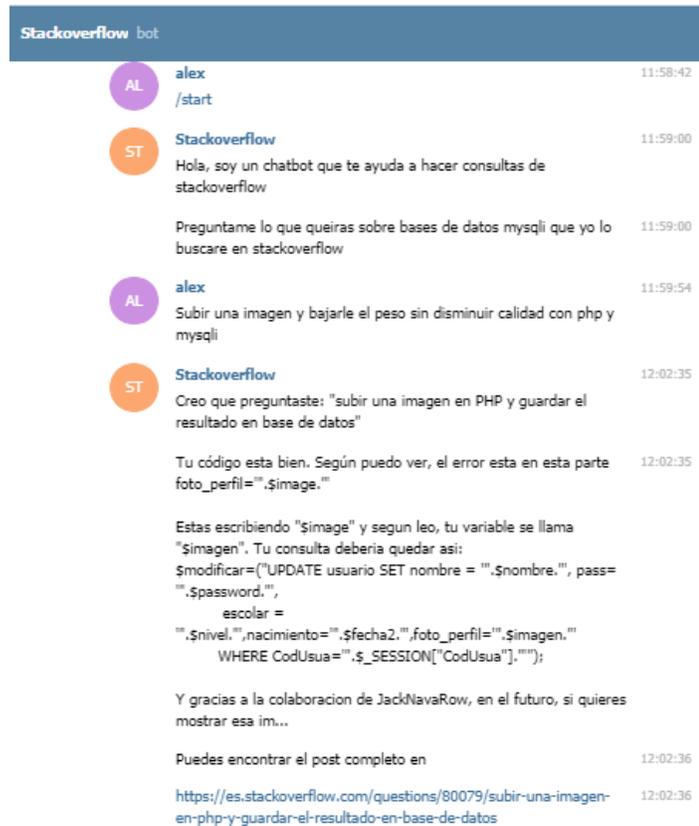
Está conformado por una serie de procesos lógicos como son el “limpiado” de datos de las palabras que ingrese el usuario denominado “stopwords”, el proceso de “lematización” y el proceso de machine learning que obtiene las respuestas adecuadas para poder responder la consulta del usuario.

El api *bot* es el método y librería que pone a disposición Telegram para el uso de todos los desarrolladores que permiten a los usuarios interactuar con el programa y el *bot* en Telegram que es la aplicación de mensajería instantánea para dispositivos móviles que se utilizará como interfaz de usuario para el USER-BOT.

Interacción con el Bot

Una vez que se ha desarrollado las etapas previas, el *bot* se encuentra en etapa de funcionamiento y puede ser utilizado a través de la aplicación de mensajería “Telegram”. Para utilizar el *bot* se debe buscar por su nombre en la interfaz de “búsqueda de contactos”. En el gráfico 13 se presenta la interfaz del usuario con el *bot*.

Gráfico 13 Interfaz User-Bot.



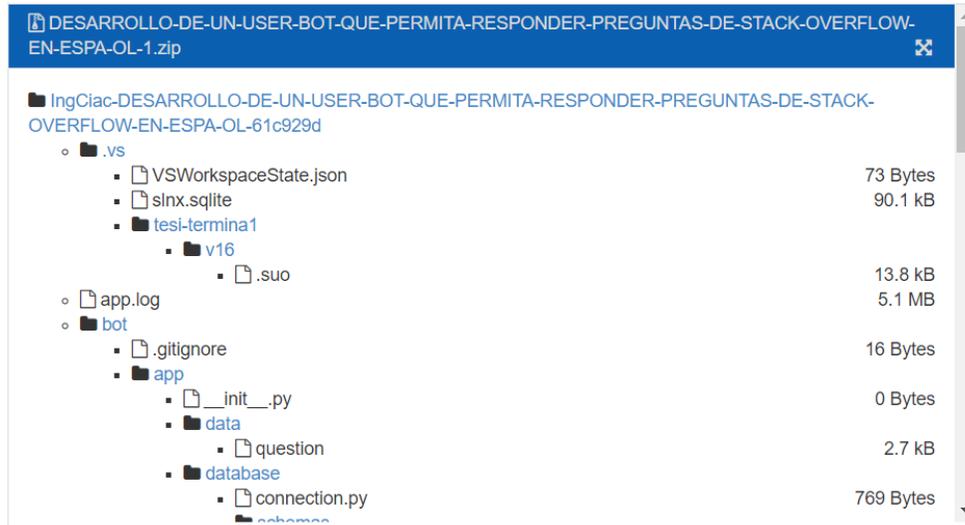
Nota: Interfaz que visualiza el usuario.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El código fuente del proyecto presentado se puede encontrar en el repositorio GitHub de zenodo (Almache & Sánchez, 2020) que tiene la siguiente estructura y el cual se muestra en el Gráfico 14:

Gráfico 14 Estructura del programa en zenodo



Nota: Interfaz.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El enlace para acceder al repositorio es el siguiente:

<https://zenodo.org/badge/latestdoi/302775912>

Procesamiento y análisis

Para verificar que la aplicación funcione de manera correcta se utilizó la técnica de juicio de expertos poniendo a consideración de cinco profesionales que trabajan en actividades relacionadas al desarrollo de software para validar la propuesta de investigación. A cada uno de los expertos entrevistados se les solicitó que utilicen el USER-BOT a través de su aplicación de mensajería instantánea “Telegram” y, además, se les presentó una serie de preguntas relacionadas al proyecto de investigación. Cada uno de los profesionales que participaron en el estudio llenaron

un formulario de recolección de datos con la finalidad de que sus opiniones formen parte del resultado de la investigación.

Beneficiarios directos e indirectos del proyecto

El proyecto de titulación está dirigido a satisfacer las necesidades de información a todas las personas que se encuentran interesadas en conocer a cerca del proceso de desarrollo de software y que consideran a la comunidad “Stack Overflow en español” como una fuente de información válida para el desarrollo de sus actividades profesionales. Adicionalmente, se beneficia a la comunidad académica, docentes y estudiantes de las instituciones de educación superior.

Criterios de validación de la propuesta

Para el proyecto de investigación se utilizó la técnica de Análisis de contingencia (Estadístico Chi-cuadrado) para el contraste de hipótesis.

Los criterios que se tomaran en cuenta son los siguientes:

Criterio de toma de decisión en el caso que se utilice el criterio de expertos:

- Si $p \leq 0.05$ entonces se rechaza H_0 y se acepta H_1 y se dice significativo (Si la probabilidad correspondiente al valor calculado por la prueba estadística es menor o igual que su respectivo valor crítico al nivel de 0.05, entonces se rechaza H_0 y se dice significativo).
- Si $p > 0.05$ entonces se acepta H_0 y se dice no significativo (Si la probabilidad correspondiente al valor calculado por la prueba estadística es mayor que su respectivo valor crítico al nivel de 0.05, entonces se acepta H_0 y se dice no significativo).

Donde:

p: valor de probabilidad

α : Nivel de significancia, con un valor de 0.05

H0: ¿El modelo realizado mediante aprendizaje automático responde preguntas de Stack Overflow en español de manera correcta?

H1: ¿El modelo realizado mediante aprendizaje automático responde preguntas de Stack Overflow en español de manera incorrecta?

Para calcular los resultados de juicio de expertos, los datos fueron ingresados a la “calculadora estadística” que nos ofrece el sitio web especializado “Social Science Statistics” que, en una de sus secciones tienen a disposición la calculadora Chi-Cuadrado (<https://www.socscistatistics.com/tests/>).

Tabla 6 de contingencia. Pregunta #5

	correcto	incorrecta	Totales de filas marginales
Con aprendizaje automático	4 (3,5) [0,07]	1 (1,5) [0,17]	5
Sin aprendizaje automático	3 (3,5) [0,07]	2 (1,5) [0,17]	5
Totales de columna marginal	7	3	10 (Gran Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El estadístico de chi-cuadrado es 0.4762.

El valor p es 0.490153.

P muestra un valor mayor a α ($0.49 > 0.05$), por lo tanto, el resultado no es significativo en $p < 0.05$ de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Tabla 7 de contingencia. Pregunta #6

	a tiempo	a destiempo	Marginal Row Totals
Con Machine learning	3 (2.5) [0.1]	2 (2.5) [0.1]	5
Sin machine learning	2 (2.5) [0.1]	3 (2.5) [0.1]	5
Marginal Column Totals	5	5	10 (Grand Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

La estadística de chi-cuadrado es 0.4.

El valor p es 0.527089.

P muestra un valor mayor a α ($0.4 > 0.05$), por lo tanto, el resultado no es significativo en $p < 0.05$. de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Tabla 8 de contingencia. Pregunta #7

	Útil	No útil	Marginal Row Totals
Con Machine learning	4 (3.5) [0.07]	1 (1.5) [0.17]	5
Sin machine learning	3 (3.5) [0.07]	2 (1.5) [0.17]	5
Marginal Column Totals	7	3	10 (Grand Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El estadístico de chi-cuadrado es 0.4762.

El valor p es 0.490153.

P muestra un valor mayor a α ($0.490153 > 0.05$), por lo tanto, el resultado no es significativo en $p < 0.05$ de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Tabla 9 de contingencia. Pregunta #8

	Mejora	No mejora	Marginal Row Totals
Con Machine learning	4 (3) [0.33]	1 (2) [0.5]	5
Sin machine learning	2 (3) [0.33]	3 (2) [0.5]	5
Marginal Column Totals	6	4	10 (Grand Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El estadístico de chi-cuadrado es 1.6667.

El valor p es 0.196706

P muestra un valor mayor a α ($0.196706 > 0.05$), por lo tanto, el resultado no es significativo en $p < 0.05$. de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Dataset

A continuación, en el Gráfico 15 se describen los *dataset* usados, en el cual se toma en cuenta las preguntas sin contestar de la comunidad en línea Stack Overflow en español que contengan la etiqueta MySQLi, tomadas de la página <https://es.stackoverflow.com/>

Gráfico 15 Dataset1

id_no	id_posible		similar
120671	26146		1
240887	224130		1
283513	104647		1
329622	143686		0
55103	43323		0
320035	224646		0

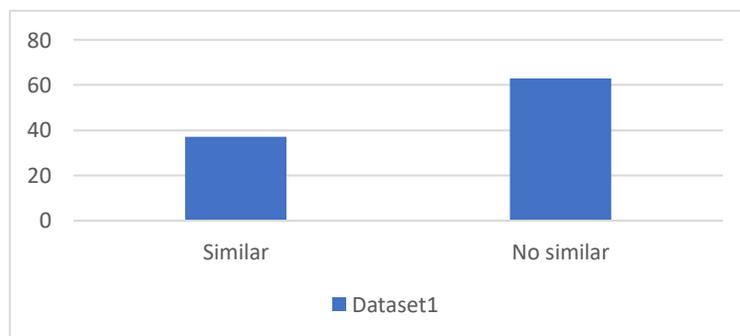
Nota: *Id_no* = Id de las preguntas;
Id_posible = Posible repuesta a la pregunta sin contestar;
Similar = Si *id_no* es similar a *id_posible*=1 caso contrario = 0.
Fuente: Investigación
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Durante la recolección y filtrado de preguntas se obtuvo un total de 234 preguntas sin contestar de las cuales se tomó en cuenta 186 preguntas.

Las 48 preguntas restantes no se las tomó en cuenta por uno de los siguientes motivos:

- Fueron eliminadas por no cumplir los estándares requeridos.
- No se encontraron resultados para las preguntas.

Gráfico 16 Valores Dataset1



Nota:
Fuente: Elaboración propia
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

De las 186 preguntas seleccionadas se procedió a realizar la validación del cual se obtuvo que el 37,1% de las preguntas tienen similitud y el 62,9% restante no como se muestra en el gráfico 16.

Gráfico 17 Dataset2

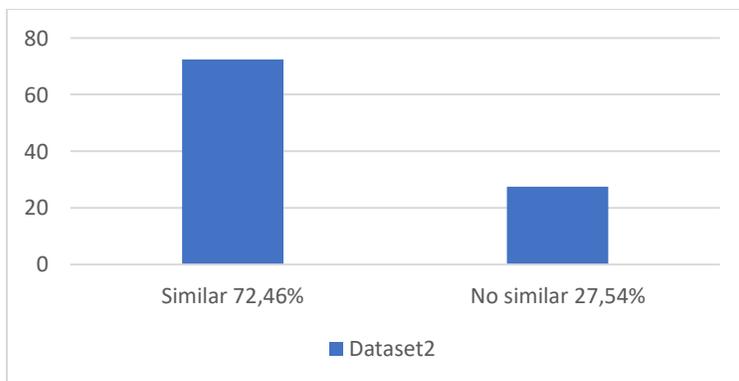
id_no	id_bot	Similitud
245356	195353	1
109806	87006	0
97756	258085	1
172458	271761	0

Nota: *Id_bot* = Id dado por el user-bot
 Similitud = Si *id_posible* es similar a *id_bot* = 1 caso contrario = 0.

Fuente: Elaboración propia

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Gráfico 18 Valores Dataset2



Nota:

Fuente: Elaboración propia

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Como se puede apreciar en el gráfico 18 refleja el resultado del análisis obtenido entre la comparación de las columnas *id_no* y *id_bot* campos que se pueden visualizar en el gráfico 17.

Para este análisis solo se tomó en cuenta las preguntas que dieron como resultado 1 en el *dataset1*.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Una vez que se ha culminado el trabajo de investigación se puede concluir:

- Se desarrolló un modelo USER-BOT que, mediante técnicas de aprendizaje automático, responde preguntas de la comunidad en línea Stack Overflow en español.
- Se desarrolló un repositorio de información para almacenar los resultados de la extracción de información obtenida de la comunidad en línea Stack Overflow en español. Esta base de datos interactúa con el USER-BOT para entregar las respuestas solicitadas al usuario.
- Se realizó la investigación bibliográfica para formar los fundamentos teóricos y prácticos necesarios al equipo de trabajo a través de la búsqueda, análisis, selección y registro de datos de fuentes bibliográficas.
- Se desarrolló un modelo USER-BOT que utiliza algoritmos de machine learning que permite responder preguntas que el usuario ingresa en la aplicación de mensajería instantánea “Telegram” con información obtenida de la comunidad en línea Stack Overflow en español.

- Los resultados obtenidos por la aplicación son positivos de acuerdo con la valoración acreditada por los profesionales que realizan sus actividades profesionales en el área de desarrollo de software.

Recomendaciones

Una vez que se ha concluido el desarrollo del modelo USER-BOT que implementan algoritmos de aprendizaje automático para responder preguntas que ingrese el usuario, los autores de la investigación recomiendan:

- Poblar la base de datos del USER-BOT con la mayor cantidad de información posible con la finalidad de que la aplicación pueda presentar resultados adecuados.
- Realizar el proceso de “limpiado” de la información obtenida de la comunidad en línea Stack Overflow en español, con la finalidad de eliminar caracteres o palabras que alteren los resultados búsqueda.
- Monitorear el proceso de “aprendizaje” del *bot* con la finalidad de que los resultados que se vayan presentando con el pasar del tiempo sean más efectivos.

Trabajos futuros

Una vez que se ha concluido el desarrollo de la presente investigación se puede determinar que los trabajos futuros se podrían enfocar en:

- Implementar el USER-BOT en otra aplicación de mensajería instantánea que ponga a disposición de la comunidad del software su api de desarrollo
- Implementar diferentes algoritmos de aprendizaje automatizado en el modelo de USER-BOT para evaluar las diferencias en cuanto a las respuestas obtenidas con cada una de ellas.
- Que la extracción de información se pueda realizar de diferentes comunidades virtuales de desarrollo de software que permita centralizar la información y que el USER-BOT pueda tener diversas fuentes.

REFERENCIAS BIBLIOGRÁFICAS

- Abreu, J. (2012). Hipótesis, Método y diseño de investigación. *International Journal of Good Conscience*, 187-197.
- Almache, C., & Sánchez, A. (01 de 10 de 2020). *zenodo*.
doi:<http://doi.org/10.5281/zenodo.4075675>
- Azzola, F. (2014). *Android UI Design*. Exelixis Media Ltd.
- Báez, M., Borrego, Á., Cordero, J., Cruz, L., González, M., Hernández, F., . . . Álvaro, Z. (2016). *Introducción a Android*. Madrid: Universidad Complutense de Madrid.
- Barrientos, E., & Mamani, S. (2019). *Modelos de aprendizaje supervisado como apoyo a la toma de decisiones en las organizaciones basados en datos de redes sociales: Una revisión sistemática de la literatura*. Lima.
- Bohórquez-Muñoz, J. (2018). *Algoritmo de ubicación de fallas de un terminal basado en modelos de machine learning*.
- Bosch, J., & Bosch-Sijtsema. (2010). From integration to composition: On the impact of software product lines, global development and ecosystems. *ScienceDirect*.
- Botto, M. (2014). SOA2Cloud: Un marco de trabajo para la migración de aplicaciones SOA a Cloud siguiendo una aproximación dirigida por modelos. 176.
- Calibar, A., J, H., & RO, K. (2018). Análisis de similitud en documentos de texto mediante técnicas de ciencia de datos basadas en aprendizaje profundo (deep learning).
- Carod, M., & Aranda, C. (2017). *Búsqueda de estrategias para la clasificación del contenido en foros técnicos de discusión*.

- Castaño, Y. (2016). Evaluación de ecosistemas de desarrollo de software desde la perspectiva de cobit. *IV Taller Internacional Las TIC en la Gestión de las Organizaciones*. La Habana. Obtenido de https://d1wqtxts1xzle7.cloudfront.net/57613192/EVALUACION_DE_ECOSISTEMAS_DE_DESARROLLO_DE_SOFTWARE_DESDE_LA_PERSPECTIVA_DE_COBIT.pdf?1540233793=&response-content-disposition=inline%3B+filename%3DEVALUACION_DE_ECOSISTEMAS_DE_DESARROLLO.pdf&Expires=16017445
- Celdrán, M., Mazón, J., Ivars, J., & Vera, J. (2018). SMART TOURISM. UN ESTUDIO DE MAPEO SISTEMÁTICO. 32.
- Donoso, Í., & Vega, V. (2017). Factores sociales y humanos que afectan el proceso de educación de requerimientos: una revisión sistemática. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, 69-83.
- Flórez, H. (2012). *Programación orientada a objetos usando Java*. Bogotá: ECOE Ediciones.
- G., D. (1 de 11 de 2016). *My Python*. Obtenido de <http://www.pythondiario.com/2016/11/los-5-mejores-ide-para-python.html>
- Gallardo, E. (2019). *Creación de un modelo predictivo para clasificar las consultas ciudadanas sobre el transporte público*. Valencia.
- García, A. (2013). *Análisis de integración de soluciones basadas en software como servicio para la implantación de ecosistemas tecnológicos corporativos*. Salamanca.
- Gil, C. (2015). *Los web services y características de calidad*.
- González, R. (2016). *Python para todos*.

- González, R., Pérez, I., García, R., Peñaherrera, E., & Piñero, P. (2016). *Ecosistema de Software GESPRO-16.05 para la Gestión de Proyectos*. La Habana: Scielo.
- Gramajo, M., Ballejos, L., & Ale, M. (2018). Análisis de las técnicas de aprendizaje automático aplicadas en la ingeniería de requerimientos de software: un mapeo sistemático de la literatura.
- Gras, V. (2020). Detección de palabras cortadas al inicio y fin de línea en textos manuscritos.
- Katiyar, S., Kumar, S., & Walia, H. (2020). Predicción de personalidad a partir de Stack Overflow mediante el uso del teorema de Naïve Bayes en minería de datos. 5.
- Leyva, M., & Smarandache, F. (2018). Inteligencia Artificial: retos, perspectivas y papel de la neutrosofía.
- M Danilo, T. L. (2019). Un enfoque de Machine Learning en el desarrollo de Sistema Recomendadores para Procesos de Investigación.
- Manterola, C., & Otzen, T. (2015). Estudios Experimentales 2 Parte. Estudios Cuasi-Experimentales. *International Journal of Morphology*.
- Martech. (07 de 01 de 2020). *Martech Forum*. Obtenido de <https://www.martechforum.com/articulo/las-mejores-comunidades-para-programadores-web/>
- Martínez, N., Zoratto, V., Aranda, G., Cechich, A., Chiarotto, A., Noda, C., & Sagripanti, M. (2018). *Análisis de calidad y recuperación de información en foros de discusión*. Buenos Aires.
- Mitra, G. (2016). *Android user interfaz*.
- MongoDB. (01 de 02 de 2020). *MongoDB*. Obtenido de <https://www.mongodb.com/es>
- Mysql. (2014). *MySQL 5.0 Reference Manual*.

- Paramio, C. (01 de 02 de 2020). *Genbeta*. Obtenido de <https://www.genbeta.com/desarrollo/una-introduccion-a-mongodb>
- Parejo, J., Fernández, P., & Ruíz-Cortés, A. (2010). *De frameworks a Ecosistemas: Evolución del software para optimización metaheurística*. Sevilla: Universidad de Sevilla.
- Pereira-Toledo, A., López-Cabrera, J., & Quintero-Domínguez, L. (2017). Estudio experimental para la comparación del desempeño de Naïve Bayes con otros clasificadores bayesianos. *Revista Cubana de Ciencias Informáticas*.
- Piñero, P., & Pérez, I. (2015). *Modelo de desarrollo de software basados en líneas de productos de software y ecosistemas de software. Reporte de Investigación. Laboratorio de investigación en Gestión de Proyectos*. La Habana: Universidad de las Ciencias Informáticas.
- Priego, B., Anaya, S., & Reyes, J. (2018). Análisis del comportamiento de diferentes. 13.
- Recena, M., Muñiz, M., Gómez, A., & Beas, J. (2012). *Software Development Ecosystem*. Sevilla, España.
- Recena, S. (2012). Ecosistemas de desarrollo software. Líneas de automatización. *Escuela Técnica Superior de Ingeniería Informática*, 20-30.
- Russo, C., Ramón, H., Alonso, N., Cicerchía, B., Esnaola, L., & Tessore, J. (2018). Tratamiento Masivo de Datos Utilizando Técnicas de Machine Learning. *Instituto de investigación y transferencia en tecnología*.
- Sandoval-Serrano, L. (2018). *Algoritmos de aprendizaje automático para análisis y predicción de datos*.
- Siles, I. (2005). *Comunidades en línea historia, comunicación y tecnología en la emergencia de colectivos mediáticos*. San José - Costa Rica: Revista de Ciencias Sociales .

- Sommerville, I. (2015). *Ingeniería del Software*. Madrid: PEARSON EDUCACIÓN. S.A.
- Sommerville, I. (2015). *Ingeniería del Software*. Madrid: PEARSON EDUCACIÓN. S.A.
- Sosa, R., Pérez, I., García, R., Peñaherrera, E., & Piñero, P. (2016). Ecosistema de Software GESPRO-16.05 para la Gestión de Proyectos. *Revista cubana de ciencias informáticas*.
- Stalling, W. (2005). *Sistemas operativos. Aspectos internos y principios de diseño*. Madrid: Pearson.
- Talavera, A., & Luna, A. (2020). Aprendizaje Automático: Una Contribución a la Investigación Operativa. 7.
- Velásquez, F., De-Paz, J., & Castillo, M. (2017). Aplicación del análisis sintáctico automático en la atribución de autoría de mensajes en redes sociales.
- Wolf, G., Ruiz, E., Bergero, F., & Meza, E. (2015). *Fundamentos del sistemas operativos*. Coyoacán.
- Zoratto, V., Aranda, G., & N, M. (2019). Análisis de estrategias para clasificación de usuarios y post dentro de un hilo de discusión.

ANEXOS

A continuación, se presenta una lista de los anexos que se incluyen en el proyecto:

- Anexo 1. Planificación de actividades del proyecto.
- Anexo 2. Fundamentación legal.
- Anexo 3. Criterios éticos a utilizarse en el desarrollo del proyecto.
- Anexo 4. Validación de expertos.
- Anexo 5. Estudios Seleccionados.
- Anexo 6. Acta de entrega y recepción definitiva.
- Anexo 7. Artículo científico.

Anexo 1. Planificación de actividades del proyecto.

Nombre de tarea	Duración	Duración días	Comienzo	Fin
<i>Proyecto Desarrollo USER BOT</i>	4,5 mss.	122	jue 25/06/20	mié 28/10/20
<i>Elaboración y entrega de la propuesta de trabajo</i>	10 días		jue 25/06/20	mié 08/07/20
<i>Fase de exploración e inicio de proyecto -Investigación -Desarrollo -Tutorías</i>	62 días		sáb 11/07/20	dom 04/10/20
<i>Fase de producto y estabilización -Ingreso y revisión de calificaciones</i>	16 días		lun 28/09/20	dom 18/10/20
<i>Sustentación</i>	13 días		lun 12/10/20	mié 28/10/20

Elaboración: Investigadores.

Fuente: Propia.

Anexo 2. Fundamentación Legal

Las Normas Legales en un Proyecto de Titulación

Apoyo en leyes, estatutos, acuerdos, reglamentos, especialmente para proyectos especiales y factibles, debe escribir únicamente los artículos citados en la CONSTITUCIÓN DE LA REPUBLICA DEL ECUADOR; LEY ORGÁNICA DE EDUCACIÓN SUPERIOR (art. 21), REGLAMENTO DEL CONSEJO DE EDUCACIÓN SUPERIOR; LEY ÓRGANICA DE TRANSPARENCIA Y ACCESO A LA INFORMACIÓN PÚBLICA; LEY DEL SISTEMA NACIONAL DE REGISTRO DE DATOS PÚBLICOS; CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, BUEN VIVIR, etc.

- Considerar solo artículos relacionados al tema.
- Debe iniciar la redacción con un breve antecedente de la base legal para realizar el proyecto de titulación.

Ejemplo “El presente proyecto de titulación se fundamenta en la constitución, leyes y normas como se detalla a continuación...”.

ARTÍCULO DE LA LOES	CONTEXTO
¿Qué regula la LOES? ART. 1 ÁMBITO	Esta Ley regula el sistema de educación superior en el país, a los organismos e instituciones que lo integran; determina derechos, deberes y obligaciones de las personas naturales y jurídicas, y establece las respectivas sanciones por el incumplimiento de las disposiciones contenidas en la Constitución y la presente Ley ARTICULO 1
¿Cuál es el Objeto de esta Ley? ART. 2 OBJETO	Esta Ley tiene como objeto definir sus principios, garantizar el derecho a la educación superior de calidad que propenda a la excelencia, al acceso universal, permanencia, movilidad y egreso sin discriminación alguna.
<u>Entre las funciones</u> ART. 4 DERECHO A LA EDUCACION SUPERIOR	a) Garantizar el derecho a la educación superior mediante la docencia, la investigación y su vinculación con la sociedad, y asegurar crecientes niveles de calidad, excelencia académica y pertinencia; n) Garantizar la producción de pensamiento y conocimiento articulado con el pensamiento universal; y, ñ) Brindar niveles óptimos de calidad en la formación
Principio de Igualdad y Principio de Calidad	El principio de igualdad de oportunidades consiste en garantizar a todos los actores del Sistema de Educación Superior las mismas posibilidades en el acceso, permanencia, movilidad y egreso del sistema, sin discriminación de género, credo, orientación sexual, etnia, cultura, preferencia política, condición socioeconómica o discapacidad. El principio de calidad consiste en la búsqueda constante y sistemática de la excelencia, la pertinencia, producción óptima, transmisión del conocimiento y desarrollo del pensamiento mediante la autocrítica, la crítica externa y el mejoramiento permanente
ART. 87	Como requisito previo a la obtención del título, los y las estudiantes deberán acreditar servicios a la comunidad mediante prácticas o pasantías pre profesionales. debidamente monitoreadas. en los campos de su especialidad, de conformidad con los lineamientos generales definidos por el Consejo de Educación Superior.
ARTÍCULO 19.- DEL REGLAMENTO.- NÓMINA DE GRADUADOS Y NOTIFICACIÓN A LA SENESCYT	Las instituciones de educación superior notificarán obligatoriamente a la SENESCYT la nómina de los graduados y las especificaciones de los títulos que expida, en un plazo no mayor de treinta días contados a partir de la fecha de graduación. (...) este será el único medio oficial a través del cual se verificará el reconocimiento y validez del título en el Ecuador.

ARTÍCULO 144 PRINCIPIOS	Art. 144.- Tesis Digitalizadas.- Todas las instituciones de educación superior estarán obligadas a entregar las tesis que se elaboren para la obtención de títulos académicos de grado y posgrado en formato digital para ser integradas al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
------------------------------------	---

Elaboración: Investigadores.

Fuente: Ley Orgánica de Educación Superior.

ARTÍCULO DE LA CONSTITUCIÓN	CONTEXTO
ARTÍCULO 22	Establece: las personas tienen derecho a desarrollar su capacidad creativa, al ejercicio digno y sostenido de las actividades culturales y artísticas, y a beneficiarse de la protección de los derechos morales y patrimoniales que les correspondan por las producciones científicas, literarias o artísticas de su autoría.
ARTÍCULO 26	La educación es un derecho de las personas a lo largo de su vida y un deber ineludible e inexcusable del Estado. Constituye un área prioritaria de la política pública y de la inversión estatal, garantía de la igualdad e inclusión social y condición indispensable para el buen vivir.
ARTÍCULO 28	La educación responderá al interés público y no estará al servicio de intereses individuales y corporativos. Se garantizará el acceso universal, permanencia, movilidad y egreso sin discriminación alguna
ARTÍCULO 350	El sistema de educación superior tiene como finalidad la formación académica y profesional con visión científica y humanista; la investigación científica y tecnológica; la innovación, promoción, desarrollo y difusión de los saberes y las culturas; la construcción de soluciones para los problemas del país, en relación con los objetivos del régimen de desarrollo
ARTÍCULO 355 primer y segundo inciso	El Estado reconocerá a las universidades y escuelas politécnicas autonomía académica, administrativa, financiera y orgánica, acorde con los objetivos del régimen de desarrollo y los principios establecidos en la Constitución

Elaboración: Investigadores.

Fuente: Ley Orgánica de Educación Superior.

FACTIBILIDAD LEGAL. - Comprende la viabilidad legal del proyecto, es decir, conocer los alcances y limitaciones relacionadas con el desarrollo del mismo.

- La viabilidad legal busca principalmente determinar la existencia de alguna restricción legal en la realización de un proyecto.
- Se busca determinar la existencia de normas o regulaciones legales que impidan la ejecución u operación del proyecto.
- Promover el desarrollo de proyectos sin problemas y dentro de las disposiciones legales.
- Pueden ser registrados y patentados.
- Este proyecto no transgrede ninguna norma, leyes o reglamentos establecidos en la Constitución del Ecuador ni en estamentos legales, por tanto, es factible su desarrollo y aplicación.

CODIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INVENCION

Artículo 104.- Obras susceptibles de protección. - La protección reconocida por el presente Título recae sobre todas las obras literarias, artísticas y científicas, que sean originales y que puedan reproducirse o divulgarse por cualquier forma o medio conocido o por conocerse. 12.- SOFTWARE

Artículo 131.- Protección de software. - El software se protege como obra literaria. Dicha protección se otorga independientemente de que hayan sido incorporados en un ordenador y cualquiera sea la forma en que estén expresados, ya sea como código fuente; es decir, en forma legible por el ser humano; o como código objeto; es decir, en forma legible por máquina, ya sea sistemas operativos o sistemas aplicativos, incluyendo diagramas de flujo, planos, manuales de uso, y en general, aquellos elementos que conformen la estructura, secuencia y organización del programa. Se excluye de esta protección las formas estándar de desarrollo de software. En este sentido, los documentos y textos producidos en las Instituciones de Educación Superior desarrollados con el objeto de obtener sus grados académicos y/o trabajos de facultad, son autores intelectuales con el patrocinio de cada institución, por lo tanto, son acreedores a los derechos de protección intelectual dispuestos en la normativa vigente.

Adicionalmente, considere revisar las siguientes fuentes:

Artículos de la Constitución
Política vigente (Año 2018).

Se recomienda lo siguiente:

- ✓ Artículo 22
- ✓ Artículo 26
- ✓ Artículo 28
- ✓ Artículo 350
- ✓ Artículo 355 primer y segundo inciso
- ✓ Artículo 424 primer inciso

Se entiende por inciso a un párrafo.



Artículos de la Ley Orgánica de Educación Superior:

Se recomienda lo siguiente:

- ✓ Artículo 1
- ✓ Artículo 2
- ✓ Artículo 4
- ✓ Artículo 19
- ✓ Artículo 21
- ✓ Principio de Igualdad y Principio de Calidad
- ✓ Artículo 87
- ✓ Artículo 144
- ✓ Artículo 204

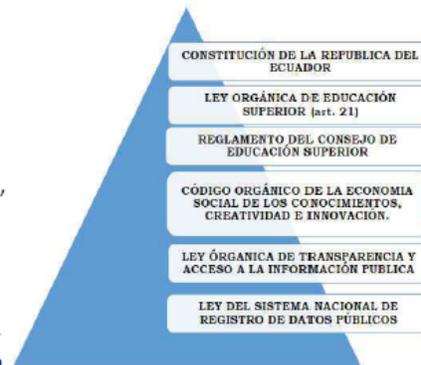
En la siguiente lámina se expresan textualmente los principios de Igualdad y de Calidad.



Artículos de la Ley Orgánica de Educación Superior

El **principio de igualdad** de oportunidades consiste en garantizar a todos los actores del Sistema de Educación Superior las mismas posibilidades en el acceso, permanencia, movilidad y egreso del sistema, sin discriminación de género, credo, orientación sexual, etnia, cultura, preferencia política, condición socio económica o discapacidad.

El **principio de calidad** consiste en la búsqueda constante y sistemática de la excelencia, la pertinencia, producción óptima, transmisión del conocimiento y desarrollo del pensamiento mediante la autocritica, la crítica externa y el mejoramiento permanente.

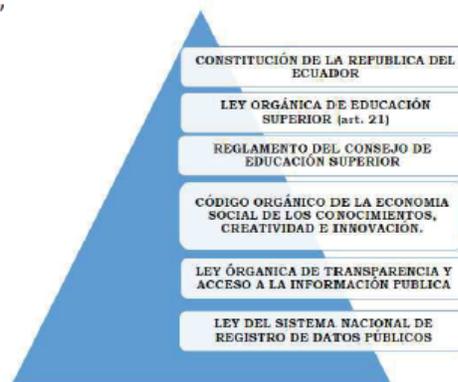


Artículos de Código Orgánico de la economía social de los conocimientos, creatividad e invención.

Se recomienda lo siguiente:

- ✓ Artículo 104
- ✓ Artículo 131

Cualquier otra Ley o reglamento que se desee citar, debe estar alineado a lo ya establecido.



Elaboración: Investigadores.
Fuente: Constitución del Ecuador (2010).

Anexo 3. Criterios éticos a utilizarse en el desarrollo del proyecto

Criterios	Características del criterio	Procedimientos
Credibilidad Valor de la verdad/autenticidad	Aproximación de los resultados de una investigación frente al fenómeno observado.	- Los resultados son reconocidos "verdaderos" por los participantes. - Observación continua y prolongada del fenómeno. - Triangulación.
Transferibilidad Aplicabilidad	Los resultados derivados de la investigación cualitativa no son generalizables sino transferibles.	- Descripción detallada del contexto y de los participantes. - Muestreo teórico. - Recogida exhaustiva de datos.
Consistencia Dependencia/replicabilidad	La complejidad de la investigación cualitativa dificulta la estabilidad de los datos. Tampoco es posible la replicabilidad del estudio.	- Triangulación - Empleo de evaluador externo. - Descripción detallada del proceso de recogida, análisis e interpretación de datos. - Reflexibilidad del investigador.
Confirmabilidad o Reflexibilidad Neutralidad / Objetividad	Los resultados de la investigación deben garantizar la veracidad de las descripciones realizadas por los participantes.	- Transcripciones textuales de las entrevistas. - Contratación de los resultados con la literatura existente. - Revisión de hallazgos por otros investigadores. - Identificación y descripción de limitaciones y alcances del investigador.
Relevancia	Permite evaluar el logro de los objetivos planteados y saber si se obtuvo un mejor conocimiento del fenómeno de estudio.	- Configuración de nuevos planteamiento teóricos o conceptuales. - Comprensión amplia del fenómeno. - Correspondencia entre la justificación y los resultados obtenidos.
Adecuación teórica-epistemológica	Correspondencia adecuada del problema por investigar y la teoría existente.	- Contratación de la pregunta con los métodos. - Ajustes de diseño.

Elaboración: Investigadores.

Fuente: Propia.

Anexo 4. Validación de expertos.



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS



CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Formulario de recolección de información por parte de expertos en desarrollo de software.

DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL.
Indicaciones: Lea detenidamente cada una de las interrogantes planteadas y marque con una X en la columna que considere adecuada.
Fecha: 17 Sept. 2020
Nombre: Juan Carlos Alave
Cargo (Opcional): Supervisión de Sistemas.

1. Seleccione las aplicaciones de mensajería instantánea que tiene instalado en su dispositivo móvil.

<input checked="" type="checkbox"/> WhatsApp
<input checked="" type="checkbox"/> Facebook Messenger
<input checked="" type="checkbox"/> Telegram
<input checked="" type="checkbox"/> Instagram
<input type="checkbox"/> Line
<input type="checkbox"/> Viber

2. En caso de no tener instalado telegram. ¿Estaría usted dispuesto a instalar la mencionada aplicación para realizar consultas a la comunidad SO-ES?

<input type="checkbox"/> Si
<input type="checkbox"/> No

3. ¿Qué tan útil considera usted contar con una aplicación donde puede realizar preguntas sobre mysql con la finalidad de obtener respuestas de la comunidad SO-ES a través de telegram?

<input type="checkbox"/> Extremadamente útil
<input checked="" type="checkbox"/> Muy útil
<input type="checkbox"/> Útil
<input type="checkbox"/> De poca utilidad
<input type="checkbox"/> Indiferente

4. ¿Cómo calificaría usted el tiempo de respuesta desde que envió su consulta hasta que el user-bot le presentó el resultado?

<input type="checkbox"/>	Excelente
<input checked="" type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

5. ¿Cómo calificaría usted la utilidad de las respuestas obtenidas?

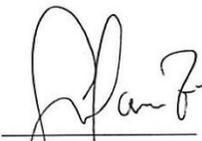
<input checked="" type="checkbox"/>	Excelente
<input type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

6. ¿Considera usted que esta aplicación mejorará el proceso de desarrollo de software?

<input type="checkbox"/>	Muy de acuerdo
<input checked="" type="checkbox"/>	De acuerdo
<input type="checkbox"/>	Indiferente
<input type="checkbox"/>	En desacuerdo
<input type="checkbox"/>	Muy en desacuerdo

7. ¿Usted recomendaría el uso del user-bot a sus compañeros desarrolladores de software?

<input checked="" type="checkbox"/>	Si
<input type="checkbox"/>	No



Firma del experto



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS



CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Formulario de recolección de información por parte de expertos en desarrollo de software.

DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL.
Indicaciones: Lea detenidamente cada una de las interrogantes planteadas y marque con una X en la columna que considere adecuada.
Fecha: 19-SEP-2020
Nombre: ALEJO LAZO A.
Cargo (Opcional):

1. Seleccione las aplicaciones de mensajería instantánea que tiene instalado en su dispositivo móvil.

<input checked="" type="checkbox"/>	WhatsApp
<input checked="" type="checkbox"/>	Facebook Messenger
<input checked="" type="checkbox"/>	Telegram
<input checked="" type="checkbox"/>	Instagram
<input type="checkbox"/>	Line
<input type="checkbox"/>	Viber

2. En caso de no tener instalado telegram. ¿Estaría usted dispuesto a instalar la mencionada aplicación para realizar consultas a la comunidad SO-ES?

<input type="checkbox"/>	Si
<input type="checkbox"/>	No

3. ¿Qué tan útil considera usted contar con una aplicación donde puede realizar preguntas sobre mysql con la finalidad de obtener respuestas de la comunidad SO-ES a través de telegram?

<input type="checkbox"/>	Extremadamente útil
<input checked="" type="checkbox"/>	Muy útil
<input type="checkbox"/>	Útil
<input type="checkbox"/>	De poca utilidad
<input type="checkbox"/>	Indiferente

4. ¿Cómo calificaría usted el tiempo de respuesta desde que envió su consulta hasta que el user-bot le presentó el resultado?

<input type="checkbox"/>	Excelente
<input checked="" type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

5. ¿Cómo calificaría usted la utilidad de las respuestas obtenidas?

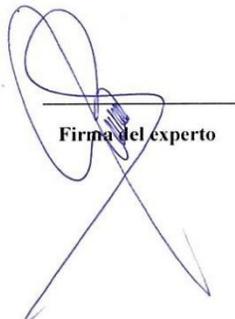
<input type="checkbox"/>	Excelente
<input checked="" type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

6. ¿Considera usted que esta aplicación mejorará el proceso de desarrollo de software?

<input checked="" type="checkbox"/>	Muy de acuerdo
<input type="checkbox"/>	De acuerdo
<input type="checkbox"/>	Indiferente
<input type="checkbox"/>	En desacuerdo
<input type="checkbox"/>	Muy en desacuerdo

7. ¿Usted recomendaría el uso del user-bot a sus compañeros desarrolladores de software?

<input checked="" type="checkbox"/>	Si
<input type="checkbox"/>	No


Firma del experto



Formulario de recolección de información por parte de expertos en desarrollo de software.

DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL.
Indicaciones: Lea detenidamente cada una de las interrogantes planteadas y marque con una X en la columna que considere adecuada.
Fecha: 17-Septiembre-2020
Nombre: José Miguel Moreno Plaza
Cargo (Opcional):

1. Seleccione las aplicaciones de mensajería instantánea que tiene instalado en su dispositivo móvil.

<input checked="" type="checkbox"/>	WhatsApp
<input checked="" type="checkbox"/>	Facebook Messenger
<input checked="" type="checkbox"/>	Telegram
<input type="checkbox"/>	Instagram
<input type="checkbox"/>	Line
<input type="checkbox"/>	Viber

2. En caso de no tener instalado telegram. ¿Estaría usted dispuesto a instalar la mencionada aplicación para realizar consultas a la comunidad SO-ES?

<input type="checkbox"/>	Si
<input type="checkbox"/>	No

3. ¿Qué tan útil considera usted contar con una aplicación donde puede realizar preguntas sobre mysql con la finalidad de obtener respuestas de la comunidad SO-ES a través de telegram?

<input type="checkbox"/>	Extremadamente útil
<input checked="" type="checkbox"/>	Muy útil
<input type="checkbox"/>	Útil
<input type="checkbox"/>	De poca utilidad
<input type="checkbox"/>	Indiferente

4. ¿Cómo calificaría usted el tiempo de respuesta desde que envió su consulta hasta que el user-bot le presentó el resultado?

<input type="checkbox"/>	Excelente
<input type="checkbox"/>	Muy Bueno
<input checked="" type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

5. ¿Cómo calificaría usted la utilidad de las respuestas obtenidas?

<input type="checkbox"/>	Excelente
<input checked="" type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

6. ¿Considera usted que esta aplicación mejorará el proceso de desarrollo de software?

<input type="checkbox"/>	Muy de acuerdo
<input checked="" type="checkbox"/>	De acuerdo
<input type="checkbox"/>	Indiferente
<input type="checkbox"/>	En desacuerdo
<input type="checkbox"/>	Muy en desacuerdo

7. ¿Usted recomendaría el uso del user-bot a sus compañeros desarrolladores de software?

<input checked="" type="checkbox"/>	Si
<input type="checkbox"/>	No



Firma del experto



CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Formulario de recolección de información por parte de expertos en desarrollo de software.

DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL.
Indicaciones: Lea detenidamente cada una de las interrogantes planteadas y marque con una X en la columna que considere adecuada.
Fecha: 17-Sept. 2020
Nombre: Victoria Murillo
Cargo (Opcional):

1. Seleccione las aplicaciones de mensajería instantánea que tiene instalado en su dispositivo móvil.

<input checked="" type="checkbox"/> WhatsApp
<input checked="" type="checkbox"/> Facebook Messenger
<input type="checkbox"/> Telegram
<input checked="" type="checkbox"/> Instagram
<input type="checkbox"/> Line
<input type="checkbox"/> Viber

2. En caso de no tener instalado telegram. ¿Estaría usted dispuesto a instalar la mencionada aplicación para realizar consultas a la comunidad SO-ES?

<input checked="" type="checkbox"/> Si
<input type="checkbox"/> No

3. ¿Qué tan útil considera usted contar con una aplicación donde puede realizar preguntas sobre mysqli con la finalidad de obtener respuestas de la comunidad SO-ES a través de telegram?

<input checked="" type="checkbox"/> Extremadamente útil
<input type="checkbox"/> Muy útil
<input type="checkbox"/> Útil
<input type="checkbox"/> De poca utilidad
<input type="checkbox"/> Indiferente

4. ¿Cómo calificaría usted el tiempo de respuesta desde que envió su consulta hasta que el user-bot le presentó el resultado?

<input checked="" type="checkbox"/>	Excelente
<input type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

5. ¿Cómo calificaría usted la utilidad de las respuestas obtenidas?

<input checked="" type="checkbox"/>	Excelente
<input type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

6. ¿Considera usted que esta aplicación mejorará el proceso de desarrollo de software?

<input checked="" type="checkbox"/>	Muy de acuerdo
<input type="checkbox"/>	De acuerdo
<input type="checkbox"/>	Indiferente
<input type="checkbox"/>	En desacuerdo
<input type="checkbox"/>	Muy en desacuerdo

7. ¿Usted recomendaría el uso del user-bot a sus compañeros desarrolladores de software?

<input checked="" type="checkbox"/>	Si
<input type="checkbox"/>	No



Firma del experto



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS



CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Formulario de recolección de información por parte de expertos en desarrollo de software.

DESARROLLO DE UN USER-BOT QUE PERMITA RESPONDER PREGUNTAS DE STACK OVERFLOW EN ESPAÑOL.
Indicaciones: Lea detenidamente cada una de las interrogantes planteadas y marque con una X en la columna que considere adecuada.
Fecha: 17/09/2020
Nombre: Guillermo Veliz S.
Cargo (Opcional): OFICIAL

1. Seleccione las aplicaciones de mensajería instantánea que tiene instalado en su dispositivo móvil.

<input checked="" type="checkbox"/>	WhatsApp
<input checked="" type="checkbox"/>	Facebook Messenger
<input type="checkbox"/>	Telegram
<input checked="" type="checkbox"/>	Instagram
<input type="checkbox"/>	Line
<input type="checkbox"/>	Viber

2. En caso de no tener instalado telegram. ¿Estaría usted dispuesto a instalar la mencionada aplicación para realizar consultas a la comunidad SO-ES?

<input checked="" type="checkbox"/>	Si
<input type="checkbox"/>	No

3. ¿Qué tan útil considera usted contar con una aplicación donde puede realizar preguntas sobre mysql con la finalidad de obtener respuestas de la comunidad SO-ES a través de telegram?

<input type="checkbox"/>	Extremadamente útil
<input checked="" type="checkbox"/>	Muy útil
<input type="checkbox"/>	Útil
<input type="checkbox"/>	De poca utilidad
<input type="checkbox"/>	Indiferente

4. ¿Cómo calificaría usted el tiempo de respuesta desde que envió su consulta hasta que el user-bot le presentó el resultado?

<input type="checkbox"/>	Excelente
<input checked="" type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

5. ¿Cómo calificaría usted la utilidad de las respuestas obtenidas?

<input type="checkbox"/>	Excelente
<input checked="" type="checkbox"/>	Muy Bueno
<input type="checkbox"/>	Bueno
<input type="checkbox"/>	Malo
<input type="checkbox"/>	Muy malo

6. ¿Considera usted que esta aplicación mejorará el proceso de desarrollo de software?

<input checked="" type="checkbox"/>	Muy de acuerdo
<input type="checkbox"/>	De acuerdo
<input type="checkbox"/>	Indiferente
<input type="checkbox"/>	En desacuerdo
<input type="checkbox"/>	Muy en desacuerdo

7. ¿Usted recomendaría el uso del user-bot a sus compañeros desarrolladores de software?

<input checked="" type="checkbox"/>	Si
<input type="checkbox"/>	No


Firma del experto

Nota: Encuesta realizada por los expertos.

Elaboración: Investigadores.

Fuente: Propia.

Anexo 5. Estudios Seleccionados

Tabla 10 Estudios Utilizados en el Mapeo Sistemático

Nº	Nombre artículo	Autores	Año
1	<u>Análisis supervisado de sentimientos políticos en español: clasificación en tiempo real de tweets basada en aprendizaje automático</u>	Carlos Arcila-Calderón, Félix Ortega-Mohedano, Javier Jiménez-Amores y Sofía Trullenque	2017
2	<u>Carlos Arcila-Calderón, Félix Ortega-Mohedano, Javier Jiménez-Amores y Sofía Trullenque</u>	Enric Hernández	2019
3	<u>Un detector de la unidad central de un texto basado en técnicas de aprendizaje automático en textos científicos para el euskera</u>	Kepa Bengoetxea, Aitziber Atucha, Mikel Iruskietta	2016
4	<u>LOS MÉTODOS DE APRENDIZAJE AUTOMÁTICO SUPERVISADO EN LA CLASIFICACIÓN TEXTUAL SEGÚN EL GRADO DE ESPECIALIZACIÓN</u>	Sergio Rodríguez-Tapia; Julio Camacho-Cañamó	2017
5	<u>Técnicas de aprendizaje de máquina utilizadas para la minería de texto</u>	Angel Freddy Godoy Viera	2017
6	<u>Aplicación de técnicas de machine learning a la ciberseguridad: Aprendizaje supervisado para la detección de amenazas web mediante clasificación basada en árboles de decisión</u>	Dueñas Quesada, José María	2020
7	<u>Algoritmos de aprendizaje automático para análisis y predicción de datos</u>	LJ Sandoval Serrano - Revista Tecnológica	2018
8	<u>Búsqueda de estrategias para la clasificación del contenido en foros técnicos de discusión</u>	N Martínez Carod, GN Aranda, A Cechich	2017
9	<u>Un enfoque de Machine Learning en el desarrollo de Sistema Recomendadores para Procesos de Investigación</u>	M Danilo, T Liliana, A Mayra...	2019
10	<u>Análisis de las técnicas de aprendizaje automático aplicadas en la ingeniería de requerimientos de software: un mapeo sistemático de la literatura</u>	MG Gramajo, LC Ballejos, M Ale	2018

11	<u>Análisis de similitud en documentos de texto mediante técnicas de ciencia de datos basadas en aprendizaje profundo (deep learning)</u>	AB Calibar, J Holleger, RO Klenzi	2018
12	<u>Aplicación del análisis sintáctico automático en la atribución de autoría de mensajes en redes sociales.</u>	FAC Velásquez, JPZ De Paz, MS Castillo	2017
13	<u>Staqc: A systematically mined question-code dataset from stack overflow</u>	Z Yao, DS Weld, WP Chen, H Sun	2018
14	<u>A Systematic Literature Review on Using Machine Learning Algorithms for Software Requirements Identification on Stack Overflow</u>	A Ahmad, C Feng, M Khan, A Khan, A Ullah...	2020
15	<u>Mining testing questions on stack overflow</u>	PS Kochhar	2016
16	<u>Towards an Accurate Prediction of the Question Quality on Stack Overflow using a Deep-Learning-Based NLP Approach.</u>	L Tóth, B Nagy, D Janthó, L Vidács, T Gyimóthy	2019
17	<u>Personality Prediction from Stack Overflow by using Naïve Bayes Theorem in Data Mining</u>	S Katiyar, S Kumar, H Walia	2020
18	<u>Aprendizaje Automático: Una Contribución a la Investigación Operativa</u>	A Talavera, A Luna	2020
19	<u>Análisis del comportamiento de diferentes algoritmos de aprendizaje automático para catalogar delitos en la zona metropolitana</u>	Bel'em Priego S´anchez, Stephany Anaya Garc´ia, Jos'e A. Reyes-Ortiz	2018
20	<u>Creación de un Corpus de Opiniones con Emociones usando Aprendizaje Automático</u>	Alejandra C. Cardoso, María Lorena Talamé, Matías N. Amor, Agustina Monge	2019
21	<u>Clasificación Automática de Textos de Desastres Naturales en México</u>	Alberto Téllez-Valero, Manuel Montes-y-Gómez, Olac Fuentes-Chávez, Luis Villaseñor-Pineda	2015

Nota: Estudios Utilizados en el Mapeo Sistemático.

Elaboración: Investigadores.

Fuente: Propia.

Anexo 6. Acta de entrega y recepción definitiva

En la ciudad de Guayaquil, a ____ días del mes de _____ de _____

Por el presente documento.

Los estudiantes no titulados de la Carrera de Ingeniería en Sistemas Computacionales **Sánchez Piguave Alexander Sleiter** con cédula de identidad N° **0954179859** y **Almache Carranza Carlos Ignacio** con cédula de identidad N° **0953705787** hacemos la entrega del código fuente del proyecto de titulación a la Dirección de la Carrera de Ingeniería en Sistemas Computacionales en un medio magnético.

Los códigos del programa/producto que se encargaron por compromiso al estar inserto en el proceso de titulación desde fecha ____ de _____.

Para efectos de dar cumplimiento a la entrega del código fuente, cedo todos los derechos de explotación sobre el programa y, en concreto, los de transformación, comunicación pública, distribución y reproducción, de forma exclusiva, con un ámbito territorial nacional.

Sánchez Piguave Alexander Sleiter

N° 0954179859

Almache Carranza Carlos Ignacio

N° 0953705787

Anexo 7. Artículo científico

Desarrollo de un USER-BOT que permita responder preguntas de Stack Overflow en español

Sánchez Piguave Alexander Sleiter, Almache Carranza Carlos Ignacio

alexander.sanchezp@ug.edu.ec, carlos.amachec@ug.edu.ec

Universidad de Guayaquil

Resumen

El trabajo de investigación tuvo como finalidad desarrollar un USER-BOT que implemente algoritmos de machine learning como “Naive-Bayes” con la finalidad de responder preguntas introducidas por usuarios a través de la aplicación de mensajería instantánea Telegram, con información almacenada en la comunidad en línea “Stack Overflow” en español. “Stack Overflow” en español es una comunidad virtual donde miles de usuarios realizan publicaciones de preguntas, respuestas e información relacionada a los procesos de desarrollo de software en sus diferentes etapas. La búsqueda de información en la web acerca de temas de desarrollo de software puede tomar una cantidad considerable de tiempo por cuanto el usuario debe identificar, entre los miles de “*post*” existentes la información relevante requerida que se ajuste a su consulta. Un Bot que ejecute algoritmos de machine learning que realice las tareas de explorar, clasificar, ordenar, seleccionar y presentar información que más relevancia tiene para el usuario es una ayuda considerable a las personas en su trabajo habitual. Para el desarrollo del trabajo se realizó un análisis de la estructura de las publicaciones de la comunidad StackOverflow para ajustar un repositorio de datos acorde a esta y se desarrolló los procesos automáticos de extracción y procesamiento de datos para que esté disponible para el BOT. Por otro lado, se desarrollaron los componentes en Python que interactúa con la información almacenada a través de algoritmos de “Naive-Bayes” para seleccionar la información relevante del repositorio. Finalmente, se configuró el BOT en “Telegram” utilizando el “Bot Father” para interactuar con los usuarios. Los resultados obtenidos por el modelo fueron validados a través de la técnica de juicio de expertos en la cual 5 profesionales en el área de desarrollo de software verificaron que el bot presenta resultados útiles.

Abstract

The purpose of the research work was to develop a USER-BOT that implements machine learning algorithms such as "Naive-Bayes" in order to answer questions entered by users through the instant messaging application Telegram, with information stored in the community in "Stack Overflow" line in Spanish. "Stack Overflow" in Spanish is a virtual community where thousands of users post questions, answers, and information related to software development processes in its different stages. Searching for information on the web about software development issues can take a considerable amount of time because the user must identify, among the thousands of existing "posts", the relevant information required that fits her query. A Bot that runs machine learning algorithms that performs the tasks of exploring, classifying, sorting, selecting and presenting information that is most relevant to the user is a considerable help to people in their daily work. For the development of the work, an analysis of the structure of the StackOverflow community publications was carried out to adjust a data repository according to this and the automatic data extraction and processing processes were developed so that it is available to the BOT. On the other hand, the components were developed in Python that interacts with the stored information through "Naive-Bayes" algorithms to select the relevant information from the repository. Finally, the BOT was configured in "Telegram" using the "Bot Father" to interact with users. The results obtained by the model were validated through the expert judgment technique in which 5 professionals in the software development area verified that the USER-BOT presents useful results.

Introducción

Durante el desarrollo de software en cualquier lenguaje de programación o plataforma tecnológica, es común que se presenten una serie de problemas técnicos en la compilación / ejecución del código creado por el profesional que causa que se consuma una gran cantidad de tiempo del programador tratando de corregir los errores o subsanar los problemas planteados. También suele ocurrir que los miembros del equipo de desarrollo desconozcan ciertos aspectos de las herramientas que deben utilizar durante el proceso lo cual puede ser compensado con la información a la cual pueda acceder para solventar la falta de información. Inclusive los programadores más experimentados pueden encontrarse en el proceso con algún inconveniente técnico que no le permita seguir avanzando en el desarrollo del sistema. Para corregir los errores o enmendar los problemas técnicos presentados, el programador habitualmente acude a la documentación del proveedor, si existiere, o a la información publicada que se ha puesto a

disposición de la comunidad. La última opción sirve mucho, sin embargo, la información obtenida debe ser evaluada adecuadamente por cuanto los usuarios de internet podrían subir información que no necesariamente es acertada al problema planteado, causando pérdida de tiempo a los que usuarios que realizan la consulta.

En internet existe una serie de sitios web que permiten publicar información muy variada en cuanto a los temas que tratan y que están a disposición de las personas, como por ejemplo la enciclopedia virtual “Wikipedia” que permite a los usuarios agregar información de una amplia gama de temáticas. Sin embargo, la información registrada puede contener inconsistencias fundamento adecuado o las referencias bibliográficas acordes a la realidad o carezcan de fundamento teórico – técnico lo que lo convierte en una información no útil para los intereses del programador que consulta.

Hace muchos años uno de los buscadores más populares de internet denominado Yahoo ha venido implementando una plataforma de “preguntas y respuestas” denominado “Yahoo respuestas” en la cual la comunidad aporta a la plataforma con respuestas a preguntas registradas previamente. Actualmente esta plataforma se encuentra disponible en 12 idiomas. En esta herramienta las preguntas se ordenan por categorías para poder mejorar el tiempo de consulta.

En el año 2008 se funda “Stack Overflow” que es la comunidad en línea más grande y confiable para cualquier persona que codifique para aprender, compartir sus conocimientos y desarrollar sus carreras. Más de 50 millones de visitantes únicos acceden a la plataforma cada mes para ayudar a resolver problemas de codificación, desarrollar nuevas habilidades y encontrar oportunidades de trabajo. Stack Overflow se asocia con empresas para ayudarles a comprender, contratar, involucrar y habilitar a los desarrolladores del mundo. En esta plataforma los desarrolladores pueden compartir sus experiencias, realizar preguntas a errores de codificación y sus soluciones, además de mejorar sus habilidades en distintos lenguajes de programación. Uno de los aspectos más relevantes del sistema es lo que denomina “Reputación” cuyo término representa una mayor confianza a las respuestas en base a los votos de los usuarios.

El presente trabajo de titulación está constituido por 4 capítulos que se detallan a continuación:

Capítulo I: Se realiza el análisis de la problemática planteada en el estudio, los objetivos generales y específicos de la investigación, los alcances, la importancia y justificación del proyecto.

Capítulo II: Se detalla el marco teórico sobre el cual se fundamenta el estudio, así como el marco legal de la investigación, determinando la no violación de ninguna ley o reglamento del Ecuador.

Capítulo III: Se presenta la metodología utilizada en la investigación, la factibilidad del proyecto y los parámetros de evaluación de la propuesta.

Capítulo IV: Se presenta los resultados, conclusiones y recomendaciones del estudio.

Metodología

En internet se puede encontrar una gran cantidad de foros o sitios especializados en tecnologías de la información y comunicaciones, así como aquellos que tratan acerca de programación y configuración de sistemas, que proporcionan información para ayudar a los desarrolladores de software en las diferentes etapas de sus proyectos. Una de las comunidades web que ha crecido significativamente en cuanto al número de usuarios y reputación al mundo de la tecnología y programación es la denominada “Stack Overflow”. Fue lanzado en el año 2008 y actualmente está posicionada como una de las comunidades más importantes dedicada a la tecnología y programación, además, “está considerada como uno de 50 sitios web más populares del mundo, con más de 50 millones de usuarios activos que se ayudan mutuamente” (Martech, 2020).

Para poder utilizar los servicios de “Stack Overflow en español SO-ES” los usuarios deben registrarse en el sitio web y acceder a realizar y responder preguntas. Una vez que la pregunta haya sido respondida los usuarios de la comunidad, los usuarios pueden calificarlas lo que permite categorizar la información asignando una “reputación” al contenido registrado. Este método de trabajo garantiza que las respuestas van a estar validadas o respaldadas por un mayor número de personas de la comunidad y, por consecuencia, tiene mayor probabilidad de ser una respuesta correcta.

Stack Overflow posee un aspecto interesante, la reputación, este término es destacado de acuerdo con la cantidad de votos que poseen las preguntas y las respuestas, a mayor cantidad de votos relevantes o aciertos, mayor es la reputación en el sitio web. De hecho, el número de votos es un indicador para muchos aspectos, entre estos tenemos:

- Confianza de los usuarios.
- Habilidades de comunicación.
- Calidad y relevancia en preguntas y respuestas.
- Manejo de los temas.
- Nivel de experiencia y participación.

- Base de datos

Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 cuyo nombre está inspirado en el grupo de cómicos ingleses “Monty Python”. Es un lenguaje similar a Perl, pero con una sintaxis muy limpia y que favorece un código legible. Se trata de un lenguaje interpretado o de script, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos (González R. , 2016).

Procesamiento del lenguaje natural

El procesamiento del lenguaje natural (NLP, por sus siglas en inglés) consiste en desarrollar aplicaciones y servicios que puedan comprender los idiomas humanos. Ejemplos prácticos de procesamiento del lenguaje natural (NLP) son el reconocimiento de voz, la traducción del habla, la comprensión de oraciones completas, la comprensión de sinónimos de palabras coincidentes y la escritura de oraciones y párrafos completos gramaticalmente correctos.

Naïve Bayes es “uno de los algoritmos de aprendizaje inductivo más eficientes y efectivos. Simplifica considerablemente el aprendizaje mediante el supuesto de independencia de los atributos y, no obstante, compite en la práctica con clasificadores más sofisticados. Ha sido considerado además como uno de los 10 algoritmos de minería de datos más influyentes por varias razones” (Pereira-Toledo, López-Cabrera, & Quintero-Domínguez, 2017). De acuerdo con los mencionados autores, es “fácil de construir y no necesita de esquemas iterativos de estimación de parámetros”. Lo anterior implica que dicho algoritmo puede ser “aplicado a grandes bases de datos. Es también fácil de interpretar por usuarios no familiarizados con los clasificadores” (Pereira-Toledo, López-Cabrera, & Quintero-Domínguez, 2017).

Machine learning

De acuerdo con Russo, entre otros, “machine learning es un área de la inteligencia artificial que engloba un conjunto de técnicas que hacen posible el aprendizaje automático a través del entrenamiento con grandes volúmenes de datos” (Russo, y otros, 2018). En la actualidad existen modelos que utilizan esta técnica logrando excelentes resultados. Además, “La construcción de modelos de Machine Learning requiere adaptaciones propias debido a la naturaleza de los datos o a la problemática a la que se aplica” (Russo, y otros, 2018).

De acuerdo con Russo, entre otros, se pueden encontrar tres grandes grupos de algoritmos “Machine Learning”, que son:

Algoritmos supervisados: Estos algoritmos utilizan un “conjunto de datos de entrenamiento etiquetados (preclasificados), los cuales procesan para realizar predicciones sobre los mismos, corrigiéndolas cuando son incorrectas. El proceso de entrenamiento continúa hasta que el modelo alcanza un nivel deseado de precisión” (Russo, y otros, 2018).

Algoritmos semi-supervisados: Estos algoritmos combinan tanto datos etiquetados como no etiquetados para generar una función deseada o clasificador. Este tipo de modelos deben aprender las estructuras para organizar los datos, así como también realizar predicciones (Russo, y otros, 2018).

Algoritmos no supervisados: El conjunto de datos no se encuentra etiquetado y no se tiene un resultado conocido. Por ello deben deducir las estructuras presentes en los datos de entrada, lo puede conseguir a través de un proceso matemático para reducir la redundancia sistemáticamente u organizando los datos por similitud (Russo, y otros, 2018).

Gráfico 1 – Clasificación de machine learning



Nota: En el gráfico 1 se muestra la clasificación de machine learning

Fuente: Investigación

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Clasificador Naive Bayes

Corresponde a los algoritmos de machine learning supervisados por clasificación. El clasificador de Naive Bayes es utilizado para predecir eventos futuros mediante operaciones probabilísticas, este método evalúa la probabilidad de cada una de las clases, así como sus variables independientes. El algoritmo NB está compuesta de un nodo padre y n nodos hijos, donde el nodo principal representa a la variable de clase (variable dependiente) y sus hijos hacen referencia a las variables independientes. Pereira y otros, citando a Witten, mencionan que “El objetivo de la clasificación en el aprendizaje automático es entrenar un método determinado a partir de un conjunto de datos para construir un modelo que sea capaz de predecir uno de los valores nominales que pertenecen al dominio de un atributo llamado clase” (Pereira-Toledo, López-Cabrera, & Quintero-Domínguez, 2017).

Por otra parte, Salamanca y Velásquez indican que Naive Bayes es un “clasificador estadístico supervisado. Está basado en el teorema de bayes en la teoría de hipótesis de independencia entre las variables X_1, \dots, X_p .”(Salamanca & Velásquez, 2018).

El proyecto se ha desarrollado por fases, las cuales se detallan a continuación:

Preparación del ambiente

Las herramientas que se utilizaron para el presente estudio son:

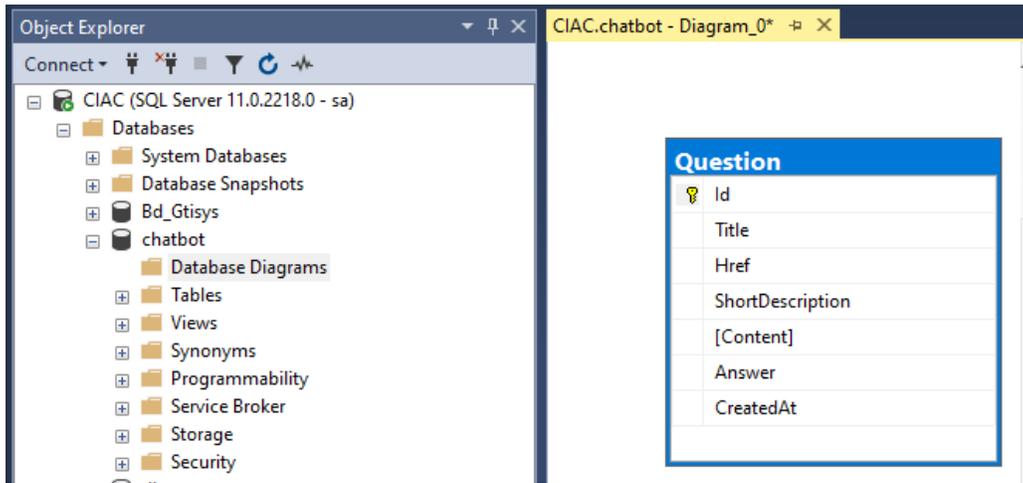
4. Estructura de datos.
5. Python 3.8.
6. Telegram.

Creación de estructuras de datos

Se realizó el diseño de la base de datos tomando como referencia la data de la red de comunidades Stack Exchange. Tomando como etiqueta la data que maneja el gestor de base MYSQLI. Como resultado a la recolección de datos obtenida se procedió a crear la base de datos llamada “CHATBOT” y con una tabla llamada “QUESTION.

En el Gráfico 2 se muestra la ilustración de la base de datos y la tabla creada.

Gráfico 2 Ilustración de la base de datos y la tabla creada.



Nota: Ilustración de la base de datos y la tabla creada.

Fuente: Investigación

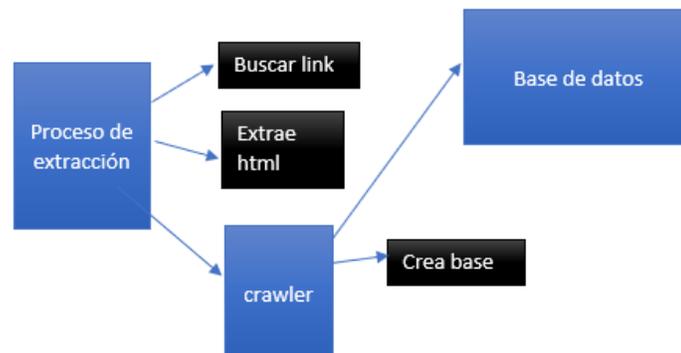
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Extracción de datos

Es el proceso a través del cual el USER-BOT recopila la información de preguntas y respuestas que contiene la comunidad en línea Stack Overflow en español, realiza un proceso de depuración y clasificación de acuerdo con los algoritmos implementados y realiza el almacenamiento en la base de datos anteriormente definida.

A continuación, en el gráfico 3 se presenta el proceso de extracción con el USER-BOT.

Gráfico 3 Proceso de extracción con USER-BOT.



Nota: Representación del Proceso de extracción con USER-BOT.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

Para el proceso de extracción:

- Se busca el enlace de la página, el cual es: <https://es.stackoverflow.com>.
- Se procedió a extraer la información en html con el siguiente filtro:
/questions/tagged/mysqli?page={page}&sort=votes&pagesize=50.

En el crawler realiza un proceso a través del cual se crea la base de datos automáticamente en el caso de al momento de ejecutarse el bot, en caso de que no exista previamente.

Por último, se almacena la información obtenida de la extracción en la base de datos.

Preprocesamiento de información

Para este proceso se tomó en consideración el análisis de contenido ya que, al trabajar con un conjunto de datos, está la posibilidad de encontrarnos con datos irrelevantes que perjudiquen el buen funcionamiento del USER-BOT.

Este proceso comienza su función al momento que el usuario envía su pregunta al USER-BOT.

A continuación, se indica los pasos de limpieza que se emplean:

5. Transformación de las letras mayúsculas a minúsculas.
6. Tokenización.
7. Eliminación de palabras irrelevantes.
8. Lematización.

El primer paso se utiliza para establecer una estandarización, el método que empleamos es “**lower()**” convierte todos los caracteres que estén en mayúsculas a minúsculas.

En el segundo paso se emplea “**tokenización**” para realizar la separación de cadenas de textos largas a cadenas de textos cortas.

La utilización del tercer paso ayuda a disminuir el tiempo de aprendizaje del USER-BOT dando como resultado que la respuesta dada por el bot sea más precisa. Para esto se utilizó la función “**stopwords**” que elimina del origen de datos las palabras “conectoras” que se repiten con frecuencia en un idioma.

Por último, se utilizó “**lematización**” que no ayuda a simplificar aún más las palabras enviadas por el usuario dando como resultado que el USER-BOT sea más preciso al momento de dar la respuesta.

Gráfico 4 Resultado.

```
message received ¿como hago una consulta en mysql?  
Pregunta dividida: ['como', 'hago', 'una', 'consulta', 'en', 'mysql']  
Palabras vacías: ['hago', 'consulta', 'mysql']  
Lematización: ['hag', 'consult', 'mysql']
```

Nota: Demostración del proceso de limpieza.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

En el Gráfico 4 se toma como ejemplo la pregunta: “¿Cómo hago una consulta en mysql?” dando como resultado lo mostrado en dicho gráfico.

Aprendizaje Automático

Durante esta fase se procedió a la construcción del machine learning usando el algoritmo de Bayes que nos ayuda con el aprendizaje automático que tiene como base una técnica de clasificación estadística llamada “**teorema de Bayes**” usando como ayuda “**Pandas**” librería destinada al análisis de datos, ayuda a proporcionar una estructura de dato flexible que permite realizar un trabajo más eficiente, trabaja en conjunto con “**Numpy**” librería que ayuda a un potente tipo matricial a Python.

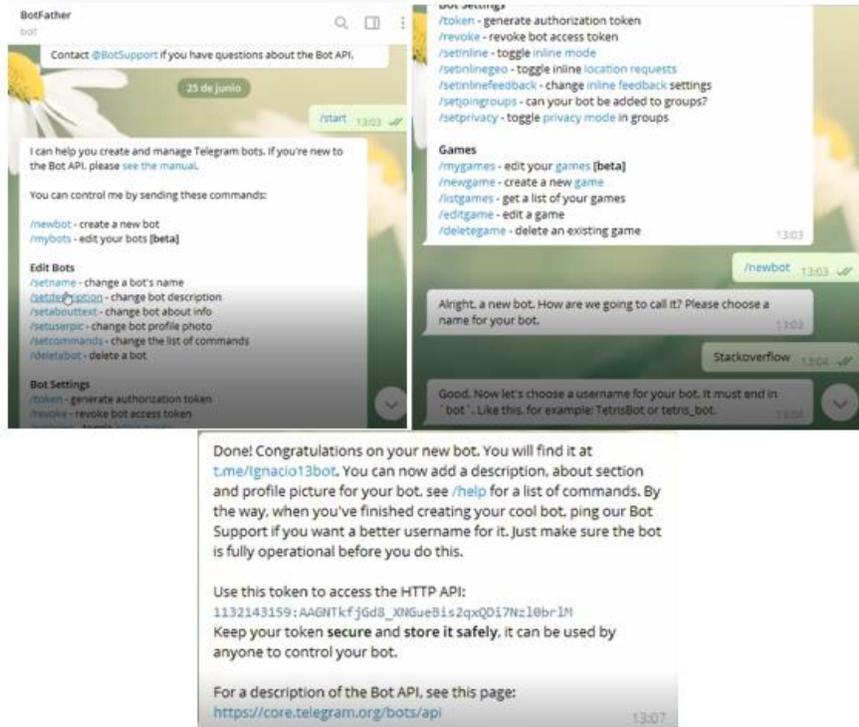
La utilización del “**teorema de Bayes**” proporciona una manera de calcular la probabilidad de la hipótesis basada en su probabilidad previa, las probabilidades de observar diversos datos dada la hipótesis y los datos observados en sí.

Registro y configuración del bot

Para la configuración del bot se utilizó el bot “BotFather” que se encuentra disponible en el API de Telegram que, como su nombre lo indica es el “papá” de todos los bot y se utiliza para poder configurar cualquier bot que se quiera implementar.

En el gráfico 5 se visualiza el proceso para el registro del bot dentro de “BotFather” donde, una vez registrado, genera un código de acceso o “token” para utilizarlo en el código fuente del Bot con la finalidad de que Telegram “reconozca” el bot y pueda ser utilizado por los usuarios finales.

Gráfico 5 Registro del bot.



Nota: Proceso para el registro del bot dentro de “BotFather”.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

En el gráfico 6 se muestra el código donde crea una instancia y se realiza la conexión entre el USER-BOT con Telegram.

Gráfico 6 Conexión a Telegram.

```
def train_bot(file):
    logger.debug('creating bot')
    telegram_token = '1132143159:AAGNTkfjGd8_XNGueBis2qxQDi7Nz10br1M'
    updater = Updater(
        token = telegram_token,
        use_context = True
    )
```

Nota: Conexión entre el USER-BOT con Telegram.

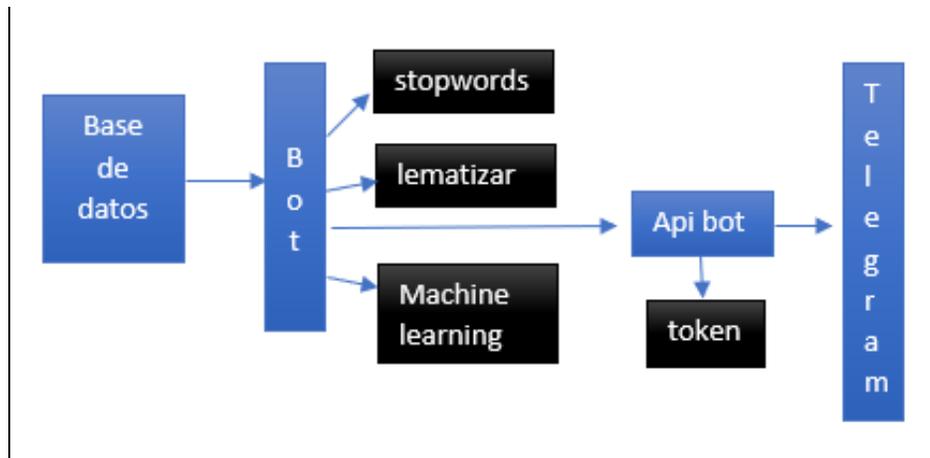
Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Consulta

A continuación, en el gráfico 7 se muestra el proceso de consulta.

Gráfico 7 Proceso de consulta con USER-BOT.



Nota: Proceso de consulta con USER-BOT.

Fuente: Investigación

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

La Base de datos es el repositorio donde se almacena la información que el *bot* obtuvo de la comunidad en línea Stack Overflow en español mientras tanto el *bot* es el conjunto de programas que realizan toda la lógica de recopilación, análisis, ordenamiento, almacenamiento de la información y la interacción con el usuario.

Está conformado por una serie de procesos lógicos como son el “limpiado” de datos de las palabras que ingrese el usuario denominado “stopwords”, el proceso de “lematización” y el proceso de machine learning que obtiene las respuestas adecuadas para poder responder la consulta del usuario.

El *api bot* es el método y librería que pone a disposición Telegram para el uso de todos los desarrolladores que permiten a los usuarios interactuar con el programa y el *bot* en Telegram que es la aplicación de mensajería instantánea para dispositivos móviles que se utilizará como interfaz de usuario para el USER-BOT.

Interacción con el Bot

Una vez que se ha desarrollado las etapas previas, el *bot* se encuentra en etapa de funcionamiento y puede ser utilizado a través de la aplicación de mensajería “Telegram”. Para utilizar el bot se

debe buscar por su nombre en la interfaz de “búsqueda de contactos”. En el gráfico 8 se presenta la interfaz del usuario con el *bot*.

Gráfico 8 Interfaz User-Bot.



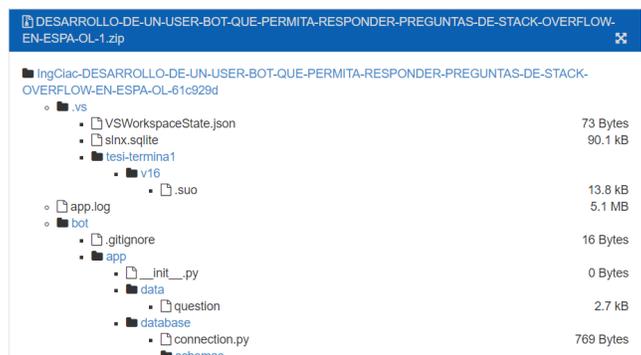
Nota: Interfaz que visualiza el usuario.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El código fuente del proyecto presentado se puede encontrar en el repositorio Git de zenodo (Almache & Sánchez, 2020) que tiene la siguiente estructura y el cual se representa en el gráfico 9:

Gráfico 9 – Estructura del programa en zenodo



Nota: Interfaz.

Fuente: Investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El enlace para acceder al repositorio es el siguiente:

<https://zenodo.org/badge/latestdoi/302775912>

Procesamiento y análisis

Para verificar que la aplicación funcione de manera correcta se utilizó la técnica de juicio de expertos poniendo a consideración de cinco profesionales que trabajan en actividades relacionadas al desarrollo de software para validar la propuesta de investigación. A cada uno de los expertos entrevistados se les solicitó que utilicen el USER-BOT a través de su aplicación de mensajería instantánea “Telegram” y, además, se les presentó una serie de preguntas relacionadas al proyecto de investigación. Cada uno de los profesionales que participaron en el estudio llenaron un formulario de recolección de datos con la finalidad de que sus opiniones formen parte del resultado de la investigación.

Resultados

Para validar los resultados obtenidos por el USER-BOT se ha utilizado la técnica de juicio de expertos a través de la cual se ha consultado a profesionales que ejercen sus actividades en el sector de desarrollo de software en la ciudad de Guayaquil, quienes han utilizado el USER-BOT y respondido algunas consultas realizadas en un formato de recolección de información. A continuación, se presenta los resultados obtenidos a través de esta técnica.

Tiempo de respuesta: la gran mayoría de los interesados confirmaron que el tiempo que se toma el *bot* desde el envío de la pregunta hasta la presentación de resultados es adecuado (Excelente y muy bueno) lo que les permite realizar búsquedas ágilmente.

Los criterios que se tomaran en cuenta son los siguientes:

Criterio de toma de decisión en el caso que se utilice el criterio de expertos:

- Si $p \leq 0.05$ entonces se rechaza H_0 y se acepta H_1 y se dice significativo (Si la probabilidad correspondiente al valor calculado por la prueba estadística es menor o igual que su respectivo valor crítico al nivel de 0.05, entonces se rechaza H_0 y se dice significativo).
- Si $p > 0.05$ entonces se acepta H_0 y se dice no significativo (Si la probabilidad correspondiente al valor calculado por la prueba estadística es mayor que su respectivo valor crítico al nivel de 0.05, entonces se acepta H_0 y se dice no significativo).

Donde:

p: valor de probabilidad

α : Nivel de significancia, con un valor de 0.05

H0: ¿El modelo realizado mediante machine learning presenta respuestas de Stack Overflow en español correctas?

H1: ¿El modelo realizado mediante machine learning NO presenta respuestas de Stack Overflow en español correctas?

Las primeras preguntas planteadas a los expertos se enfocan a determinar si tienen instalada la aplicación Telegram en sus teléfonos y, en caso de no tenerla, si estarían dispuestos a instalarla para las pruebas correspondientes. Los expertos participantes del estudio que no tenían Telegram instalado accedieron a instalar la aplicación de mensajería instantánea.

A continuación, las preguntas:

1. Seleccione las aplicaciones de mensajería instantánea que tiene instalado en su dispositivo móvil.
2. En caso de no tener instalado telegram. ¿Estaría usted dispuesto a instalar la mencionada aplicación para realizar consultas a la comunidad SO-ES?

Para calcular los resultados de juicio de expertos, los datos fueron ingresados a la “calculadora estadística” que nos ofrece el sitio web especializado “Social Science Statistics” que, en una de sus secciones tienen a disposición la calculadora Chi-Cuadrado (<https://www.socscistatistics.com/tests/>).

Tabla 1 de contingencia. Pregunta #5

	correcto	incorrecta	Totales de filas marginales
Con aprendizaje automático	4 (3,5) [0,07]	1 (1,5) [0,17]	5
Sin aprendizaje automático	3 (3,5) [0,07]	2 (1,5) [0,17]	5
Totales de columna marginal	7	3	10 (Gran Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El estadístico de chi-cuadrado es 0.4762.

El valor p es 0.490153

P muestra un valor mayor a α ($0.49 > 0,05$), por lo tanto, el resultado no es significativo en $p < 0.05$. de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Tabla 2 de contingencia. Pregunta #6

	a tiempo	a destiempo	<i>Marginal Row Totals</i>
Con Machine learning	3 (2.5) [0.1]	2 (2.5) [0.1]	5
Sin machine learning	2 (2.5) [0.1]	3 (2.5) [0.1]	5
Marginal Column Totals	5	5	10 (Grand Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

La estadística de chi-cuadrado es 0,4.

El valor p es 0.527089.

P muestra un valor mayor a α ($0.4 > 0,05$), por lo tanto, el resultado no es significativo en $p < 0.05$. de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Tabla 3 de contingencia. Pregunta #7

	Útil	No útil	<i>Marginal Row Totals</i>
Con Machine learning	4 (3.5) [0.07]	1 (1.5) [0.17]	5
Sin machine learning	3 (3.5) [0.07]	2 (1.5) [0.17]	5
Marginal Column Totals	7	3	10 (Grand Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El estadístico de chi-cuadrado es 0.4762.

El valor p es 0.490153.

P muestra un valor mayor a α ($0.490153 > 0,05$), por lo tanto, el resultado no es significativo en $p < 0.05$. de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Tabla 4 de contingencia. Pregunta #8

	Mejora	No mejora	Marginal Row Totals
Con Machine learning	4 (3) [0.33]	1 (2) [0.5]	5
Sin machine learning	2 (3) [0.33]	3 (2) [0.5]	5
Marginal Column Totals	6	4	10 (Grand Total)

Nota: Elaboración propia y fuente corresponde a datos propios de la investigación.

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos.

El estadístico de chi-cuadrado es 1.6667.

El valor p es 0.196706

P muestra un valor mayor a α ($0.196706 > 0,05$), por lo tanto, el resultado no es significativo en $p < 0.05$. de acuerdo con los criterios esto significa que, se acepta H_0 , dando como resultado que el modelo realizado mediante machine learning para responder preguntas de SO-ES es correcta.

Dataset

A continuación, en el gráfico 14 se describen los dataset usados, en el cual se toma en cuenta las preguntas sin contestar de la comunidad en línea Stack Overflow en español (SO-ES) que contengan la etiqueta mysql, tomadas de la página <https://es.stackoverflow.com/>.

Gráfico 10 Dataset1

id_no	id_posible	similar
120671	26146	1
240887	224130	1
283513	104647	1
329622	143686	0
55103	43323	0
320035	224646	0

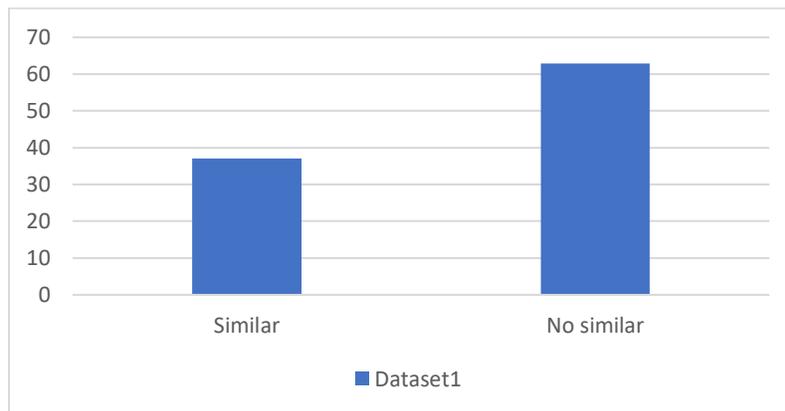
Nota: *Id_no* = Id de las preguntas;
Id_posible = Posible repuesta a la pregunta sin contestar;
Similar = Si *id_no* es similar a *id_posible*=1 caso contrario = 0.
Fuente: Investigación
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Durante la recolección y filtrado de preguntas se obtuvo un total de 234 preguntas sin contestar de las cuales se tomó en cuenta 186 preguntas.

Las 48 preguntas restantes no se las tomó en cuenta por uno de los siguientes motivos:

- Fueron eliminadas por no cumplir los estándares requeridos.
- No se encontraron resultados para las preguntas.

Gráfico 11 Valores Dataset1



Nota:
Fuente: Elaboración propia
Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

De las 186 preguntas seleccionadas se procedió a realizar la validación del cual se obtuvo que el 37,1% de las preguntas tienen similitud y el 62,9% restante no.

Gráfico 12 Dataset2

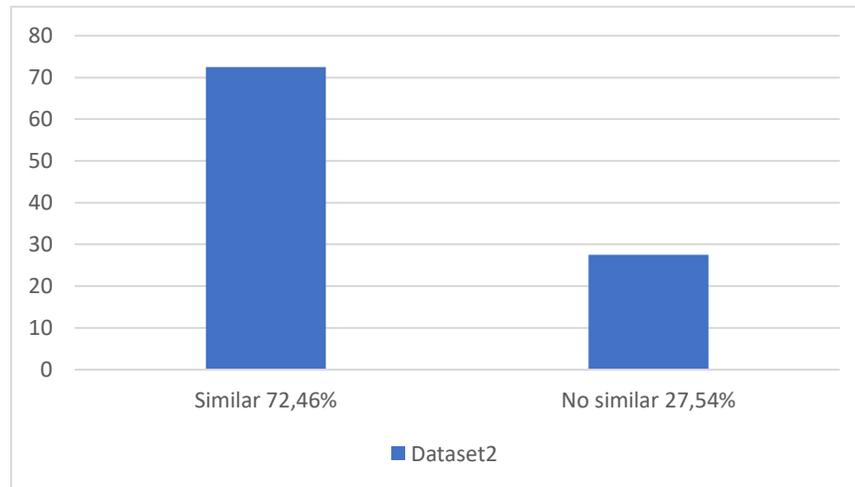
id_no	id_bot	Similitud
245356	195353	1
109806	87006	0
97756	258085	1
172458	271761	0

Nota: *Id_bot* = Id dado por el user-bot
Similitud = Si *id_posible* es similar a *id_bot* = 1 caso contrario = 0.

Fuente: Elaboración propia

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Gráfico 13 Valores Dataset2



Nota:

Fuente: Elaboración propia

Autor: Sánchez Piguave Alexander, Almache Carranza Carlos

Como se puede apreciar en el Gráfico 13 refleja el resultado del análisis obtenido entre la comparación de las columnas *id_no* y *id_bot* campos que se pueden visualizar en el gráfico 12.

Para este análisis solo se tomó en cuenta las preguntas que dieron como resultado 1 en el *dataset1*.

Discusión

De acuerdo con los resultados obtenidos se puede concluir que el USER-BOT desarrollado y los algoritmos de machine learning implementados en este funcionan de manera correcta ofreciendo al usuario respuestas adecuadas a sus necesidades en el menor tiempo posible y con la mejor calidad que puede esperarse. Para ir mejorando los resultados del *bot* se deben ejecutar y afinar los procesos de machine learning en el transcurso del tiempo, así como mantener el repositorio de

datos actualizado con la última información de la comunidad virtual. Así también, se plantea que para trabajos futuros se pueda ampliar el alcance de los orígenes de datos y que, además, la interfaz con el usuario se amplíe a otras plataformas de mensajería que soporten la configuración y uso de *bots*.

Referencias

- Abreu, J. (2012). Hipótesis, Método y diseño de investigación. *International Journal of Good Conscience*, 187-197.
- Almache, C., & Sánchez, A. (01 de 10 de 2020). *zenodo*.
doi:<http://doi.org/10.5281/zenodo.4075675>
- Azzola, F. (2014). *Android UI Design*. Exelixis Media Ltd.
- Báez, M., Borrego, Á., Cordero, J., Cruz, L., González, M., Hernández, F., . . . Álvaro, Z. (2016). *Introducción a Android*. Madrid: Universidad Complutense de Madrid.
- Barrientos, E., & Mamani, S. (2019). *Modelos de aprendizaje supervisado como apoyo a la toma de decisiones en las organizaciones basados en datos de redes sociales: Una revisión sistemática de la literatura*. Lima.
- Bohórquez-Muñoz, J. (2018). *Algoritmo de ubicación de fallas de un terminal basado en modelos de machine learning*.
- Bosch, J., & Bosch-Sijtsema. (2010). From integration to composition: On the impact of software product lines, global development and ecosystems. *ScienceDirect*.
- Botto, M. (2014). SOA2Cloud: Un marco de trabajo para la migración de aplicaciones SOA a Cloud siguiendo una aproximación dirigida por modelos. 176.
- Calibar, A., J, H., & RO, K. (2018). Análisis de similitud en documentos de texto mediante técnicas de ciencia de datos basadas en aprendizaje profundo (deep learning).
- Carod, M., & Aranda, C. (2017). *Búsqueda de estrategias para la clasificación del contenido en foros técnicos de discusión*.
- Castaño, Y. (2016). Evaluación de ecosistemas de desarrollo de software desde la perspectiva de cobit. *IV Taller Internacional Las TIC en la Gestión de las Organizaciones*. La Habana. Obtenido de https://d1wqtxts1xzle7.cloudfront.net/57613192/EVALUACION_DE_ECOSISTEMAS_DE_DESARROLLO_DE_SOFTWARE_DESDE_LA_PERSPECTIVA_DE_COBIT.pdf?1540233793=&response-content-disposition=inline%3B+filename%3DEVALUACION_DE_ECOSISTEMAS_DE_DESARROLLO.pdf&Expires=16017445
- Celdrán, M., Mazón, J., Ivars, J., & Vera, J. (2018). SMART TOURISM. UN ESTUDIO DE MAPEO SISTEMÁTICO. 32.
- Donoso, Í., & Vega, V. (2017). Factores sociales y humanos que afectan el proceso de educación de requerimientos: una revisión sistemática. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, 69-83.
- Flórez, H. (2012). *Programación orientada a objetos usando Java*. Bogotá: ECOE Ediciones.
- G., D. (1 de 11 de 2016). *My Python*. Obtenido de <http://www.pythondiario.com/2016/11/los-5-mejores-ide-para-python.html>
- Gallardo, E. (2019). *Creación de un modelo predictivo para clasificar las consultas ciudadanas sobre el transporte público*. Valencia.
- García, A. (2013). *Análisis de integración de soluciones basadas en software como servicio para la implantación de ecosistemas tecnológicos corporativos*. Salamanca.
- Gil, C. (2015). *Los web services y características de calidad*.
- González, R. (2016). *Python para todos*.
- González, R., Pérez, I., García, R., Peñaherrera, E., & Piñero, P. (2016). *Ecosistema de Software GESPRO-16.05 para la Gestión de Proyectos*. La Habana: Scielo.

- Gramajo, M., Ballejos, L., & Ale, M. (2018). Análisis de las técnicas de aprendizaje automático aplicadas en la ingeniería de requerimientos de software: un mapeo sistemático de la literatura.
- Gras, V. (2020). Detección de palabras cortadas al inicio y fin de línea en textos manuscritos.
- Katiyar, S., Kumar, S., & Walia, H. (2020). Predicción de personalidad a partir de Stack Overflow mediante el uso del teorema de Naïve Bayes en minería de datos. 5.
- Leyva, M., & Smarandache, F. (2018). Inteligencia Artificial: retos, perspectivas y papel de la neutrosfía.
- M Danilo, T. L. (2019). Un enfoque de Machine Learning en el desarrollo de Sistema Recomendadores para Procesos de Investigación.
- Manterola, C., & Otzen, T. (2015). Estudios Experimentales 2 Parte. Estudios Cuasi-Experimentales. *International Journal of Morphology*.
- Martech. (07 de 01 de 2020). *Martech Forum*. Obtenido de <https://www.martechforum.com/articulo/las-mejores-comunidades-para-programadores-web/>
- Martínez, N., Zoratto, V., Aranda, G., Cechich, A., Chiarotto, A., Noda, C., & Sagripanti, M. (2018). *Análisis de calidad y recuperación de información en foros de discusión*. Buenos Aires.
- Mitra, G. (2016). *Android user interfaz*.
- MongoDB. (01 de 02 de 2020). *MongoDB*. Obtenido de <https://www.mongodb.com/es>
- Mysql. (2014). *MySQL 5.0 Reference Manual*.
- Paramio, C. (01 de 02 de 2020). *Genbeta*. Obtenido de <https://www.genbeta.com/desarrollo/una-introduccion-a-mongodb>
- Parejo, J., Fernández, P., & Ruíz-Cortés, A. (2010). *De frameworks a Ecosistemas: Evolución del software para optimización metaheurística*. Sevilla: Universidad de Sevilla.
- Pereira-Toledo, A., López-Cabrera, J., & Quintero-Domínguez, L. (2017). Estudio experimental para la comparación del desempeño de Naïve Bayes con otros clasificadores bayesianos. *Revista Cubana de Ciencias Informáticas*.
- Piñero, P., & Pérez, I. (2015). *Modelo de desarrollo de software basados en líneas de productos de software y ecosistemas de software. Reporte de Investigación. Laboratorio de investigación en Gestión de Proyectos*. La Habana: Universidad de las Ciencias Informáticas.
- Priego, B., Anaya, S., & Reyes, J. (2018). Análisis del comportamiento de diferentes. 13.
- Recena, M., Muñiz, M., Gómez, A., & Beas, J. (2012). *Software Development Ecosystem*. Sevilla, España.
- Recena, S. (2012). Ecosistemas de desarrollo software. Líneas de automatización. *Escuela Técnica Superior de Ingeniería Informática*, 20-30.
- Russo, C., Ramón, H., Alonso, N., Cicerchía, B., Esnaola, L., & Tessore, J. (2018). Tratamiento Masivo de Datos Utilizando Técnicas de Machine Learning. *Instituto de investigación y transferencia en tecnología*.
- Sandoval-Serrano, L. (2018). *Algoritmos de aprendizaje automático para análisis y predicción de datos*.
- Siles, I. (2005). *Comunidades en línea historia, comunicación y tecnología en la emergencia de colectivos mediáticos*. San José - Costa Rica: Revista de Ciencias Sociales .
- Sommerville, I. (2015). *Ingeniería del Software*. Madrid: PEARSON EDUCACIÓN. S.A.
- Sommerville, I. (2015). *Ingeniería del Software*. Madrid: PEARSON EDUCACIÓN. S.A.

- Sosa, R., Pérez, I., García, R., Peñaherrera, E., & Piñero, P. (2016). Ecosistema de Software GESPRO-16.05 para la Gestión de Proyectos. *Revista cubana de ciencias informáticas*.
- Stalling, W. (2005). *Sistemas operativos. Aspectos internos y principios de diseño*. Madrid: Pearson.
- Talavera, A., & Luna, A. (2020). Aprendizaje Automático: Una Contribución a la Investigación Operativa. 7.
- Velásquez, F., De-Paz, J., & Castillo, M. (2017). Aplicación del análisis sintáctico automático en la atribución de autoría de mensajes en redes sociales.
- Wolf, G., Ruiz, E., Bergero, F., & Meza, E. (2015). *Fundamentos del sistemas operativos*. Coyoacán.
- Zoratto, V., Aranda, G., & N, M. (2019). Análisis de estrategias para clasificación de usuarios y post dentro de un hilo de discusión.