



“Semillero de Investigadores”

ISSN: 2007-9400

“Tecnología para la sociedad”

Índice LatIndex: Folio 23658

www.revistatecnologiadigital.com

Desde 2011

Volumen Número

11 2

Año 2021

REVISTA TECNOLOGÍA DIGITAL

CONSEJO EDITORIAL.

Presidente. Dr. Héctor Guerra Crespo.

Instituto Tecnológico de Tuxtla Gutiérrez.
hgcrespo@hotmail.com, hector.gc@tuxtla.tecnm.mx

Secretario. M.C. Walter Torres Robledo.

Instituto Tecnológico de Tuxtla Gutiérrez.
walter.tr@tuxtla.tecnm.mx

Editor M.C. Imelda Valles López.

Instituto Tecnológico de Tuxtla Gutiérrez.
imelda.vl@tuxtla.tecnm.mx

Editor traductor. Dr. Daniel Samayoa Penagos.

Instituto Tecnológico de Tuxtla Gutiérrez.
daniel.sp@tuxtla.tecnm.mx

Editor traductor. Dra. Magdalena Guerra Crespo.

Instituto de Biotecnología de la UNAM, Cuernavaca Morelos.
magdagmx@hotmail.com

Editor traductor. Dra. María Candelaria Gutiérrez Gómez.

Instituto Tecnológico de Tuxtla Gutiérrez.
maria.gg1@tuxtla.tecnm.mx

Revista Tecnología Digital.

Índice LatIndex: 23658.

ISSN: 2007-9400.

Instituto Tecnológico de Tuxtla Gutiérrez.
Carretera Panamericana Km. 1080.

C.P. 29000, Tuxtla Gutiérrez, Chiapas, México.

Departamento de Sistemas y Computación.

01-961-6150461 Ext. 319

www.revistatecnologiadigital.com

Editor Responsable.

Editorial Tecnología Didáctica®.

Dr. Héctor Guerra Crespo.

Registro del padrón de editores: 970-94054 de
fecha 22-junio-2004.

Palenque 139 Col. ISSSTE, Tuxtla Gutiérrez,
Chiapas. C.p. 29060, Tuxtla Gutiérrez, Chiapas,
México.

www.tecnologiadidactica.com

Revista Tecnología Digital. Año 11, Volumen 11, Número 2 de 22 de diciembre de 2021 es una publicación anual on-line www.revistatecnologiadigital.com editada por Héctor Guerra Crespo oficinas en Palenque 139 Col. ISSSTE Tuxtla Gutiérrez Chiapas, México C.P. 29060. correo-e hgcrespo@hotmail.com. Editor responsable: Héctor Guerra Crespo. Reserva de derechos al uso exclusivo No. 04-2014-022811040300-203 otorgado por el Instituto Nacional de Derechos de Autor. ISSN 2007-9400. Responsable de la última actualización de este número Dr. Héctor Guerra Crespo, Presidente del Consejo Editorial, Palenque 139 Colonia ISSSTE Tuxtla Gutiérrez, Chiapas, C.P. 29060, fecha de la última modificación, 22 de diciembre de 2020.

Las opiniones expresadas por los autores no necesariamente reflejan la postura del editor de la publicación. Queda prohibida la reproducción total o parcial de los contenidos e imágenes de la publicación sin previa autorización del editor responsable. Índice LatIndex: 23658.

Índice.

	Página.
Algoritmo de minería de datos para la clasificación de violencia en pareja.	 1
Análisis de un bot para la extracción de información en interacciones farmacológicas.	 11
Animatronic con forma de araña hexápoda.	 21
Características morfológicas para el análisis automático del estilo de escritura.	 31
Diseño de prototipo de direccionamiento de robot tipo Ackerman aplicado a la desinfección de COVID.	 47
Diseño y construcción de un prototipo dispensador automático de alimentos para ganado porcino.	 59
Sistema de ubicación y reconocimiento de objetos para robot autónomo.	 69
Sistema inteligente para tutores del bachillerato tecnológico del Estado de Morelos, México.	 79



Algoritmo de minería de datos para la clasificación de violencia en pareja.

Data Mining Algorithm for Classification of Intimate Partner Violence.

Miriam Vargas Reyes (1).
Tecnológico Nacional de México/TES Valle de Bravo, Estado de México, Valle de Bravo.
L201607039@vbravo.tecnm.mx.

Mariana Carolyn Cruz Mendoza* (2). Tecnológico Nacional de México/TES Valle de Bravo, Estado de México, Valle de Bravo. mariana.cm@vbravo.tecnm.mx.

Cesar Primero Huerta (3). Tecnológico Nacional de México/TES Valle de Bravo, Estado de México, Valle de Bravo.
cesar.ph@vbravo.tecnm.mx.

*corresponding author.

Artículo recibido en diciembre 18, 2020; aceptado en febrero 18, 2021.

Resumen.

Desde hace muchos años, la violencia, en todas sus clasificaciones, ha sido considerada como parte de nuestra cultura, y de esta forma, se acepta como un integrante en la formación de la sociedad, donde se daña, principalmente, la integridad de la mujer, causando daños, no solo físicos, sino también psicológicos, por lo tanto, es necesario atender el problema, para que de algún modo se pueda frenar o minimizar. La presente investigación aborda la violencia en pareja, donde aplicando instrumentos de recolección de datos se obtendrá información de diferentes participantes mujeres que sufren o han sufrido algún tipo de violencia, teniendo como propósito principal implementar algoritmos de minería de datos para clasificarla en 4 grupos; violencia física, psicológica, sexual y verbal y con ello generar planes de acción recomendados por un especialista.

Palabras clave: Violencia de pareja, algoritmos de clasificación, minería de datos, planes de acción.

Abstract.

For many years, violence, in all its classifications, has been considered as part of our culture, and in this way, it is accepted as an integral part of the formation of society, where the integrity of women is mainly damaged, causing damage, not only physical, but also psychological, therefore, it is necessary to address the problem, so that it can somehow be stopped or minimized. This research addresses intimate partner violence, where, by applying data collection instruments, information will be obtained from different female participants who suffer or have suffered some type of violence, with the main purpose of implementing data mining algorithms to classify it into 4 groups; physical, psychological, sexual and verbal violence and thereby generate action plans recommended by a specialist.

Keywords: Intimate partner violence, classification algorithms, data mining, action plans

1. Introducción.

La violencia ha existido desde tiempos muy remotos hasta llegarse a considerar como parte de nuestra cultura. Los factores que la caracterizan, como el comportamiento agresivo, ha estado presente a través de todos los tiempos, y ha quedado plasmada en documentos que van desde las antiguas escrituras hasta las estadísticas actuales. Gran parte de estas cifras se centran en la violencia de género, ya que, organismos internacionales como ONUMUJERES, estima que el 35 % de las mujeres de todo el mundo ha sufrido algún tipo de violencia ya sea física y/o sexual, por parte de una persona con lazos sentimentales. Es por esto, que es necesario atender el problema, para que de algún modo se pueda frenar o minimizar buscando soluciones con ayuda de las tecnologías de la información. El siguiente documento presenta una investigación acerca de la violencia en pareja, la información será utilizada para implementar algoritmos de clasificación como J48 o árboles de decisión y Bayes Net haciendo uso de minería de datos ya que ésta ayuda a analizar grandes volúmenes de información, para clasificar, agrupar o crear modelos predictivos.

Dichos datos serán recolectados mediante una encuesta, que incluye datos personales y características matrimoniales o de relación de las participantes. La clasificación que se desea obtener es verbal, sexual, psicológica y física para recomendar planes de acción a las víctimas de acuerdo con la violencia que están viviendo, dichos planes serán recomendados por un especialista.

En la actualidad existen diversas aplicaciones web, donde las víctimas de violencia pueden ingresar de manera anónima y contestar encuestas, como, por ejemplo, “Encuesta sobre violencia contra la mujer” por parte del Instituto Mexicano de Seguro Social, con el propósito de conocer si la víctima sufre de violencia, estos datos son enviados y posteriormente colocan enlaces donde se puede pedir ayuda o simplemente se envían los datos y no se tiene ninguna otra función. Por otra parte, se han desarrollado diversas investigaciones dentro del marco de minería de datos, por ejemplo, el proyecto “Aplicación de minería de datos para determinar las causas de la violencia de género contra las mujeres en Ecuador” donde se analiza información para determinar cuáles son las causas principales de la violencia hacia la mujer, se enfoca en la violencia de pareja, noviazgo, en la niñez y en el ámbito social, educativo y laboral.

2. Métodos.

Metodología.

KDD (Knowledge discovery in databases): Es una de las metodologías dominantes para el proceso de la minería de datos.

Fases

- Selección o búsqueda de información: Investigaciones que proporcionen información sobre el problema, lineamientos para la obtención de bases de datos
- Preprocesamiento: obtención de datos, cumplimiento de lineamientos establecidos
- Transformación: depuración de base de datos, selección de información concreta para la resolución del problema
- Minería de datos: Aplicación de algoritmos de clasificación orientados a cada caso
- Evaluación e implantación: interpretación de los datos: presentación de resultados simplificados y comprensibles.

Algoritmos de clasificación.

- Redes Bayesianas: Son una representación gráfica donde se modelan dependencias para el razonamiento probabilístico, consiste en una red que contiene nodos que representan las variables aleatorias y los arcos que representan las relaciones de dependencia directa entre dichas variables.
- Árboles de clasificación, algoritmo J48 (Clasificador C4.5): El algoritmo J48 de WEKA es una implementación del algoritmo C4.5, algoritmo de minería de datos más utilizado. (Sierra, s.f.).
- El algoritmo C4.5 permite generar un árbol de decisión a partir de datos obtenidos.

- Árboles de decisión: son una representación de los procesos que se involucran en las tareas de clasificación.

Software de entrenamiento Weka.

WEKA (Waikato Environment for Knowledge Analysis) es un software gratuito de aprendizaje automático y minería de datos distribuido bajo licencia GNU-GPL. Tiene una gran colección de algoritmos para tareas de minería de datos y modelado predictivo. También incluye herramientas para el preprocesamiento de los datos (filtros), clasificación (árboles, tablas), clustering, reglas de asociación, y adicionalmente, diversas formas de visualización de los datos, tanto en el inicio del proceso de carga de datos, como después de haber aplicado un algoritmo (Pineda, 2017).

Muestra.

Para poder armar el banco de datos es necesario tener al menos 100 datos, por lo tanto, fue necesario elegir tres comunidades; Loma de Rodríguez, La Rinconada y El castellano, pertenecientes al municipio de Valle de Bravo, Estado de México. Teniendo un total de 142 parejas en las tres comunidades.

Se utilizó el software SURVEYMONKEY, donde se ingresa el tamaño de la población, que sería el total de parejas, con un nivel de confianza del 95% y con un margen de error del 5%, dando un total de 104 mujeres para encuestar.

Recolección de datos.

El siguiente instrumento de recolección de datos se creó con ayuda de una encuesta validada del autor Javier Gustavo Fernández Teruelo, la cual recolecta información de violencia que sufre la víctima, en situación de pareja o expareja, para ayudar a predecir y prevenir feminicidios. Así mismo se utilizaron algunas características del instrumento que creó la doctora Martha Alicia Tronco Rosas directora de la Unidad Politécnica de Gestión con Perspectiva de Género del Instituto Politécnico Nacional, denominado “Violentometro”. Por otro lado, para obtener los datos personales se acudió a un trabajo de investigación denominado Escala de Evaluación del Tipo y Fase de Violencia de Género (EETFVG). Así mismo se pidió ayuda para la revisión de la encuesta, a un científico calificado.

Proyecto “Algoritmo de minería de datos para la clasificación de violencia en pareja”
Encuesta

Objetivo: Obtener información de mujeres que viven en pareja de las comunidades de Loma de Rodríguez, La Rinconada y El Castellano para conocer si sufren violencia y con ello aplicar técnicas de minería de datos para clasificarla en física, verbal, sexual y psicológica.

Instrucciones. Indique de forma voluntaria sus características de:

A. Datos Personales		
Edad:		
Estado civil:	☐ Casado	☐ Unión libre
Situación económica:	☐ Clase alta	☐ Clase media ☐ Clase baja
Trabaja:	☐ Si	☐ No
Número de hijos:		
Lugar de nacimiento:		
Lugar de residencia:		
B. Religión		
☐ Católica	☐ Bíblicas diferentes a evangélicas	
☐ Protestantes y evangélicas	☐ Otro	
C. Formación integral de la salud		
¿Cómo considera su estado de salud?		
☐ Malo	☐ Regular	
☐ Bueno	☐ Excelente	
¿Tiene alguna dependencia sobre las siguientes sustancias?		
☐ Alcohol	☐ Tabaco	☐ Sustancias ilegales
D. Nivel de educación y analfabetismo		
☐ No sabe leer ni escribir	☐ Lee y escribe	☐ Primaria
☐ Secundaria	☐ Preparatoria	☐ Estudios universitarios
E. Factores de vulnerabilidad de la víctima		
Discapacidad	☐ Si	☐ No
En periodo de gestación	☐ Si	☐ No
Víctima extranjera	☐ Si	☐ No

Figura 1. Encuesta. Datos personales. Elaboración propia

F. Situación matrimonial o de relación

¿Ha existido algún tipo de violencia por parte del agresor?	☐ Si	☐ No
¿Ha existido reacción defensiva de la víctima ante la agresión?	☐ Si	☐ No
¿Su pareja ha empleado armas u objetos contra usted?	☐ Si	☐ No
Su pareja empleó:	☐ Arma blanca	☐ Arma de fuego
¿Su pareja tiene acceso a armas de fuego?	☐ Si	☐ No
Señale marcando un Si o un No si su pareja le ha hecho alguna de las siguientes acciones:		
Su pareja la ha acosado	☐ Si	☐ No
Su pareja ha intentado aislarla de su familia y/o amigos	☐ Si	☐ No
Su pareja le ha hecho bromas hirientes	☐ Si	☐ No
Su pareja la ha chantajeado	☐ Si	☐ No
Su pareja intenta controlarla	☐ Si	☐ No
Su pareja la culpabiliza	☐ Si	☐ No
Su pareja la ha descalificado	☐ Si	☐ No
Su pareja ha disminuido su persona	☐ Si	☐ No
Su pareja no muestra interés o tiene falta de atención hacia usted	☐ Si	☐ No
Su pareja constantemente le grita	☐ Si	☐ No
Su pareja le dice groserías	☐ Si	☐ No
Su pareja la ha humillado	☐ Si	☐ No
Su pareja la ignora	☐ Si	☐ No
Su pareja la insulta	☐ Si	☐ No
Se siente intimidada por su pareja	☐ Si	☐ No
Su pareja le ha dejado de hablar o le ha aplicado la ley de hielo	☐ Si	☐ No
Su pareja la ha maldecido	☐ Si	☐ No
Su pareja la ha manipulado	☐ Si	☐ No

Figura 2. Fragmento de encuesta datos matrimoniales. Elaboración propia**3. Desarrollo.****Análisis de datos.**

Una vez que se diseñó el instrumento de recolección de datos se localizaron 86 variables para el entrenamiento de los datos, las cuales se describen a continuación.

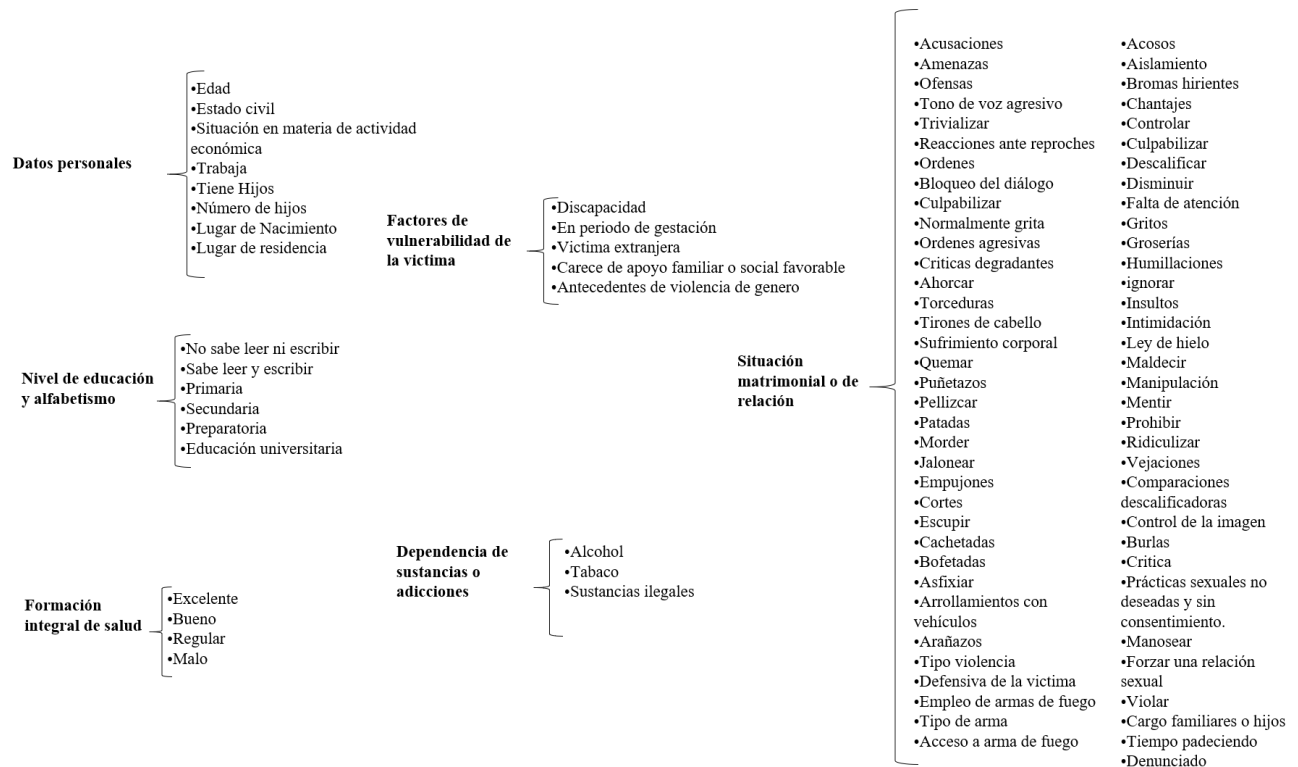


Figura 3. Variables localizadas

Aplicación de la encuesta y creación del dataset.

Las entrevistas se aplicaron de manera presencial, debido a que las comunidades seleccionadas son rurales, mucha de las mujeres no tiene acceso a internet o teléfonos celulares, para que se haya aplicado de manera digital. En un análisis previo las encuestas contestadas arrojaron que la edad de las participantes se repite más de los 29 a 50 años y la mayoría están casadas por el civil y por la iglesia, todas las mujeres practican la religión católica y la mitad trabajan, la mayoría de las mujeres arriba de 50 años no lee ni escribe y muy pocas saben escribir y leer, así mismo debajo de la edad de 49 años sabe leer y escribir, asistieron a la educación básica y superior y muy pocas mujeres tienen estudios de nivel medio superior y superior. Su estado de salud se encuentra en el nivel “Bueno”, no tienen discapacidad y son originarias de Valle de Bravo, así mismo solo tres o cuatro mujeres están en periodo de gestación. Los resultados obtenidos se colocaron en una hoja de cálculo de Excel para crear el dataset para el entrenamiento de los datos, a continuación, se muestra un fragmento.

edad	estado civil	situacion_economica	trabaja	hijos	numero_hijos	lugar_nacimiento	lugar_residencia	religion	formacion_integral_salud	adiciones	grado_estudios	discapacidad	periodo_gestacion	victima_extranjera	carece_apoyo	antecedentes_violencia	tipo_violencia_agresor	defensiva_victima
42	Casado	Clase media	No	Si	3	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Primaria	No	No	No	No	No	No	No
30	Casado	Clase media	Si	Si	2	Cerro Colora	Loma de Rod	Catolica	Bueno	Ninguna	Estudios univ	No	No	No	No	No	No	No
60	Casado	Clase baja	No	Si	18	La Rinconada	Loma de Rod	Catolica	Regular	Ninguna	No sabe leer	No	No	No	No	No	No	No
30	Casado	Clase media	Si	Si	2	La Mesa	Loma de Rod	Catolica	Bueno	Ninguna	Primaria	No	No	No	No	No	Si	No
21	Union libre	Clase media	No	Si	1	El Castellano	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
47	Casado	Clase media	Si	Si	3	Rincon de Es	Loma de Rod	Catolica	Bueno	Ninguna	Lee y escribe	No	No	No	No	No	Si	No
48	Casado	Clase media	No	Si	5	Loma de Chil	Loma de Rod	Catolica	Bueno	Ninguna	No sabe leer	No	No	No	No	No	No	No
41	Casado	Clase media	Si	Si	5	Rincon de Es	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	Si	No	No	No
45	Union libre	Clase media	Si	Si	4	Rincon de Es	Loma de Rod	Catolica	Bueno	Ninguna	Primaria	No	No	No	No	No	Si	No
34	Casado	Clase media	Si	Si	2	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
56	Union libre	Clase baja	Si	Si	6	Loma de Rod	Loma de Rod	Catolica	Regular	Ninguna	Lee y escribe	No	No	No	Si	No	Si	Si
49	Casado	Clase media	Si	Si	5	San Mateo	Loma de Rod	Catolica	Regular	Ninguna	Lee y escribe	No	No	No	No	No	Si	Si
28	Union libre	Clase media	Si	Si	2	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	Si	No	No	No	No	No
65	Casado	Clase baja	No	Si	2	El Fresno	Loma de Rodriguez	Regular	Alcohol	No sabe leer	No	No	No	No	Si	No	Si	No
30	Union libre	Clase media	Si	Si	2	Loma de Chil	Loma de Rod	Catolica	Regular	Ninguna	Secundaria	No	No	No	No	No	Si	No
27	Union libre	Clase media	Si	Si	1	Valle de Brav	Loma de Rod	Catolica	Bueno	Ninguna	Estudios univ	No	No	No	No	No	No	No
33	Union libre	Clase media	Si	Si	3	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	Si	Si
21	Union libre	Clase media	No	Si	2	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Estudios univ	No	No	No	No	No	No	No
49	Union libre	Clase media	Si	Si	3	Avandaro	Loma de Rod	Catolica	Regular	Ninguna	No sabe leer	No	No	No	No	No	No	No
24	Union libre	Clase media	Si	No	0	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
25	Union libre	Clase media	Si	Si	1	Polvillos Am	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
27	Union libre	Clase media	Si	Si	1	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
40	Casado	Clase baja	No	Si	2	Loma de Chil	Loma de Rod	Catolica	Bueno	Ninguna	Lee y escribe	No	No	No	No	No	No	No
70	Casado	Clase media	No	Si	9	Los Alamos	Loma de Rod	Catolica	Bueno	Ninguna	No sabe leer	No	No	No	No	No	Si	No
36	Union libre	Clase media	Si	Si	3	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
34	Casado	Clase media	Si	Si	2	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
30	Union libre	Clase media	Si	No	0	Rincon de Es	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	Si	No
40	Casado	Clase media	Si	Si	3	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
32	Union libre	Clase media	Si	Si	2	Barrio de Gu	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
89	Casado	Clase media	No	Si	8	El Castellano	Loma de Rod	Catolica	Regular	Ninguna	Lee y escribe	No	No	No	No	No	Si	No
46	Casado	Clase media	Si	Si	3	Los Alamos	Loma de Rod	Catolica	Bueno	Ninguna	Lee y escribe	No	No	No	No	No	No	No
65	Casado	Clase media	No	Si	5	San Pedro	Loma de Rod	Catolica	Regular	Tabaco	Lee y escribe	No	No	No	Si	No	Si	Si
52	Casado	Clase media	No	Si	13	El Castellano	Loma de Rod	Catolica	Regular	Ninguna	No sabe leer	No	No	No	Si	No	Si	No
53	Casado	Clase media	No	Si	5	La Rinconada	Loma de Rod	Catolica	Regular	Ninguna	Lee y escribe	No	No	No	No	No	Si	No
26	Casado	Clase media	Si	Si	2	Cerro Colora	Loma de Rod	Catolica	Bueno	Ninguna	Preparatoria	No	No	No	No	No	Si	Si
48	Union libre	Clase media	No	Si	6	San Juan Ate	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
56	Casado	Clase media	No	No	0	Loma de Rod	Loma de Rod	Catolica	Regular	Ninguna	Primaria	No	No	No	Si	No	No	No
49	Casado	Clase media	Si	Si	6	Loma de Rod	Loma de Rod	Catolica	Regular	Ninguna	Secundaria	No	No	No	Si	No	No	No
55	Casado	Clase media	No	Si	5	Rincon de Es	Loma de Rod	Catolica	Bueno	Ninguna	Lee y escribe	No	No	No	No	No	No	No
32	Casado	Clase media	Si	Si	1	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
68	Casado	Clase media	No	Si	4	San Mateo	Loma de Rod	Catolica	Regular	Ninguna	Lee y escribe	No	No	No	No	No	Si	No
50	Casado	Clase media	No	Si	4	San Mateo	Loma de Rod	Catolica	Bueno	Ninguna	Estudios univ	No	No	No	No	No	No	No
28	Union libre	Clase media	Si	Si	3	San Mateo	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
38	Union libre	Clase media	Si	Si	3	San Mateo	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
28	Union libre	Clase media	Si	Si	1	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	No	No
34	Casado	Clase baja	No	Si	6	Loma de Rod	Loma de Rod	Catolica	Bueno	Ninguna	Lee y escribe	No	No	No	Si	Si	Si	Si
26	Union libre	Clase media	No	Si	1	Rincon de Es	Loma de Rod	Catolica	Bueno	Ninguna	Secundaria	No	No	No	No	No	Si	No
32	Union libre	Clase media	No	Si	2	San Juan Ate	Loma de Rod	Catolica	Bueno	Ninguna	Estudios univ	No	No	No	No	No	No	No
51	Casado	Clase media	No	Si	5	Ciudad de M	Loma de Rod	Catolica	Regular	Ninguna	Primaria	No	No	No	No	No	No	No

Figura 4. Fragmento de dataset. Elaboración propia.

Implementación del algoritmo J48.

Los archivos contenidos en la hoja de cálculo Excel, serán guardados en archivo CSV, para que el software de entrenamiento de datos pueda leerlos. La implementación del algoritmo J48 genera arboles de clasificación para la toma de decisiones, utiliza únicamente variables de tipo cualitativo. Para poder utilizarlo es necesario abrir al archivo de tipo CSV con los datos que se desean entrenar mediante este algoritmo. En la parte superior izquierda localizar CLASSIFY y dar clic.

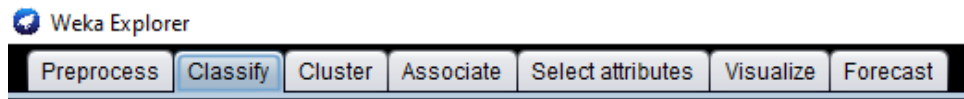


Figura 5. Menú. Software Weka.

Localizar el apartado de TREES y elegir el algoritmo J48

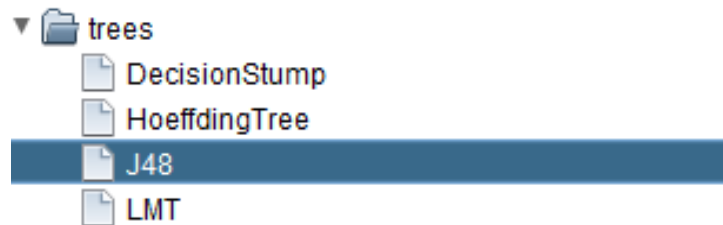


Figura 6. Seleccionar algoritmo J48. Software Weka.

Este algoritmo se aplica a cada una de las variables que se localizaron, generando arboles para la toma de decisiones. A continuación, se presentan algunos ejemplos y su análisis.

Variable estado_civil: Este árbol tiene un tamaño de 11, y se muestra la relación entre la variable estado_civil con las variables periodo_gestacion, edad y acceso_armas. Las mujeres que están en periodo de gestación están en unión libre, las mujeres que no están en periodo de gestación tienen una edad mayor a 39 años y están casadas, y las mujeres menores o igual a 39 años y que no tienen acceso a arma están en unión libre, así mismo las parejas de las mujeres que tienen acceso a arma y son menores de o igual a 39 años están casadas.

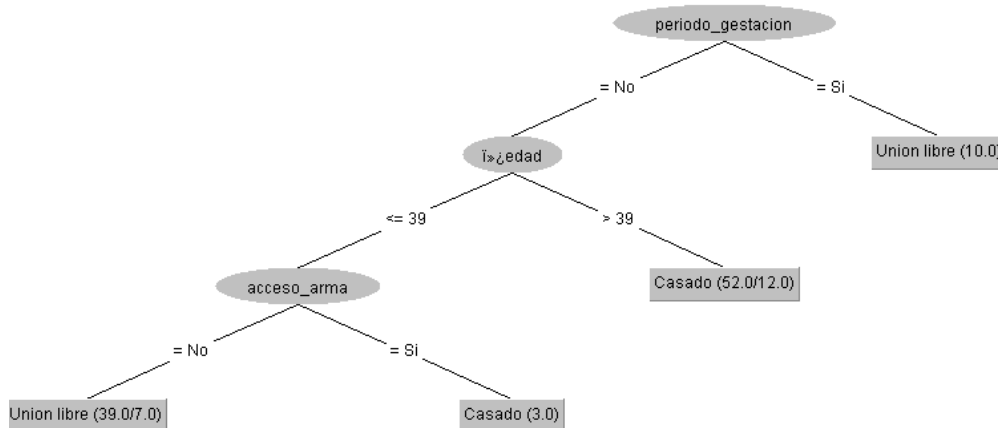


Figura 7. Árbol J48. Variable estado_civil

Variable situación_economica: El tamaño de este árbol es de 4 y relaciona las variables de situación_economica y adicciones.

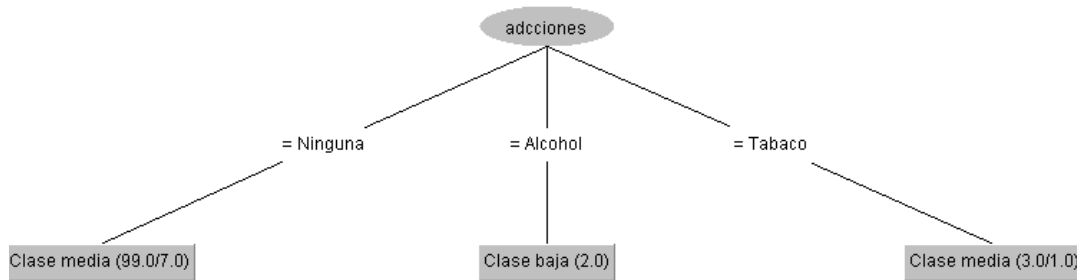


Figura 8. Árbol J48. Variable situacion_economica

Resultados obtenidos del análisis de los árboles de decisión.

Al hacer el análisis de todos los árboles generados, se encuentra que algunas variables no tienen relación entre si y se obtienen dos grupos. El primer grupo con características de violencia física y el segundo con características de violencia verbal, sexual y psicológica.

Ahorcar, arañazos, tirones de cabello, patadas, jalonear, sufrimiento_corporal, trabaja, carece_apoyo, maldecir, puñetazos, sufrimiento_corporal, pellizcar, patadas, tirones_de_cabello, sufrimiento_corporal, morder, jalonear, empujones, cortes, escupir, cachetadas, bofetadas, asfixiar, arrollamineto_con_vehiculos, arañazos.

Figura 9. Primer grupo obtenido.

estado_civil, lugar_de_nacimiento, situación_economica, grado_estudios, carece_apoyo, trabaja, comparaciones_descalificadoras, tiempo_padeciendo, formacion_integral_salud, edad, antecedentes_violencia, culpabilizar, intimidación, ley_hielo, manipulación, periodo_gestacion, acusaciones, tipo_violencia_agresor, acceso_arma, ignorar, acoso, armas_contra_la_victima, tipo_arma, asfixiar, defensiva_victima, bromas_hirientes, falta_atencion, chantajear, maldecir, insultos, descalificar, humillaciones, gritos, groserías, insultos, control_imagen, disminuir, maldecir, tono_de_voz_agresivo, pellizcar, tiempo_padeciendo, asifixiar, mentir, burlas, trivilizar, prohibir, control_imagen, controlar, criticas, ridiculizar, vejaciones, comparaciones_descalificadoras, practicas_sexuales_no_deseadas, violar, forzar_relacion, manosear, amenazas, reacciones_ante_reproches, ofender, trivializar, criticas_degradantes, vejaciones, bloqueo_del_dialogo, normalmente_grita, ordenes_agresivas,

Figura 10. Segundo grupo obtenido.

Conclusiones.

La aplicación de nuevas tecnologías a problemas que son bastante graves ha ido aumentando de manera considerable, la minería de datos ha ayudado a analizar grandes volúmenes de información para crear modelos de clasificación, agrupamiento o modelos predictivos.

La elaboración y aplicación del instrumento de recolección de datos o encuesta es bastante difícil ya que se deben buscar encuestas que ya están validadas o crear una con ayuda de un especialista, en el momento de aplicar la encuesta se deben de tomar en cuenta varios aspectos para que las participantes no se sientan intimidadas o con miedo, por ejemplo proporcionar datos personales, debido a la seguridad, incomodidad debido al tema, las emociones que pueden surgir debido a las preguntas y no tener conocimiento para manejar la situación, por lo que es necesario pedir ayuda de un psicólogo para que la situación esté un poco más controlada. Weka es un software con una interfaz sencilla de entender y se pueden elegir los algoritmos para aplicar de una manera rápida. Se localizaron 86 variables de la encuesta que se aplicó, se acomodaron en la hoja de cálculo de Excel con las respuestas que se obtuvieron de las participantes y se exportó el archivo a formato CSV, que es el formato que admite Weka. Con esto se pudo observar cómo se visualizan cada una de las variables y el porcentaje de respuestas que tiene cada una. El algoritmo J48 crea árboles para la toma de decisiones donde se puede observar la relación entre las variables y se pueden llegar a agrupar. Algunas no presentan relación y pueden ser descartadas.

Referencias bibliográficas.

- Ayala Salgado, L. y. (1 de Mayo de 2012).** *Contribuciones a las Ciencias Sociales*. Obtenido de "La violencia hacia la mujer. Antecedentes y aspectos teóricos": <http://www.eumed.net/rev/cccss/20/ashm.html>
- Campos, A. (Mayo de 2003).** *La violencia Familiar (Tesis de grado. "Master en derecho penal") Universidad Autonoma de Nuevo Leon*. Obtenido de <http://eprints.uanl.mx/1412/1/1020148989.PDF>
- Esnel Montero Hechavarria, M. (Abril de 2011).** *Scielo*. Obtenido de Realidades de la violencia familiar en el mundo contemporáneo: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1029-30192011000400016
- Garnica, N. J. (15 de diciembre de 2011).** *Scielo*. Obtenido de Exploración de la violencia sexual en la ciudad de Bogotá: una aplicación de técnicas de minería de datos: <http://www.scielo.org.co/pdf/crim/v53n2/v53n2a07.pdf>
- INEGI. (12 de Mayo de 2017).** Obtenido de Tiempo digital: <https://tiempodigital.mx/2017/05/12/los-datos-duros-de-la-violencia-intrafamiliar-en-mexico/>
- López, C. P. (2007).** *Minería de datos: Técnicas y herramientas*. Madrid, España: Paraninfo Cengage Learning.
- María García Jiménez, A. Á. (s.f.).** *Universidad Carlos III*. Obtenido de Análisis de Datos en WEKA – Pruebas de selectividad: <http://www.it.uc3m.es/jvillena/irc/practicas/06-07/28.pdf>
- Orallo, J. H. (s.f.).** *Introducción a la minería de datos*. España: Pearson .
- Pilar Jara Romero, A. R. (s.f.).** *Universitat Jaume-I*. Obtenido de Escala de evaluación del tipo y fase de la violencia de género (EETFGV): http://repositori.uji.es/xmlui/bitstream/handle/10234/77672/forum_2009_18.pdf?sequence=1&fbclid=IwAR02HfEfQdfEUkwz6O5ttD05nmoZq7GlkXe8bDYnZM9m70vsl0kBaK1JvWc
- Pineda, O. M. (julio de 2017).** *ResearchGate*. Recuperado el 12 de octubre de 2020, de https://www.researchgate.net/publication/325766285_Categorizacion_automatizada_de_tweets_sobre_el_tem_a_politico_electoral_aplicando_algoritmos_de_clasificacion_supervisada#pfc
- SurveyMonkey. (12 de junio de 2020).** *SurveyMonkey*. Obtenido de SurveyMonkey: <https://www.surveymonkey.com/mp/sample-size-calculator/>

Teruelo, J. G. (2017). *Revista Electrónica de Ciencia Penal y Criminología*. Obtenido de Diagnóstico del sistema de protección y propuestas de intervención para la predicción y prevención de feminicidios en contexto de pareja o expareja: <http://criminnet.ugr.es/recpc/19/recpc19-23.pdf>

Información de los autores.



Miriam Vargas Reyes estudiante del noveno semestre de la carrera de Ingeniería en Sistemas Computacionales en el Tecnológico de Estudios Superiores de Valle de Bravo, Estado de México.



Mariana Carolyn Cruz Mendoza es Ingeniero en Sistemas Computacionales por el Tecnológico de Estudios Superiores de Valle de Bravo, Estado de México, cuenta con Maestría en Ingeniería en Sistemas Computacionales por el Tecnológico de Estudios Superiores de Ecatepec. Tiene la distinción de Perfil Deseable ante PRODEP y es Miembro del Cuerpo Académico en Formación “Procesamiento de datos con industria 4.0” del Tecnológico de Estudios Superiores de Valle de Bravo.



César Primero Huerta es Ingeniero en Sistemas Computacionales por el Tecnológico de Estudios Superiores de Valle de Bravo, Estado de México, cuenta con Maestría en Ingeniería en Sistemas Computacionales por el Tecnológico de Estudios Superiores de Ecatepec. Tiene la distinción de Perfil Deseable ante PRODEP y es Líder del Cuerpo Académico en Formación “Procesamiento de datos con industria 4.0” del Tecnológico de Estudios Superiores de Valle de Bravo.

Análisis de un bot para la extracción de información en interacciones farmacológicas.

Analysis of a bot for the extraction of information in drug interactions.

Cesar Eduardo Placido Velazco (1).

Estudiante. Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla.

cesar.placido@alumno.buap.mx.

Luis Enrique Colmenares Guillén* (2). Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, enrique.colmenares@correo.buap.mx.

Gustavo López y López (3). Facultad de Ciencias Químicas, Benemérita Universidad Autónoma de Puebla, jose.lopez@correo.buap.mx.

José Luis Hernández Ameca (4). Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, joseluis.hdameca@correo.buap.mx.

Maya Carrillo Ruíz (5). Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, maya.carrillo@correo.buap.mx.

*corresponding author.

Artículo recibido en diciembre 21, 2020; aceptado en febrero 11, 2021.

Resumen.

En este trabajo se analiza el procedimiento para extraer información acerca de interacciones farmacológicas y amplía la información médica obtenida del sitio web iDoctus, siguiendo como modelo la guía de estándares hospitalarios para la prescripción idónea de un perfil farmacoterapéutico de un paciente con una prescripción médica. Con plataformas electrónicas como iDoctus México es posible consultar estas reacciones en diversos fármacos, el diccionario médico cuenta con información detallada para cada compuesto activo. Además, se prueba la funcionalidad de un software automatizado diseñado para extraer información precisa de fármacos que permitirá mejorar la farmacovigilancia en los pacientes de las unidades hospitalarias. Mediante la creación de archivos con formatos diferentes, permite la consulta de información de una forma más rápida y eficaz, posicionándose así, como una herramienta útil y fácil de usar.

Palabras clave: Farmacovigilancia, interacción farmacológica, idoneidad de prescripción.

Abstract.

This paper analyzes the procedure to extract information about pharmacological interactions, and expand the medical information obtained from the iDoctus website, following as a model the hospital standards guide for the ideal prescription of a pharmacotherapeutic profile of a patient with a medical prescription. With electronic platforms such as iDoctus Mexico it is possible to consult these reactions in various drugs, the medical dictionary has detailed information for each active compound. Furthermore, the functionality of automated software designed to extract precise information on drugs that will improve pharmacovigilance in patients in hospital units is tested. By creating

files with different formats, it allows information to be consulted in a faster and more efficient way, thus positioning itself as a useful and easy-to-use tool.

Keywords: Pharmacovigilance, drug interaction, appropriateness of prescribing.

1. Introducción.

La Organización Mundial de la Salud en 1985, definió el uso racional de la siguiente manera: Usar racionalmente los medicamentos significa que “los pacientes reciban fármacos apropiados para sus necesidades clínicas, a dosis ajustadas a su situación particular, durante un período adecuado de tiempo y al mínimo costo posible para ellos y para la comunidad” (Organización Mundial de la Salud, 2006). La mayoría de los gobiernos incorporados a dicha organización han implementado diferentes políticas en salud para cumplir con el uso adecuado de los medicamentos. A nivel latinoamericano, en el 2010 se dan a conocer los resultados del Estudio Iberoamericano de Eventos Adversos (estudio IBEAS), como resultado de la colaboración de la Organización Mundial de la Salud (OMS), la Organización Panamericana de la Salud (OPS), el Ministerio de Sanidad y Política Social de España, y los Ministerios de Salud e instituciones de Argentina, Colombia, Costa Rica, México y Perú. Los resultados de este estudio indican que, de cada 100 pacientes ingresados, 20 presentaron al menos un incidente dañino a lo largo de su estancia en el hospital y que de cada 100 pacientes que sufrieron incidentes dañinos, 7 murieron, 17 quedaron con una incapacidad total, 12 con una incapacidad severa y 64 con incapacidades leves o sin incapacidad (Organización Mundial de la Salud, 2006). Los eventos adversos detectados se relacionaron con la infección nosocomial en un 35.99%, con algún procedimiento en un 26.75%, con los cuidados en un 16.24%, con el uso de la medicación en un 9.87%, y con el diagnóstico en un 5.10% (Ministerio de Sanidad y Política Social en 2010). También se describe que de cada 10 incidentes debidos a los cuidados sanitarios que produjeron daño, casi 6 podrían haberse evitado.

En México, la farmacovigilancia tiene sus primeros avances desde 1997 (SiNaCEAM, 2018) con la creación del Centro Nacional de Farmacovigilancia (CNFV) y su integración al Programa Internacional de Farmacovigilancia coordinado por la OMS y su Centro Colaborador en Uppsala, Suecia. Más recientemente, con la actualización del Reglamento de Insumos para la Salud y del Programa de Reformas al Sistema Nacional de Salud, el gobierno de México asume la responsabilidad de garantizar la calidad, la seguridad y la eficacia de los medicamentos que se comercializan y utilizan en el país. En este sentido la Secretaría de Salud establece las políticas y define los requerimientos en todo el país a través de los centros estatales e institucionales de farmacovigilancia a través del Centro Nacional de Farmacovigilancia que es el organismo que coordina y unifica las actividades de farmacovigilancia en el país. Los principales objetivos de la farmacovigilancia son: a) la detección temprana de las reacciones adversas e interacciones desconocidas hasta ese momento, b) la detección de aumentos de la frecuencia de reacciones adversas (conocidas), c) la identificación de factores de riesgo y de los posibles mecanismos subyacentes de las reacciones adversas, d) la estimación de los aspectos cuantitativos de la relación beneficio/riesgo y difusión de la información necesaria para mejorar la regulación y prescripción de medicamentos, e) el uso racional y seguro de los medicamentos, f) la evaluación y comunicación de los riesgos y beneficios de los medicamentos comercializados, y g) la educación y la información a los pacientes. Para cumplir con dichos objetivos, se establece la Norma Oficial Mexicana (NOM-220-SSA1-2002) (Norma Oficial Mexicana, 2002) sobre la instalación y operación de la farmacovigilancia. Esta norma menciona que los profesionales de la salud deben identificar y notificar los efectos colaterales de los medicamentos, es decir, identificar cualquier efecto no intencionado de un producto farmacéutico que se produzca con dosis normalmente utilizadas en el hombre, y que esté relacionado con las propiedades farmacológicas del medicamento.

Los sistemas de consulta con bases de datos almacenan una enorme cantidad de información de diccionarios médicos, medicamentos, pacientes, historiales médicos, interacciones farmacológicas y otros datos. Actualmente, estos sistemas son los más utilizados por su funcionalidad y efectividad al consultar información, sin embargo, no utilizan algoritmos de clasificación para predecir una interacción farmacológica. Las bases de datos de los sistemas de consulta se pueden utilizar como fuentes de información para generar nuevas investigaciones como es el caso de este proyecto, al utilizar la información del diccionario médico iDoctus para desarrollar un corpus de interacciones farmacológicas. En la farmacovigilancia los métodos de causalidad o imputabilidad son la evaluación estructurada y

normalizada, en pacientes individuales o casos clínicos, para determinar la probabilidad de que una interacción farmacológica sea causada por un medicamento sospechoso. Los métodos de causalidad en sus algoritmos utilizan un conjunto de preguntas que pretenden calificar cada sospecha sobre la relación medicamento-reacción con un determinado grado de probabilidad (Alcántara, L., 2010). Este método realiza la identificación de una interacción farmacológica posteriormente a la administración de un medicamento. A diferencia de este proyecto que pretende predecir e identificar una interacción antes de la administración de un medicamento en un paciente.

En el artículo llamado *Data-Driven Prediction of Drug Effects and Interactions* de la revista *Science Translational Medicine* (Tatonetti, Patrick, Daneshjou, & Altman, Data-Driven, 2012) investigadores de la escuela de medicina de la Universidad de Stanford, en Estados Unidos, diseñaron un algoritmo que analiza millones de informes de la Agencia de Alimentos y Medicamentos de Estados Unidos (FDA, por sus siglas en inglés) y que identifica los efectos secundarios así como las posibles interacciones entre miles de medicamentos a partir de diferentes bases de datos. En este artículo se presenta que las reacciones no deseadas de medicamentos es una de las principales causas de mortalidad en todo el mundo porque muchas de estas reacciones no se detectan durante los ensayos clínicos, antes de que un fármaco sea aprobado para su uso médico y comercial. Este artículo confirmó 47 de las 395 interacciones que esperaba identificar.

La revisión de la idoneidad de la prescripción tiene como objetivo revisar que la medicación sea la adecuada para cada paciente en particular, considerando sus características clínicas, fisiológicas, interacciones farmacológicas, historia de alergias, entre otras. La revisión de la idoneidad de la prescripción se realiza basándose en la realización de un perfil farmacoterapéutico a cada paciente. Este perfil incluye todos los medicamentos prescritos y administrados, esta información permite llevar a cabo el proceso de análisis para revisar la idoneidad de la prescripción. Como es una barrera de seguridad, la revisión de la idoneidad se debe realizar antes de la administración de los medicamentos y, si se detectan problemas relacionados con la medicación, se contacta a la persona que prescribió el medicamento, para determinar la conducta a seguir. El proceso para revisar la idoneidad de una prescripción incluye, la evaluación y el análisis de:

- Dosis, la frecuencia y la vía de administración.
- Duplicación terapéutica.
- Alergias o sensibilidades.
- Interacciones reales o potenciales entre el medicamento y otros medicamentos o alimentos.
- Variación con respecto al criterio del uso del medicamento en el establecimiento;
- Peso del paciente y demás información fisiológica.
- Otras contraindicaciones.

El profesional de la salud analiza el perfil farmacoterapéutico (medicamentos prescritos) a fin de identificar algún problema relacionado con el medicamento (PRM), es decir alguna situación que en el proceso de uso de medicamentos puedan causar la aparición de un resultado negativo asociado a la medicación. Es decir, para cada medicamento se revisa lo siguiente:

- Vía de Administración del medicamento.
- Conservación del medicamento.
- Dosis, pauta y/o duración.
- Duplicidad de dosis.
- Errores en la dispensación.
- Errores en la prescripción.
- Incumplimiento.
- Interacciones.
- Otros problemas de salud que afectan al tratamiento (comorbilidades).
- Probabilidad de efectos adversos.
- Problema de salud insuficientemente tratado.
- Características personales del paciente.
- Contraindicaciones.

El proceso para revisar la idoneidad de la prescripción se realiza de forma manual en la mayoría de los casos. Es decir, se contrasta la información de las guías de práctica clínica y programas de información de medicamentos con los datos de la prescripción del medicamento elaborada por el médico. Desafortunadamente esta actividad requiere gran inversión de tiempo por cada paciente (aproximadamente 30 minutos) dando lugar a que el proceso sea lento en el suministro de los medicamentos, las intervenciones de la farmacia con el medico se invierte mucho tiempo, y además aumenta la probabilidad de errores humanos.

El presente proyecto es una extensión de un trabajo previo en el cual se desarrolló un algoritmo de validación y que clasifica interacciones farmacológicas a partir de información extraída del sitio web de medicina iDoctus. Para ello se generó un modelo de clasificación basado en el algoritmo Naïve Bayes que calcula la probabilidad de que exista una interacción farmacológica entre medicamentos (Luis E. Colmenares-Guillen, 2019).

El objetivo de este trabajo además de extraer información acerca de interacciones farmacológicas es ampliar la información médica obtenida del sitio web iDoctus, siguiendo como modelo la guía de estándares hospitalarios para la prescripción idónea de un perfil farmacoterapéutico, tratando de cubrir el resto de los puntos vistos previamente y así ayudar a cualquier profesional de la salud a realizar esta tarea de manera más sencilla.

2. Métodos.

Para llevar a cabo este trabajo se construyó un modelo de clasificación basado en el diccionario digital médico iDoctus México el cual cuenta con un comprobador de interacciones que permite verificar la seguridad del tratamiento completo de un paciente (iDoctus, 2016).

Etapas de desarrollo.

- Recolección de datos
- Clasificación de información
- Interacciones.

La recolección de datos se realiza mediante una aplicación de software que ejecuta tareas automatizadas al diccionario digital medico iDoctus del cual se obtiene información tal como interacciones, advertencias, contraindicaciones, dosis, frecuencia y vía de administración de diferentes medicamentos.

Esta aplicación está escrita en el lenguaje de programación Python en su versión 3.8.0 (Python, 2020) con una interfaz gráfica basada en la librería PyQt5 (PyQT5, 2020) la cual permite generar una interacción clara con el usuario. Se ingresa con las credenciales a la página del diccionario iDoctus, una vez dentro se busca la información que se requiere generando distintos archivos de salida. Perfiles de pacientes son creados para que se tenga un historial clínico digital y así se puedan detectar otras posibles interacciones farmacológicas. En la fig. 1, se muestra el diagrama general de funcionamiento del software.

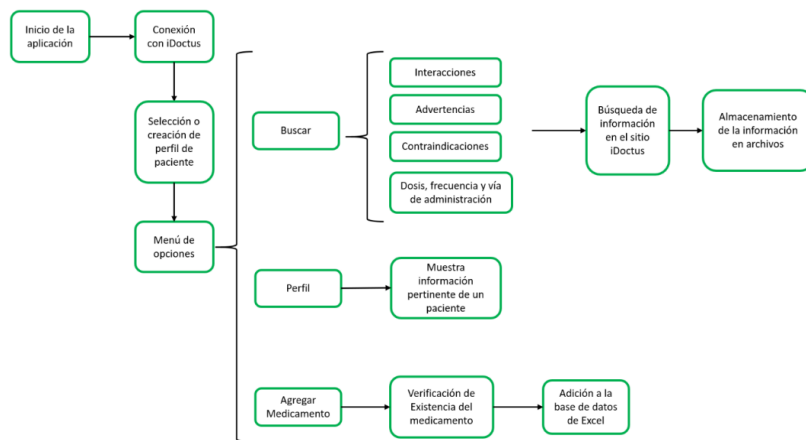


Figura 1. Diagrama de comportamiento general del software automatizado.

Los sistemas de medicación hospitalarios son bastante complejos, por lo cual se identifican cuatro etapas principales las cuales son servicio de admisión, servicio clínico, servicio de enfermería y servicio farmacéutico.

En el primer servicio se tiene el ingreso del paciente a la unidad médica, después se encuentra el servicio clínico el cual se encarga del llenado de expedientes electrónicos con datos generales, diagnóstico, tratamiento y prescripción. El siguiente servicio es el de enfermería donde se elabora la requisición o solicitud de medicamentos. En la última etapa está el servicio farmacéutico en donde se crea el perfil farmacoterapéutico es ahí donde se tienen elementos medibles para la idoneidad de la prescripción. Se evalúa si el perfil farmacoterapéutico es idóneo, si resulta cierto pasará al servicio de enfermería para su administración, de lo contrario regresará al servicio clínico para su revaloración. La Fig. 2. Muestra esta secuencia en un diagrama de flujo.

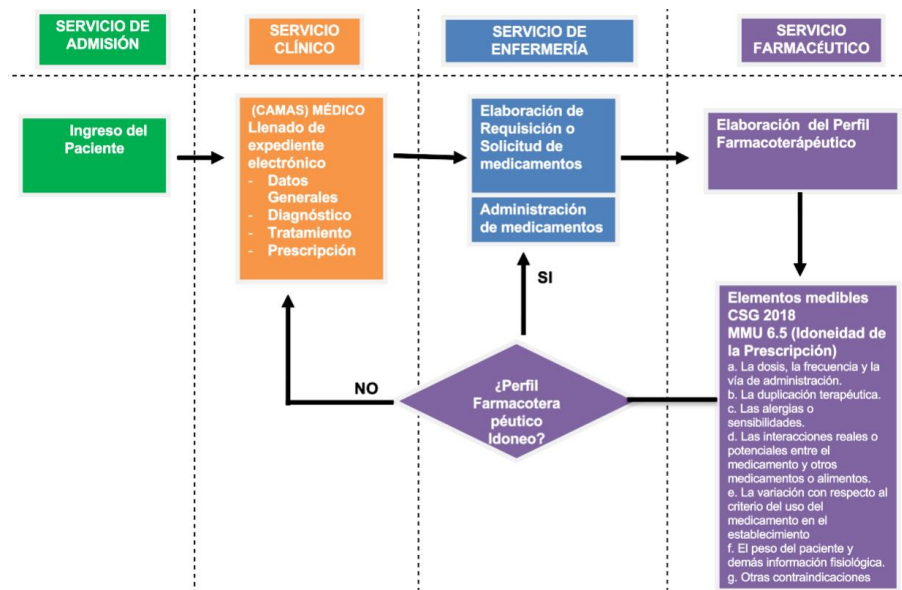


Figura 2. Diagrama simplificado de distribución de medicamentos en unidades hospitalarias.

3. Desarrollo.

El proceso de recolección de información comienza con el inicio de sesión en la página web del diccionario médico iDoctus a través del software automatizado, una vez que el servidor permite el acceso se solicita al usuario que, después de seleccionar el perfil del paciente, indique el nombre del fármaco a buscar y así como las características que se requieren conocer como son indicaciones, advertencias, contraindicaciones, dosis, frecuencia y vía de administración.

El funcionamiento de la página iDoctus es como la mayoría de las páginas web un modelado en contenedores donde cada contenedor almacena algún elemento que compone la página, lo que nos interesa, es la manera en la que muestra la información y como la almacena en la página, ya que será la que se utilice para poder tener el texto plano con las interacciones que se buscan. El modelado es el siguiente, después de que se busca un fármaco se divide en secciones, las cuales son advertencias, dosis, indicaciones terapéuticas, interacciones, reacciones adversas, contraindicaciones, farmacología, precauciones generales, embarazo y lactancia, otra información de seguridad, presentaciones y ver ficha técnica completa. Lo que respecta a este trabajo es la sección de interacciones.

Esta sección está comprendida por un contenedor que a su vez contiene otros dos contenedores el primero tiene información sobre interacciones con alimentos o alcohol. Lo que se busca se encuentra en el segundo contenedor el cual contiene las interacciones con medicamentos, las cuales pueden ser graves, moderadas y leves. Cada grupo de interacciones a su vez está comprendido en un contenedor y cada contenedor está dividido de la siguiente manera: en la cabecera del contenedor esta la pareja de fármacos después está dividida en secciones la primera y la que nos importa es el efecto, luego la de importancia, mecanismo, evidencias y referencias. Así es, con cada interacción. La manera de extraer el contenido de cada grupo de interacciones esta agrupadas en un contenedor. Por conveniencia este contenedor debe tener un id que lo identifique, aunque no es el caso, porque lo que tiene es una clase (una clase es un identificador que sirve para poder aplicar estilos css a un contenedor entre otras opciones que tiene que ver con el estilo en el que se muestra la información) Al no poseer un id, lo que se utilizó para poder extraer la información en texto plano fue un Query en JavaScript (un Query es un método que dependiendo del tipo, devuelve elementos que coinciden con el elemento seleccionado). Después de ejecutar el Query devuelve todo el contenido de la clase, en este caso el que contiene las interacciones.

El medicamento es buscado usando la dirección URL con el numero identificador del medicamento para tener acceso a toda su información, una vez que el servidor responde con los datos solicitados se extrae el contenido de toda la página en formato html, es ahí donde comienza la clasificación de información que se realiza con palabras clave que fueron previamente dadas por el usuario. Se crean diccionarios locales para disminuir el número de consultas web. Una vez que la información fue clasificada y organizada, se vacía en archivos con formatos varios listos para ser consultados. En el siguiente diagrama se describe el diagrama de funcionalidad. La estructura general del software se describe en la figura 3.

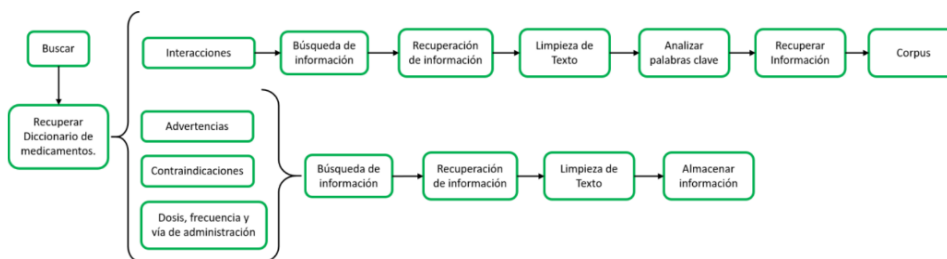


Figura 3. Diagrama de funcionalidad de método Buscar.

Búsqueda de medicamentos en el diccionario digital iDoctus. Las consultas de fármacos se realizan con un ID el cual se describe en la URL, con ese identificador se puede obtener datos como son interacciones o advertencias.

Algoritmo 1. Ejemplo de método de búsqueda de medicamentos en la página iDoctus.

```

def buscar(self,info,id,farm):
    try:
        self.farmaco = farm
        url='http://mx.idoctus.com/consulta/medicamento/idpa/'+str(id)+'/'
        aux_url = 'http://mx.idoctus.com/consulta/medicamento/medicamento/'+str(id)+'/'
        info[1] = info[0].get(url, headers=self.headers)
        soup = BeautifulSoup(info[1].text, "html.parser")
        element = soup.find("div", attrs={"class": self.CONTENT_CLASS_NAME})
        if element == None:
            info[1] = info[0].get(aux_url, headers=self.headers)
            soup = BeautifulSoup(info[1].text, "html.parser")
            element = soup.find("div", attrs={"class": self.CONTENT_CLASS_NAME})
        else:
            pass
    except Exception as e:
        self.error_log(':::::: Error durante la busqueda {0} : {1}'.format(url, str(e)))
        element=1
    return element

```

El algoritmo describe el método de búsqueda de un fármaco, se usa la dirección url del diccionario medico iDoctus y se concatena el id o nombre del medicamento a consultar, después se hace una validación de la existencia de la página y se comprueba que no se trate de una página en blanco, de ser así se avisa con un mensaje de error.

En la figura 4, se puede observar a la izquierda un ejemplo de búsqueda de información en tiempo de ejecución con salida en terminal, ahí se muestra el número de interacciones para dos medicamentos separando los resultados en cuatro categorías: interacciones graves, moderadas, leves y sin tipo. Esos datos son guardados en archivos locales.

A la derecha de la imagen se muestra la interfaz de usuario de Bot iDoctus, esta sección muestra el apartado de búsqueda donde se permite elegir los datos que se desean recuperar como son interacciones, contraindicaciones, advertencias y dosis. También se muestran los otros menús disponibles dentro de la aplicación como es el apartado de perfil del paciente.

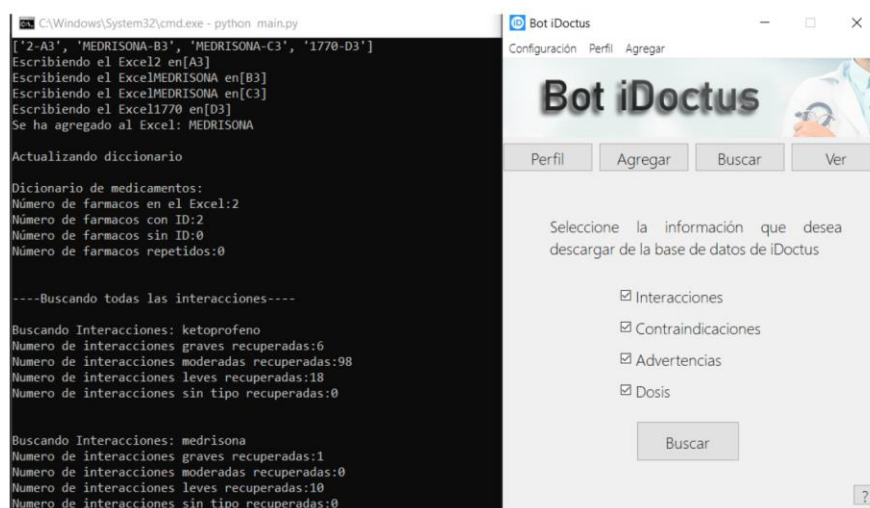


Figura 4. Búsqueda de Información.

Además de los archivos asociados con cada criterio de búsqueda, también se utiliza el archivo log, para llenar el registro de actividades durante la ejecución del programa. Este inicia, desde que se hace el ingreso a la cuenta de iDoctus hasta que el programa finaliza, el cual muestra la fecha y hora de cuando se realiza. El archivo se encuentra bajo el nombre “log” en la carpeta de archivos nuevamente, y la escritura de datos es controlada por la función `error_log()` de la clase archivo, y se solicita dentro de la mayoría de clases, pues como ya se mencionó, se va haciendo un registro de cada acción realizada por el programa, ver figura 5.

Name	Date modified	Type	Size
 201760284	11-May-20 9:03 PM	Microsoft Excel W...	5 KB
 corpus_if	11-May-20 9:03 PM	TXT File	31 KB
 preCorpus	11-May-20 9:03 PM	TXT File	30 KB
 Texto_Adv	11-May-20 9:03 PM	TXT File	1 KB
 Texto_Contra	11-May-20 9:03 PM	TXT File	1 KB
 Texto_Dosis	11-May-20 9:03 PM	TXT File	2 KB
 Texto_Interacciones	11-May-20 9:03 PM	TXT File	672 KB

Figura 5. Archivos de salida para un paciente.

Conclusiones.

La idoneidad de las prescripciones farmacológicas es una práctica importante en el sistema de distribución de medicamentos, esta herramienta propone una consulta precisa y rápida de las interacciones de diferentes fármacos prescritos a pacientes en unidades hospitalarias, gracias a la información disponible en el diccionario médico iDoctus la búsqueda de interacciones se puede escalar a otros ámbitos no menos importantes como son dietas, edad del paciente, otras enfermedades, por mencionar algunas. Con el continuo soporte del lenguaje de programación Python y con la constante integración de funciones y librerías, se da la oportunidad de optimizar las funciones y módulos para una mejor adaptación a las nuevas generaciones de hardware disponibles en el mercado mejorando así su implementación en los sistemas de medicación. Se concluye la primera etapa de Interacciones este trabajo que brinda un proceso automatizado a las áreas de ciencias de la computación, ciencias químicas, farmacología, farmacovigilancia y medicina para predecir e identificar la gravedad de las interacciones farmacológicas en tres clases: leve, moderada y grave.

Agradecimientos.

Agradecimientos. Al MPS Juan Carlos Bastida Herrera de la Facultad de Ciencias Químicas de la BUAP por su invaluable aportación. A la Vicerrectoría de Investigación y Estudios de Posgrado (VIEP) y a la Dirección de Innovación y Transferencia de Conocimiento de la Benemérita Universidad Autónoma de Puebla.

Referencias Bibliográficas.

- Alcántara, L. (2010).** Aplicación de Algoritmos Estandarizados internacionalmente para evaluar Notificaciones Espontáneas de Sospechas de Reacción Adversa en el CNFV del año 2010. Cd. de México: Universidad Nacional Autónoma de México.
- iDoctus (2016).** iDoctus, Medicina en la palma de tu mano 2016 recuperado de <https://mx-public.idoctus.com/conoce-idoctus> el 18 de abril de 2020.

- Luis E. Colmenares-Guillen (2019).** et al. Validación de un algoritmo de clasificación para la identificación de interacciones farmacológicas. Ingeniería Investigación y Tecnología volumen XX (número 2), abril-junio 2019 1-8 ISSN 2594-0732 FI-UNAM. <https://doi.org/10.22201/fi.25940732e.2019.20n2.014>.
- Norma Oficial Mexicana (2002).** NOM-220-SSA1-2002, Instalación y operación de la farmacovigilancia. Recuperado el 2 de septiembre de 2020 de <http://www.salud.gob.mx/unidades/cdi/nom/220ssa102.html>.
- Organización Mundial de la Salud (2006).** Uso racional de los medicamentos: progresos realizados en la aplicación de la estrategia farmacéutica de la OMS (2006, 11 mayo), recuperado 7 junio de 2019 de https://apps.who.int/gb/archive/pdf_files/EB118/B118_6-sp.pdf.
- Python (2020).** Welcome to Python.org. (s.f.-b). Recuperado 3 mayo, 2020 de <https://www.python.org/doc/>.
- PyQT5 (2020).** Python Software Foundation. (2019, 6 mayo). PyQt5. Recuperado 1 mayo 2020 de <https://pypi.org/project/PyQt5/>.
- SiNaCEAM (2018).** Modelo de seguridad del paciente del SiNaCEAM, Estándares para implementar el modelo en hospitales Edición 2018, recuperado 12 de junio de 2019 de http://www.csg.gob.mx/descargas/pdf/certificacion-establecimientos/modelo_de_seguridad/hospitales/Estandares-Hospitales-Edicion2018.pdf.
- Tatonetti, Patrick, Daneshjou, & Altman, Data-Driven (2012).** Prediction of Drug Effects and Interactions de la revista Science Translational Medicine Sci Transl Med. 2012 Mar 14; 4(125). doi: 10.1126/scitranslmed.3003377, 2012.

Información de los autores.



Cesar Eduardo Placido Velazco es estudiante de ingeniería en la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla. En 2016, Diplomado en inglés The Anglo Mexican Foundation, Cambridge Certificate of Proficiency in English. Es desde el 2016 miembro del laboratorio de autoacceso bajo la dirección del investigador de tiempo completo Dr. Luis Enrique Colmenares Guillén trabajando en diferentes áreas como son: modelado e impresión en 3D, montaje y mantenimiento de servidores y clustering de alto desempeño para la realización de pruebas de diversos proyectos.



Luis Enrique Colmenares Guillén. Actualmente es profesor investigador de tiempo completo de la Facultad de Ciencias de la Computación de la BUAP. Pertenece a un cuerpo académico PRODEP consolidado. Es miembro del Registro CONACYT de Evaluadores Acreditados (RCEA). Pertenece al Sistema Nacional de Investigadores (SNI) y a dos redes temáticas CONACYT. Pertenece al padrón de Investigadores y de Consultores de la BUAP. Tiene publicaciones nacionales e Internacionales, además

tiene cuatro patentes registradas y dos títulos de patente. En la Facultad de Ciencias de la Computación ha impartido las cátedras de Sistemas Operativos, Administración de proyectos, Sistemas Distribuidos, Procesamiento Digital de imágenes, Sistemas de tiempo real. Actualmente ha desarrollado algoritmos y sistemas clasificadores para el área de la Inteligencia artificial y reconocimiento de patrones.



Gustavo López y López, Profesor de tiempo completo que ha impartido las cátedras de Farmacia Clínica, Farmacia Hospitalaria y Farmacología para los alumnos de la Licenciatura en Farmacia. Ha desarrollado líneas de aplicación de conocimiento en el área de gestión de servicios farmacéuticos en el ambiente hospitalario en instituciones de salud públicas y privadas. El objetivo general de dichas líneas se centra en la detección y resolución de problemas relacionados con los medicamentos durante la logística de suministro de insumos para la salud y mejorar el manejo y uso de los medicamentos. Los resultados de estas actividades se han publicado en revistas internacionales indizadas.



José Luis Hernández Amecca. Obtuvo el título de Ingeniero Electrónico del Instituto Tecnológico de Puebla en 2001. En 2005 recibió el título de Maestro en Ciencias de la Computación por la Benemérita Universidad Autónoma de Puebla (BUAP). En 2017 obtuvo el título de Doctor en Planificación Estratégica y Gestión de la Tecnología de la Universidad Popular Autónoma de Puebla (UPAEP). De 2002 a 2004 trabajó como Ingeniero en el departamento de implementación de proyectos de la empresa digital IUSACEL. Trabaja como Profesor en la facultad de Ciencias de Computación de la BUAP desde 2007, donde su línea de investigación es la robótica y educación.



Maya Carrillo Ruiz. Doctorado en Ciencias Computacionales por el Instituto Nacional de Astrofísica Óptica y Electrónica. Estancia de investigación en la Universidad de Waterloo, Canadá. Empleada de IBM de México de 1991 a 2000. Profesora Investigadora de la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla desde el año 2000. Ha fungido como vocal de la Sociedad Mexicana de Inteligencia Artificial. Su trabajo de investigación, se centran en la aplicación de métodos novedoso de aprendizaje automático principalmente en las áreas de análisis de sentimientos, determinación del perfil de autores y análisis de señales de electroencefalograma (EEG).

Animatronic con forma de araña hexápoda.

Animatronic shaped like a hexapod spider.

Raúl Moreno Rincón* (1).
Tecnológico Nacional de México campus Tuxtla Gutiérrez
raull.mr@tuxtla.tecnm.mx.

Lizette Adriana Zebadúa Chavarría (2), Estudiante del Tecnológico Nacional de México campus Tuxtla Gutiérrez,
lzebadua@ittg.edu.mx.

Álvaro Hernández Sol (3). Tecnológico Nacional de México campus Tuxtla Gutiérrez, alvarohs@tuxtla.tecnm.mx.

José Ángel Zepeda Hernández (4). Tecnológico Nacional de México campus Tuxtla Gutiérrez,
jose.zh@tuxtla.tecnm.mx.

Rubén Herrera Galicia (5). Tecnológico Nacional de México campus Tuxtla Gutiérrez, ruben.hg@tuxtla.tecnm.mx.

*corresponding author.

Artículo recibido en octubre 29, 2021; aceptado en noviembre 25, 2021.

Resumen.

En este artículo se realizó el diseño y construcción de un sistema robótico con forma de araña hexápoda con alimentación autónoma, movimiento constante, estable y sostenido. El mecanismo diseñado para cumplir los movimientos deseados se centra en el uso de servomotores, el sistema de control se basa en un microcontrolador PIC. La aplicación de 6 extremidades al robot permite alcanzar mayor estabilidad sobre todo cuando se detiene, ya que puede quedar parado en 3 patas en formato de tripie, el uso de servomotores permite facilidad la programación de la secuencia de movimientos, ya que son especialmente diseñados para el control de posición que es el caso que nos compete, y la utilización del microcontrolador PIC, aunque requirió del diseño de su shell para su funcionamiento es considerado de los microcontroladores más rápidos y estables del mercado, así también presenta gran cantidad de memoria que permite diseño de programas extensos aplicando métodos novedosa de inteligencia artificial para control. Por todo lo anterior el objetivo de este diseño es contar con un robot versátil que permita aceptar diferentes métodos de control para poder realizar un análisis comparativo entre ellos.

Palabras clave: Hexápoda, grados de libertad, tripié, shell, radial.

Abstract.

In this paper the design and construction of a robotic spider-shaped system with autonomous feeding, constant, stable and sustained movement was carried out. The mechanism designed to fulfill the desired movements focuses on the use of servomotors, the control system is based on a PIC microcontroller. The application of 6 limbs to the robot allows to achieve greater stability especially when it stops, since it can stand on 3 legs in a tripping format, the use of servomotors allows easy programming of the sequence of movements, since they are specially designed to the position control that is the case that concerns us, and the use of the PIC microcontroller, although it required the design of its shell for its operation is considered the fastest and most stable microcontrollers on the market, so it also has a large

amount of memory that allows Extensive program design applying innovative artificial intelligence methods for control. For all the above, the objective of this design is to have a versatile robot that allows to accept different control methods in order to perform a comparative analysis between them.

Keywords: Hexapod, degrees of freedom, tripod, shell, radial.

1. Introducción.

La Robótica es una ciencia que surge a finales de la década de los 50's, y que, ha demostrado ser un importante motor para el avance tecnológico que impactan diversas áreas del conocimiento, tales como la industria de manufactura, las ciencias médicas, la industria espacial, entre otras. Así pues, se generan expectativas para un tiempo no muy lejano.

En un principio esta se centró en la automatización de procesos de producción. Esto se ve reflejado sobre todo en la industria automotriz. Debido al alto costo que representa el automatizar y robotizar un proceso de producción, la tendencia actual en Robótica es la investigación de micro-robots y robots móviles autónomos con cierto grado de inteligencia, este último es el campo en el que se basa este proyecto de investigación, de acuerdo a Manzanarez (2017).

Lo anteriormente expuesto explica la necesidad y la importancia de que Institutos de Investigación, se den a la tarea de destinar recursos tanto económicos y humanos para aliviar el rezago tecnológico que el país y el estado padece.

El campo de los robots móviles es muy extenso, teniendo como clasificación principal el tipo de locomoción. Este proyecto se centra en el sistema de locomoción a través de extremidades, siendo la configuración hexápoda la más utilizada debido a que presenta mayor estabilidad, tanto estática como dinámica, además de ser prácticamente todo terreno y no limitarse a superficies planas como las ruedas, debido a que su avance se realiza por el levantamiento de dos o tres extremidades mientras que las otras lo soportan, esto le permite avanzar en terrenos rocosos e inclusive en gradas, sin embargo la capacidad de avanzar en terrenos de diferentes alturas depende del tamaño de las extremidades y del levantamiento máximo que estas pueden alcanzar, Hidalgo Panchana (2015). Esto hace que sea una alternativa viable para que sustituya o interactúe con el sistema de locomoción a base de ruedas, que es la que domina el campo de los robots móviles debido a su fácil control y aplicación, ver figura 1.



Figura 1. Robot Tarry.

El desarrollo de este tipo de sistema de locomoción se vio obstaculizado en un principio por la complejidad inherente a la coordinación de las extremidades. Esta complejidad necesita una alta capacidad de cálculo aplicando técnicas de control clásico, sin embargo, con el avance de la microelectrónica se ha hecho posible el acceso a recursos programables que hacen viable la implementación de este tipo de sistemas.

En el desarrollo de un robot hexápodo debe tenerse en cuenta aspectos tales como la robustez, mantenimiento del equilibrio, coordinación de los movimientos entre otros. Un sistema que ataque todos los aspectos y resuelva estos problemas resulta de difícil implementación y un estudio muy complejo (inteligencia artificial: algoritmos genéticos),

por lo cual el proyecto se centra en problemas puntuales: Construcción de chasis, extremidades, arquitectura de hardware, locomoción, tipos de sensores.

Los robots son usados hoy en día para llevar a cabo tareas sucias, peligrosas, difíciles o repetitivas para los humanos. Esto usualmente toma la forma de un robot industrial usado en las líneas de producción. La manufactura continúa siendo el principal mercado donde los robots son utilizados. Las aplicaciones incluyen soldado, pintado y carga de maquinaria. La Industria automotriz ha tomado gran ventaja de esta nueva tecnología donde los robots han sido programados para reemplazar el trabajo de los humanos en muchas tareas repetitivas.

Pero no sólo se pretenden utilizar a los robots en esta área, sino que se quiere llegar más lejos creando robots que tengan mayor movilidad. Por eso se han creado pequeños hexápodos y otros tipos de robots de múltiples piernas. Estos robots imitan insectos y artrópodos en funciones y forma que ofrecen gran flexibilidad y han probado adaptabilidad a cualquier ambiente. Por lo que la intención de esta investigación fue, diseñar y construir un sistema robótico con forma de araña de 6 patas para que con el movimiento de estas se pueda obtener un movimiento constante, estable y sostenido durante su avance, con una estructura robusta que soporte mayor peso y el acondicionamiento de una batería para darle mayor autonomía. También se pretendió dotar de sensores que detecten obstáculos para que cambie de trayectoria y no choque con estos.

2. Métodos.

Se consideró para el diseño mecánico del sistema; un estudio de los elementos empleados en su construcción (actuadores, sensores, etc.), la fisiología de la extremidad (grados de libertad, ángulos de barridos y avance lineal por cada articulación), limitaciones impuestas por el software de control y el hardware, etc., bajo la prioridad de que la funcionalidad básica de la extremidad es caminar. Para lo cual fue necesario una serie de actuadores (servomotores) que la impulsen a través del medio.

Un concepto importante a la hora de elegir el número de pata del robot es el de estabilidad estática. Este término se refiere a la capacidad del robot para permanecer estable (sin caerse) cuando no está en movimiento. Es más fácil mantener la estabilidad estática en un robot hexápodo que en uno de menor número de patas debido a que hay más patas libres para reposicionar el cuerpo del robot mientras éste se apoya en tres de ellas.

Existen dos configuraciones de distribución de las patas alrededor del cuerpo del robot. La bilateral, ver figura 2, Chamba (2019). Esta distribución presenta una simetría a lo largo del eje longitudinal del robot. Tiene la ventaja de una mayor simplicidad a la hora de programar los movimientos, debido a que el robot está preparado físicamente para facilitar el avance en direcciones paralelas al eje de simetría.

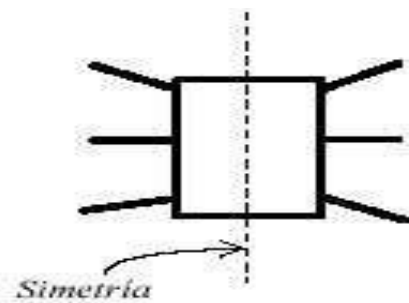


Figura 2. Robot Hexápodo bilateral.

Y la distribución radial, que fue la escogida para nuestra araña, donde cualquier dirección es exactamente igual a cualquier otra. En dos dimensiones, esta distribución es totalmente simétrica, y eso lleva a un software más genérico y más simple. No necesita movimientos específicos de giro puesto que es capaz de desplazarse en cualquier dirección

sin tener previamente que cambiar de orientación (omnidireccional), ver figura 3, diseño realizado en el software Autocad Mechanical

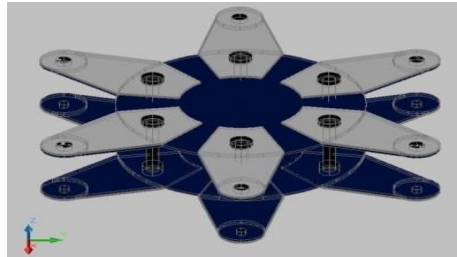


Figura 3. Robot Hexápodo radial.

3. Desarrollo.

Para realizar el movimiento de cada una de las articulaciones de cada extremidad se utilizaron los servomotores MG995 (servo1 para el avance-retroceso y servo 2 para subir y bajar la pata) que son motores de corriente continua que barren un ángulo de 0° a 180° , determinado por una señal de control y que cuenta con un sistema de engranaje metálico que le permite mayor robustez. En lo referente a la señal de control del servo, este recibe un pulso con duración equivalente al ángulo que debe tomar el servo con respecto a su referencia. Destacar que estos ángulos (estas posiciones) son absolutos, ver la figura 4 (tomada del datasheet de servomotores)



Figura 4. Servomotor MG995.

Diseño de las patas: Desde el primer momento se pensó en utilizar servos para las articulaciones, así como la aplicación de dos articulaciones (grados de libertad) por cada pata, una para levantarla y otra para desplazarla (12 servos). Se diseñó una estructura de tipo (paralelograma deformable) la cual consiste en un paralelogramo cuyos ángulos pueden variar mediante servos situados lejos de la articulación, este diseño agrega un grado más de libertad por cada pata (3 grados de libertad en cada una), de acuerdo a Rivas (2015), sin embargo, por conceptos técnicos, ya que solamente vamos a controlar 2 articulaciones con los servos 1 y 2 y esta tercera articulación únicamente va a servir para plegar la pata al ser levantada y lo hace por sistema mecánico de paralelograma deformable, preferimos obviarla y considerar que solo se tienen dos. Ver figura 5 diseñada en el software Autocad Mechanical para nuestro proyecto.

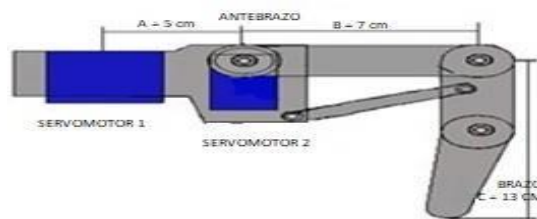


Figura 5. Diseño de una extremidad

Se determinó el tamaño de las patas y las ecuaciones (EC.1 y EC.2) para calcular la distancia de avance en función del valor del ángulo que se le asigna al servomotor 1 y la altura de levantamiento en función del servomotor 2 que levanta y baja la extremidad durante el recorrido, aplicando trigonometría básica para triángulos isósceles y rectángulos, ver figuras 6 y 7 (figuras diseñadas para la presentación).

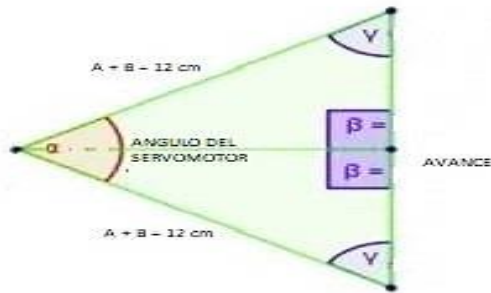


Figura 6. Distancia de avance en cada paso

$$avance = 2(A + B) * \sin\left(\frac{\alpha}{2}\right) \quad (1)$$



Figura 7. Altura de levantamiento de cada pata

$$altura = B * \sin(\alpha) \quad (2)$$

El software utilizado para el diseño y simulación fue el Autocad Mechanical, que tiene capacidad de diseñar piezas de mecanismos con dimensiones reales y ensamblarlo para realizar la simulación de su comportamiento, ver figura 8. Las patas (6 patas) fueron impresas en PLA al 95% dando un peso total de 50.8 grs. cada una y considerando, que el cuerpo lo conforman 2 placas de PLA de 55.3 grs. cada una, el peso resultante fue de:

$$415.4 \text{ grs.} = 2(55.3 \text{ grs.}) + 6(50.8 \text{ grs.}) \quad (3)$$

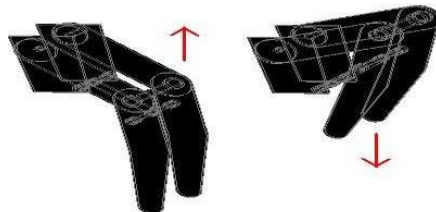


Figura 8. Diseño de las extremidades y simulación de funcionamiento

Tarjeta de control; El circuito está formado por un microcontrolador PIC16F887A al que previamente se le carga el programa de control. Al PIC se le conecta un cristal de cuarzo de 4Mhz, los condensadores de 22pF para el oscilador y un pulsador que cuando se activa pone un nivel bajo en la entrada MCLR para provocar un RESET del sistema, ver figura 9, diseño realizado en el software Proteus.

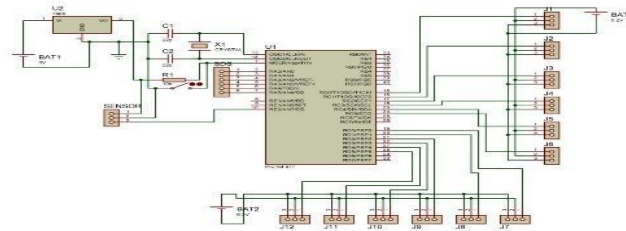


Figura 9. Tarjeta de control.

El shell diseñado para el microcontrolador facilita la conexión de los servomotores que consiste en 3 pines por cada uno, el primero es la señal de control proveniente del microcontrolador y los otros dos son la alimentación de la fuente de potencia, la cual consiste en una pila recargable de 6.0 volts y 3000 mA, además se utiliza una batería de 9 v para alimentar al microcontrolador y se tiene la conexión para un sensor de presencia.

El Shell del PIC permite la reprogramación de cualquier método de control al sistema (lo cual es uno de los fines del presente trabajo) pero para su demostración se le programo el método de acción básica de control denominado PID (proporcional, integrativo y derivativo) el cual se caracteriza por actuar ante la presencia de un error de manera rápida y buscando la exactitud del resultado.

Resultados: Como se puede observar en la figura 5, cada extremidad de la araña está formada por cuatro elementos, los primeros dos forman el antebrazo, el tercero es el brazo y el otro es un elemento de apoyo que permite utilizar una tercera articulación sin actuador para que la extremidad se recorte al levantarse (paralelograma deformable). El primer elemento mide 5cm. y en uno de sus extremos este ensamblado con el servomotor 1 para un movimiento horizontal, en el otro extremo tiene fijado al servomotor 2, para un movimiento vertical en el cual se ensambla al elemento 2 de 7 cm. Cuando estos dos elementos están alineados el antebrazo alcanza un tamaño horizontal de 12 cm. que es la distancia del cuerpo de la araña al punto de apoyo, la cual al relacionarse con el ángulo de movimiento del servomotor 1 de acuerdo a la ecuación (1) determina el avance de la pata.

El brazo (elemento 3) tiene un tamaño de 13 cm. que determina la altura que alcanza la araña con respecto al suelo, al estar ensamblada con el elemento 2 a través de la articulación 3, es levantado verticalmente, el cual al relacionarse con el ángulo del servomotor 2 de acuerdo a la ecuación (2) determina la altura de levantamiento de la pata.

Servomotor 1; movimiento de avance horizontal de las patas. Tomando $A+B=12\text{cm.}$ y la (1), tenemos;

Tabla 1. Distancia de avance de la araña.

α	Calculado	Real
0°	0 cm.	0 cm.
45°	9.18 cm.	9 cm.
60°	12 cm.	11.5 cm.
75°	14.6 cm.	14 cm.

Observaciones: Por aspectos físicos, 75° es el máximo avance a lograr, puede ser considerado también hacia atrás si se ubica el servomotor para trabajar de 0 a 180° y la referencia en 90° , es conveniente que todas las patas tengan el mismo valor de avance, de lo contrario la araña no seguirá una línea recta, α determina cambio de velocidad.

Servomotor 2; levantamiento vertical de las patas. De acuerdo a $B=7\text{cm.}$ y la $Ec2$, tenemos;

Tabla 2. Distancia de levantamiento de las patas.

α	Calculado	Real
0°	0 cm.	0 cm.
45°	4.94 cm.	4.6 cm.
60°	6.06 cm.	5.8 cm.
75°	6.76 cm.	6.4 cm.

Observaciones: Por aspecto físico, 75° es el máximo levantamiento a lograr. Se presenta una pequeña diferencia entre lo calculado con lo real debido al concepto de paralelograma deformable (0.36 cm) pero lo consideramos despreciable. Para el movimiento de la araña se requirió establecer una secuencia de movimientos tipo trípode que permitiera facilitar la programación. Método de pasos para el avance

- Primer paso; tomar una configuración de radial a bilateral, ver figura 11a y 11b, (fotografías de la araña) previa decisión de qué punto se establece como el frente (esta decisión puede cambiarse cuando la araña encuentre un obstáculo y en vez de dar la vuelta se puede cambiar el punto de frente).
- Segundo pasó, levantar las 2 patas extremos de la derecha y la pata central de la izquierda (grupo1) por medio de los servomotores 2 correspondientes, quedando sostenida como un tripié por las otras 3 patas (grupo2), ver figura 10 y 11c.

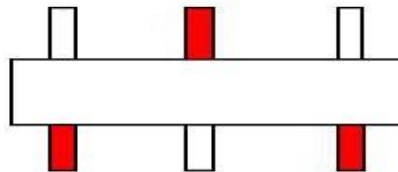


Figura 10. Trípede, en formato bilateral mostrando los 2 grupos a manipular.

- Tercer paso, aventar las patas del grupo1 un ángulo hacia adelante con los servomotores 1 correspondientes, bajarlas y recuperación de las referencias de sus servomotores 2.
- Cuarto paso, levantar las extremidades del grupo2 (con sus servomotores 2) y recuperar las referencias de los servomotores 1 del grupo1 provocando el avance, ver figura 11d.
- Quinto paso, las extremidades levantadas grupo2 son avanzadas un ángulo, para luego bajarlas.
- Nuevamente; levantar las extremidades del grupo1 (con sus servomotores 2) y se recuperan las referencias del grupo2 (servomotores 1) y continuamos repitiendo los pasos.

El control de toda esta secuencia está a cargo del microcontrolador PIC16F887A.

Todas las pruebas se realizaron en el laboratorio de electrónica del ITTG, considerando como obstáculos los cambios de nivel de las losas que forman el sendero al edificio I (Laboratorio de electrónica) que no sobrepasan los 5.5 cm. y el pasillo dentro del laboratorio de electrónica para hacer pruebas de giro, durante estas pruebas se realizaron diversos ajustes, como son:

- Debido a que los servomotores no son exactamente iguales se tuvo que ajustar el tiempo de los pulsos para cada uno de ellos con la intensidad que estos presentaran el mismo desplazamiento.
- La velocidad de respuestas tampoco es la misma por lo que entre un paso y otro de la secuencia se colocó un retardo que permitiera que todos los servomotores involucrados alcanzaran su valor final.
- Las patas derrapaban en el piso pulido, fue necesario ponerles tapones de hule

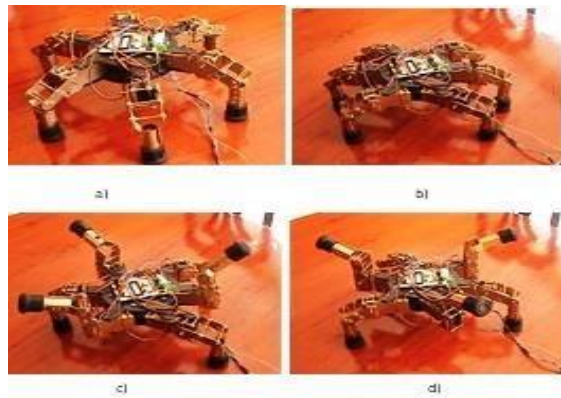


Figura 11. Fotografías de la araña en posición; a) radial, b) bilateral, c) Paso 2 y d) Paso 4.

Conclusiones.

Al final se logró satisfactoriamente el objetivo de este proyecto, que es tener una estructura rígida que tiene un movimiento constante y estable, que además es autónomo y que permite establecer libremente al programador el método de control que desee evaluar como son; control clásico, redes difusas, redes neuronales, entre otras lo cual lo convierte en un verdadero laboratorio de aplicación de técnicas de inteligencia artificial.

La tarjeta de control (shell) permite eliminar cruce de cables y el cambio rápido de cualquier servomotor que se dañe, debido al diseño de tres pines y a la conexión serial que presenta la fuente de potencia, ver figura 12, fotografía del shell diseñado y construido.



Figura 12. Fotografía del shell del PIC.

Se comprobó bajo la premisa de que el diseño de un robot tipo radial permite un cambio de dirección sin necesidad de desplazamiento y el tipo bilateral es más fácil de manipular para su avance, que la combinación de estos (un robot radial que se configura a formato bilateral) da como resultado obtener un desplazamiento optimo y un cambio de dirección fácil.

También pudimos comprobar que, en los casos de diseño y construcción de robots con articulaciones para su movimiento, la aplicación del método del paralelograma deformable permite el pliegue de las articulaciones durante su desplazamiento lo cual resulta en que se puede trabajar en un espacio reducido, sin necesidad de manipular otra articulación.

Por último, la aplicación del software Autocad Mechanical, permite el diseño y simulación de los sistemas electromecánicos, eliminando la necesidad de construir para probar, reduciendo de esta manera tiempo, dinero y esfuerzo.

Créditos.

Los autores de la presente investigación agradecen al Tecnológico Nacional de México campus Tuxtla Gutiérrez por las facilidades otorgadas para la realización de este trabajo y el apoyo proporcionado a través de sus alumnos de electrónica en diferentes modalidades de colaboración, como son residencias profesionales y servicios social.

Referencias Bibliográficas.

Manzanarez, I. R. A. R. (2017). *Análisis Cinemático de un robot metamórfico tipo hexápodo*, Tesis (Doctoral dissertation, instituto Politécnico Nacional).

Hidalgo Panchana, A. C. (2015). *Implementación de un prototipo de un robot móvil tipo hexápodo teledirigido para la contribución de la limpieza aplicado tecnología Open Source* (Bachelor's thesis, La Libertad: Universidad Estatal Península de Santa Elena, 2015).

Chamba, J., Sánchez, M., Moya, M., Noroña, J., & Franco, R. (2019) *Motion simulation of a hexapod robot in virtual reality environments*. Enfoque UTE, 10(1), 173-184.

Rivas, J. D. L. (2015). *Definición y análisis de los modos de marcha de un robot hexápodo para tareas de búsqueda y rescate* (Doctoral dissertation, Master's thesis, Escuela Superior de Ingeniería Industrial. Universidad Politécnica de Madrid).

Información de los autores.



Raúl Moreno Rincón, Ingeniero en Comunicaciones y Electrónica, egresado de la ESIME IPN, en la ciudad de México, D.F. Maestro en Ciencias en Ingeniería Electrónica egresado del Instituto Tecnológico de Toluca. Maestro en Educación Superior por la Universidad Autónoma de Chiapas. Certificado en SolidWorks Associate. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez y jefe de Proyectos de Docencia de Ingeniería Electrónica. Es investigador desde 1999 y Colabora en la línea de investigación “Robótica” de Ingeniería electrónica y es parte del cuerpo académico en formación “sistemas de control inteligentes”. Ha realizado proyectos como: Sistema de alarma para personas con deficiencia auditiva basado en XBEE, robot de cafetería, araña hexápoda, sistema de control de animatronic, entre otros.



Lizette Adriana Zebadúa Chavarría, es Ingeniera en Electrónica egresada del Instituto Tecnológico de Tuxtla Gutiérrez. Durante su etapa académica participo activamente en el Club de Robótica “Engineer Bots” del I.T.T.G., siendo parte del equipo de diseño

para los concursos del torneo de VEX ROBOTIC. Certificada en SolidWorks Associate. En la actualidad es alumno de tiempo completo de la Maestría en Ciencias de Mecatrónica del Tecnológico Nacional de México Campus Tuxtla Gutiérrez.



Álvaro Hernández Sol, es Ingeniero en electrónica, egresado del Instituto Tecnológico de Tuxtla Gutiérrez ITTG, en Tuxtla Gutiérrez, Chiapas. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez y es investigador desde 1997. Certificado en SolidWorks Associate. Jefe de proyectos de investigación de ingeniería electrónica desde el 2001. Fundó y asesora el “Club de robótica del ITTG”. Colabora en la línea de investigación de “Robótica” de Ingeniería electrónica. Dirige el área de trabajo en “Robótica” y es parte del cuerpo académico “sistemas de control inteligentes”. Ha realizado investigaciones en el área de los sistemas alternativos de comunicación y en sistemas robóticos, Así como en sistemas traductores de lenguaje.



José Ángel Zepeda Hernández, es Ingeniero Industrial en Eléctrica y Maestro en Ciencias en Ingeniería Mecatrónica, egresado del Instituto Tecnológico de Tuxtla Gutiérrez ITTG. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez e investigador desde 1999, Imparte cátedra en el área de Ingeniería Electrónica y la Maestría en Ciencias en Ingeniería Mecatrónica en el ITTG, donde imparte las asignaturas de Diseño con Transistores y Electrónica Básica respectivamente. Líder del cuerpo académico en formación “sistemas de control inteligentes”.



Rubén Herrera Galicia, obtuvo el título de doctor en ciencias técnicas con especialidad en electrónica por la Technical University of Warsaw, Polonia. Es profesor de tiempo completo adscrito al departamento de ingeniería eléctrica y electrónica del Tecnológico Nacional de México campus Tuxtla Gutiérrez. Es investigador desde 1985 y colabora en la línea de investigación de “Robótica” de Ingeniería electrónica. Es parte del cuerpo académico “sistemas de control inteligentes”.

Características morfológicas para el análisis automático del estilo de escritura.

Morphological characteristics for the automatic analysis of the writing style.

Germán Ríos Toledo* (1).
Tecnológico Nacional de México campus Tuxtla Gutiérrez.
german.rt@tuxtla.tecnm.mx.

Cesar Alejandro Meza Pérez (2). Estudiante del Tecnológico Nacional de México campus Tuxtla Gutiérrez,
L16270802@tuxtla.tecnm.mx.

Jonathan Velázquez Trinidad (3). Estudiante del Tecnológico Nacional de México campus Tuxtla Gutiérrez,
L16270802@tuxtla.tecnm.mx.

Héctor Guerra Crespo (4). Tecnológico Nacional de México campus Tuxtla Gutiérrez, hector.gc@tuxtla.tecnm.mx.

Galdino Belizario Nango Solís (5). Tecnológico Nacional de México campus Tuxtla Gutiérrez,
galdino.ns@tuxtla.tecnm.mx.

Aída Guillermina Cossio Martínez (6). Tecnológico Nacional de México campus Tuxtla Gutiérrez,
aida.cm@tuxtla.tecnm.mx.

*corresponding author.

Artículo recibido en noviembre 08, 2021; aceptado en diciembre 14, 2021.

Resumen.

En el análisis automático del estilo de escritura se requieren marcadores de estilo que, idealmente sean persistentes a factores como el tópico del documento, ya sea que se traten de correos electrónicos, artículos, novelas, cartas. Otro factor importante a considerar es la cantidad de texto presente en dichos documentos. Comúnmente, un análisis de estilo de escritura se inicia utilizando las palabras como un marcador de estilo. Sin embargo, es relativamente sencillo manipular las palabras que una persona elige para elaborar sus escritos, además de que el vocabulario de un texto es dependiente del tópico que en él se trate. Por ello, es necesario explorar marcadores de estilo más robustos, principalmente aquellos relacionados con la estructura morfológica o sintáctica de las oraciones. En este artículo se evalúa el uso de n-gramas de etiquetas POS en el análisis del cambio de estilo de escritura en novelas de diez autores de habla inglesa. Las novelas de cada autor se ordenaron de cronológicamente para establecer dos etapas, a las cuales se les denominó etapas inicial y final. El problema se abordó por medio de un algoritmo de aprendizaje automático supervisado. Los resultados mostraron que en algunos de los autores evaluados existe evidencia clara de un cambio en el estilo de escritura a través del tiempo.

Palabras clave: Marcadores de estilo, n-gramas, etiquetas POS, frecuencia, características, clasificación

Abstract.

In the automatic analysis of the writing style, style markers are required that are ideally persistent to factors such as the topic of the document, whether they are e-mails, articles, novels, letters. Another important factor to consider is

the amount of text present in such documents. Commonly, a writing style analysis begins by using the words as a style marker. However, it is relatively easy to manipulate the words that a person chooses to elaborate their writings, in addition to the fact that the vocabulary of a text is dependent on the topic in question. Therefore, it is necessary to explore more robust style markers, mainly those related to the morphological or syntactic structure of sentences. This article evaluates the use of POS tag n-grams in the analysis of writing style change in novels by ten English-speaking authors. The novels of each author were ordered chronologically to establish two stages, which were called the initial and final stages. The problem was addressed using a supervised machine learning algorithm. The results showed that in some of the evaluated authors there is clear evidence of a change in writing style over time.

Keywords: Style markers, n-grams, POS tags, frequency, features, classification

1. Introducción.

Una forma común de representar objetos es utilizando las características que los definen y sus valores. Suponga que se desea describir un automóvil, algunas de las características que lo definen son marca, modelo, tipo de transmisión, número de cilindros, tipo de combustible, etc. Este tipo de representación también es aplicable a textos o documentos. En este caso las características son: nombre del autor, título, fecha de publicación, género literario, editorial, precio, etc. La cantidad de características necesarias depende del nivel de descripción de cada objeto. En el Procesamiento del Lenguaje Natural existen tareas como la detección del plagio, creación de perfiles de autor, protección del anonimato y atribución de autoría, todas ellas enfocadas al análisis automático del estilo de escritura. La idea detrás de este tipo de análisis consiste en que, por medio de un conjunto de características preestablecidas, sea posible identificar quién es el autor de un texto entre una colección de autores. En términos computacionales, el estilo de escritura se refiere a la frecuencia de uso de elementos del texto, conocidos como marcadores de estilo o características estilométricas (Toledo, Sánchez, Sidorov, & Durán, 2019), (en lo sucesivo se les denomina características).

Para cuantificar el estilo de escritura, es fundamental determinar qué características estilométricas se van a utilizar para representar los textos, como son palabras más frecuentes, tipos de palabras utilizadas (verbos, sustantivos, adjetivos) o longitud de la oración. A través de los años, se han propuesto muchas características para el análisis de estilo de escritura. (Stamatatos, 2009) detalla una clasificación de características. Otros factores que juegan un papel importante en el desempeño del análisis del estilo de escritura son: número de autores candidatos, número de textos por autor y cantidad de texto en cada documento. Actualmente, el principal desafío en el análisis de estilo de escritura es identificar las características adecuadas para cada tarea en particular del Procesamiento del Lenguaje Natural.

Una forma simple y natural de ver el texto es como una secuencia de elementos (palabras, dígitos, signos de puntuación) agrupados en oraciones. Dado que, el análisis de estilo de escritura se centra en la forma del texto y no en su contenido, es importante explorar la funcionalidad de las características estilométricas que sean robustas o independientes al tema que aborda el texto.

La atribución de autoría consiste en determinar de entre un conjunto de autores, quién es el autor de un texto en particular (Juola, 2008). Desde el enfoque de aprendizaje automático supervisado, esta tarea se propone como un problema de clasificación. Más precisamente como una tarea de categorización de texto multiclase y de etiqueta única (Sebastiani, 2002).

Este artículo se enfoca en el estudio de una característica estilométrica conocida como *n-grama*. Un *n-grama* es una secuencia de elementos obtenidos siguiendo el orden lineal del texto. Los elementos pueden ser caracteres, palabras, etiquetas POS e información sintáctica. Un *n-grama* supone una ventana imaginaria de tamaño *n* que mueve *n* elementos en cada iteración, hasta llegar al final del texto (Sidorov, 2013). La tabla 1 muestra diferentes tipos de *n-gramas* para la oración “Juan lee un libro interesante” y $n = \{1,2,3,4\}$. Para el ejemplo, se ha omitido el espacio en blanco.

Tabla 1. N-gramas para la oración “Juan lee un libro interesante”.

<i>n</i>	n-gramas de carácter
1	J, u, a, n, l, e, e, u, n, l, i, b, r, o, i, n, t, e, r, e, s, a, n, t, e
2	Ju, ua, an, nl, le, ee, eu, un, nl, li, ib, br, ro, oi, in, nt, te, er, rs, sa, an, nt, te
3	Jua, anl, lee, eun, nli, ibr, roi, int, ter, rsa, ant
4	Juan, nlee, eunl, libr, roin, nter, resa, ante
<i>n</i>	n-gramas de palabras
1	Juan, lee, un, libro, interesante
2	Juan lee, lee un, un libro, libro interesante
3	Juan lee un, un libro interesante
4	Juan lee un libro
<i>n</i>	n-gramas de etiquetas POS
1	NP, VM, DI, NC, AQ
2	NP VM, VM DI, DI NC, NC AQ
3	NP VM DI, DI NC AQ
4	NP VM DI NC

A medida que aumenta el valor de *n*, la frecuencia de uso y la probabilidad de encontrar n-gramas comunes en diferentes documentos disminuye significativamente, estos problemas ocurren independientemente del tipo de n-grama. Según (Houvardas & Stamatos, 2006), la selección de un valor óptimo de *n* depende del idioma.

2. Métodos y materiales.

En trabajos anteriores (Toledo, Sánchez, Sidorov, & Durán, 2019) y (Gómez-Adorno, Posadas-Durán, Ríos-Toledo, Sidorov, & Sierra, 2018) se analizó el cambio en el estilo de escritura de autores a través del tiempo. El corpus está compuesto por novelas de 10 autores de habla inglesa, cada autor cuenta con 6 novelas. Las novelas se obtuvieron del sitio web de Gutenberg Project¹. El corpus se muestra en la Tabla 2.

Tabla 2. Corpus para el análisis del cambio de estilo de escritura.

Autor	Etapa Inicial		Etapa Final	
	Año	Novela	Año	Novela
Boot Tarkington(BT)	1899	Gentleman	1919	Ramsey
	1902	Vanrevels	1921	Alice Adams
	1905	Canaan	1922	Gentle Julia
Charles Dickens (CD)	1838	Nicholas Nickleby	1859	Two Cities
	1838	Oliver Twist	1861	Expectations
	1841	Barnaby	1865	Our mutual friend
Edgar Rice (ER)	1912	A Princess of Mars	1941	Llama of Gathol
	1914	The good of Mars	1942	Skeleton Men of Jupiter
	1918	The warlord of Mars	1944	Lasnd of terror
Frederick Marryat (FM)	1830	The King's Own	1845	The Mission
	1831	Jacob Faithful	1847	New Forrest
	1831	Newton Forster	1848	The Little Savage
George Macdonald (GM)	1863	David Elginbrod	1888	Electrical Lady
	1864	Adela	1891	Flight of Shadow
	1865	Alec Forbes	1892	Hope of gospel

¹ <https://www.gutenberg.org/>

George Vaizey (GV)	1901	School Story	1914	Cassandra
	1902	Pixie	1914	College Girl
	1902	Houseful of Girls	1915	claire
Iris Murdoch (IM)	1954	Under the net	1973	The black prince
	1956	The Flight from the Enchanter	1975	a word child
	1958	The Bell	1995	Jackson's Dilema
John Buchan	1910	Prester John	1932	The gap in the curtain
	1915	The thirty nine steps	1936	the island of sheeps
	1916	Green mantle	1941	sick heart river
Louis Tracy (LT)	1903	Wings of morning	1912	Romance of NY
	1904	The revelers	1916	The day of wrath
	1905	Disapperance	1919	Mortimer Fenley
Mark Twain (MT)	1869	Innocents Abroad	1897	the Equator
	1872	Roughing It	1905	What is man?
	1876	Tom Sawyer	1906	Dollar

2.1. Preprocesamiento de texto.

Las novelas se dividieron en oraciones con NLTK² (Kit de herramientas de Lenguaje Natural). Las oraciones de 1 y 2 palabras se eliminaron, ya que los 3-gramas de palabras requieren al menos 3 términos o tokens. Cada novela se convirtió a caracteres en minúsculas para lograr que cadenas de caracteres superficialmente diferentes tengan la misma forma (por ejemplo: car, Car, cars y Cars se podría normalizar a car). Los signos de puntuación se utilizaron para formar parte de los n-gramas de palabras. Además, cada texto se etiquetó con el etiquetador POS de NLTK³. La tabla 3 muestra el número de oraciones para todo el corpus.

Tabla 3. Oraciones compuestas por 3 o más términos.

Autor	Fecha/Novela	Oraciones	Autor	Fecha/Novela	Oraciones
Boot Tarkington	1899 Gentleman	5,380	George Vaizey	1901 School	3,520
	1902 Vanrevels	2,802		1902 Houseful	4,128
	1905 Canaan	4,598		1903 Pixic	4,549
	1919 Ramsey	2,179		1914 Cassandra	7,194
	1921 Alice Adams	5,602		1914 College	4,840
	1922 Julia	4,312		1915 Claire	6,112
Charles Dickens	1838 Nicholas	14,718	Iris Murdoch	1954 Under the net	1,702
	1838 Oliver	8,138		1956 The Flight from the Enchanter	2,029
	1841 Barnaby	11,480		1958 The Bell	1,531
	1859 Two cities	7,007		1973 The Black Pince	3,446
	1860 Expectations	8,979		1975 A word child	4,356
	1805 Mutual Friend	18,539		1995 Jackson's Dilema	1,689
Edgar Rice	1912 A Princess of Mars	5,688	John Buchan	1910 Prester John	6,158
	1914 The gods of mars	7,265		1915 The thirty nine steps	3,404
	1918 The warlord of mars	5,291		1916 green mantle	8,244
	1941 Llama of Gathol	7,274		1932 the gap in the curtain	6,838
	1942 Skeleton Men of Jupiter	1,883		1936 the island of sheeps	1,182
	1944 Land of Terror	1,431		1941 sick heart river	1,287
Frederick Marryat	1830 Kings Own	6,216	Louis Tracy	1903 Wings	6,234
	1832 Forster	5,944		1904 Revelers	6,480
	1834 Faithful	7,304		1905 Disappear	5,541
	1845 Mission	3,919		1912 Romance	4,106
	1847 New Forest	5,366		1916 Wrath	4,153
	1848 Little Savage	3,859		1919 Mottimer	4,894
George Macdonald	1863 Elginbrod	9,696	Mark Twain	1869 Innocents	8,270
	1864 Adela Volt	2,988		1872 Roughing It	6,772
	1865 Forbes	10,315		1876 Tom Sawyer	4,485

² <https://www.nltk.org>

³ <https://www.nltk.org/api/nltk.tag.html?highlight=stanford%20pos%20tagger>

1888 Elect Lady	3,493	1897 Equator	8,411
1891 Flight Shadow	3,868	1906 Dollar	4,519
1892 Cospel	2,086	1906 What is man?	4,842

Es recomendable realizar la normalización de la longitud del texto para asegurar que todos los autores tengan aproximadamente la misma cantidad de información (Singhal, Salton, & Buckley, 1995). Sin embargo, este trabajo, el análisis del estilo de escritura es de tipo intra-autor. Un análisis de estilo de escritura intra-autor evalúa los escritos del mismo autor a través del tiempo. Para una situación más realista, el número de oraciones no está normalizado.

Las novelas de cada autor se dividieron en 4 bloques con el mismo número de oraciones. Así, el número de muestras de texto aumenta, pero la cantidad de texto o palabras en cada una de ellas disminuye. En la tabla 4 se observa el número de oraciones en cada muestra de las novelas de Booth Tarkington.

Tabla 4. Oraciones en las novelas de Booth Tarkington.

Novela	Número de ejemplos			
	1	2	3	4
Canaan	4,598	2,299	1,532	1,149
Gentleman	5,350	2,675	1,783	1,337
Penrod	3,841	1,740	1,160	870
Seventeen	3,917	1,958	1,305	979
Turmoil	5,892	2,946	1,964	1,473
Vanrevels	5,802	1,401	934	700

2.2. Características Estilométricas.

Se generaron n-gramas de palabras y etiquetas POS utilizando el programa *text2ngram*⁴. Al crear n-gramas, especifica la cardinalidad de n-grama y la frecuencia de corte. En la Sección 1 se mencionó que comúnmente, la cardinalidad de un n-grama es $n = \{1,2,3,4\}$. En trabajos previos, sobre tareas de procesamiento de lenguaje natural, tales como, detección de plagio, atribución de autoría, categorización de texto e identificación de autores, se reporta que $n = \{3\}$ proporciona el mejor rendimiento (Houvardas & Stamatatos, 2006), (Escalante, Solorio, & Montes, 2011), (Sapkota, Solorio, Montes, Bethard, & Rosso, 2014), (Sidorov, Velasquez, Stamatatos, Gelbukh, & Chanona-Hernández, 2014) y (Zuo, Zhao, & Banerjee, 2019). Además de determinar la cardinalidad de un n-grama, se debe encontrar el valor de frecuencia de corte apropiado. Esto depende principalmente de 4 factores: idioma, número de textos (muestras), cantidad de texto en cada muestra (en palabras u oraciones) y la característica estilométrica seleccionada para el análisis. La frecuencia indica la cantidad de veces que aparece una característica estilométrica en el texto. En general, cuanto más frecuente es un rasgo, más variación estilística captura (Stamatatos, 2009). Las tablas 5, 6 y 7 muestran el número de características (palabras n-gramas) para $n = \{1,2,3\}$.

Tabla 5. Total de 1-gramas de palabras.

Autor	1	2	3	4
BT	6,598	5,133	4,400	3,995
CD	17,346	14,017	12,357	11,274
ER	6,395	5,215	4,585	4,215
FM	10,173	8,133	7,164	6,562
GM	8,695	6,785	5,879	5,301
GV	6,971	5,316	4,463	3,966
IM	10,236	8,185	7,199	6,550
JB	7,581	6,010	5,277	4,794

⁴ <https://helpmanual.io/man1/text2ngram/>

LT	7,681	5,783	4,921	4,335
MT	11,804	9,522	8,187	7,606

Tabla 6. Total de 2-gramas de palabras.

Autor	1	2	3	4
BT	8,850	6,187	4,966	4,279
CD	36,181	26,231	21,508	18,778
ER	10,170	7,354	6,090	5,326
FM	16,292	11,762	9,675	8,464
GM	13,672	9,693	7,951	6,828
GV	9,168	6,192	4,883	4,047
IM	18,908	13,483	11,013	9,588
JB	11,291	7,812	6,288	5,382
LT	8,749	5,837	4,582	3,850
MT	15,603	11,173	9,114	8,183

Tabla 7. Total de 3-gramas de palabras.

Autor	1	2	3	4
BT	2,613	1,569	1,134	949
CD	19,772	12,959	10,031	8,413
ER	4,554	2,984	2,293	1,899
FM	7,209	4,571	3,464	2,860
GM	5,046	3,084	2,299	1,868
GV	2,746	1,556	1,070	870
IM	8,649	5,426	4,103	3,455
JB	4,623	2,783	2,067	1,680
LT	2,427	1,366	961	717
MT	5,697	3,762	2,910	2,612

2.3. Matriz término-documento.

Los n-gramas y sus frecuencias se almacenan en la matriz denominada término-documento: las filas representan textos y las columnas el conjunto de n-gramas de un autor. La tabla 8, muestra una matriz término-documento de 1-gramas de palabras. Las matrices están ordenadas en orden descendente, la columna de la izquierda es el 1-grama que el autor usa con más frecuencia en sus textos.

Tabla 8. Las diez palabras más frecuentes de 1-grama para Booth Tarkington.

Ejemplos	the	and	to	of	a	he	in	was	his	i
<i>Texto 1</i>	2,757	1,342	993	1,146	1,057	803	680	568	552	371
<i>Texto 2</i>	2,663	1,630	1,249	1,170	1,162	701	709	598	548	483
<i>Texto 3</i>	1,890	828	750	880	700	452	523	512	341	144
<i>Texto 4</i>	1,460	739	708	569	531	428	375	290	325	259
<i>Texto 5</i>	1,921	869	932	989	807	646	516	442	475	318
<i>Texto 6</i>	2,043	1,038	996	917	744	666	509	467	484	381
<i>Texto 7</i>	903	597	534	512	404	377	288	265	236	179
<i>Texto 8</i>	624	528	535	371	329	195	218	206	111	282
<i>Texto 9</i>	1,313	866	1,150	803	829	469	471	375	286	448

<i>Texto 10</i>	1,177	860	1,177	643	731	714	389	344	328	565
<i>Texto 11</i>	1,388	965	912	797	752	601	489	452	335	335
<i>Texto 12</i>	1,107	749	871	680	572	453	384	340	253	376

2.4. Pruebas de clasificación.

A continuación se describen los conjuntos de entrenamiento y prueba. La Tabla 9 muestra las novelas de Booth Tarkington.

Tabla 9. Novelas de Booth Tarkington.

Etapa Inicial		Etapa Final	
Novela	Año	Novela	Año
Gentleman	1899	Ramsey	1919
Vanrevels	1902	Alice Adams	1921
Canaan	1905	Julia	1922

Los datos se dividieron en conjuntos de entrenamiento y prueba utilizando la estrategia Leave-One-Out: una novela por clase se usa una vez como conjunto de prueba y el resto para el conjunto de entrenamiento. Así, se crearon nueve tuplas de entrenamiento y prueba por autor. La tabla 10, muestra los conjuntos de prueba y entrenamiento para Booth Tarkington. En cada una de las nueve iteraciones, cada novelas solo pertenece a uno de los dos conjuntos.

Tabla 10. Conjuntos de prueba y entrenamiento de Booth Tarkington.

Conjunto de Prueba	Conjunto de Entrenamiento
Gentleman, Ramsey	VanRevels, Alice Adams, Canaan, Julia
Gentleman, Alice Adams	VanRevels, Ramsey, Canaan, Julia
Gentleman, Julia	VanRevels, Ramsey, Canaan, Alice Adams
VanRevels, Ramsey	Getnleman, Alice Adams, Canaan, Julia
VanRevels, Alice Adams	Getnleman, Ramsey, Canaan, Julia
VanRevels, Julia	Getnleman, Ramsey, Canaan, Ramsey
Canaan, Ramsey	Getnleman, Vanrevels, Alice Adams, Julia
Canaan, Alice Adams	Getnleman, Vanrevels, Ramsey, Julia
Canaan, Julia	Getnleman, Vanrevels, Ramsey, Alice Adams

La distribución de los conjuntos de prueba y entrenamiento se muestran en la tabla 11, una proporción de 1/3($\approx$ 33%) para pruebas y 2/3 ($\approx$ 67%) para entrenamiento.

Tabla 11. Distribución para conjuntos de entrenamiento y prueba.

Tamaño	Ejemplos	Ejemplos para prueba	Ejemplos para entrenamiento
1	6	2	4
2	12	4	8
3	18	6	12
4	24	8	16
5	30	10	20

El problema se abordó como una atribución de autoría supervisada: dado un documento D y dos etapas $S = \{Inicial, Final\}$ para un autor único, determinar a cuál de las dos etapas en S, D pertenece. Este es un problema de clasificación binaria, la clase positiva está etiquetada con la etiqueta inicial. El clasificador binario predice las instancias del conjunto de pruebas como positivas o negativas y produce cuatro resultados: Verdadero Positivo (TP), Verdadero Negativo (TN), Falso Negativo (FN) y Falso Positivo (FP).

Las pruebas de clasificación se realizaron con el algoritmo de Aprendizaje Automático Supervisado Regresión Logística implementado en scikit-learn. La métrica de precisión no es una buena opción cuando hay el desbalance de clases. Sin embargo, en estos experimentos ambas clases están equilibradas (ver tabla 10), por lo que la precisión (accuracy) resulta apropiada para la evaluación (García, Mollineda, & Sánchez, 2009). La precisión es la fracción de predicciones que el modelo hizo correctamente. Su representación matemática se muestra en la siguiente ecuación.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Se obtuvo el promedio de nueve experimentos para los diferentes tamaños de texto. La línea de base es un modelo aleatorio hipotético con una 50% de precisión, la cual representa la probabilidad de asignar correctamente una muestra a una de las dos etapas.

3. Resultados.

3.1. n-gramas de palabras.

En las siguientes tablas, las etiquetas significan: **1h**= 100 n-gramas más frecuentes, **2h**= 200 n-gramas más frecuentes y así sucesivamente. La leyenda **todo** significa que se están utilizando todos los n-gramas del conjunto.

La tabla 12 muestra el total de n-gramas de palabras para $n = \{1,2,3,4\}$. A medida que el texto se divide en bloques más pequeños, la cantidad de características tiende a disminuir. Por ejemplo, para el autor ER en 2-gramas de palabras pasa de 10,170 en novela completa (1) a 5,326 en cuartos de novela (4).

Tabla 12. Total de características de n-gramas de palabras.

Autor	1-gramas de palabras				2-gramas de palabras				3-gramas de palabras			
	1	2	3	4	1	2	3	4	1	2	3	4
BT	6,598	5,133	4,400	3,995	8,850	6,187	4,966	4,279	2,613	1,569	1,134	949
CD	17,346	14,017	12,357	11,274	36,181	26,231	21,508	18,778	19,772	12,959	10,031	8,413
ER	6,395	5,215	4,585	4,215	10,170	7,354	6,090	5,326	4,554	2,984	2,293	1,899
FM	10,173	8,133	7,164	6,562	16,292	11,762	9,675	8,464	7,209	4,571	3,464	2,860
GM	8,695	6,785	5,879	5,301	13,672	9,693	7,951	6,828	5,046	3,084	2,299	1,868
GV	6,971	5,316	4,463	3,966	9,168	6,192	4,883	4,047	2,746	1,556	1,070	870
IM	10,236	8,185	7,199	6,550	18,908	13,483	11,013	9,588	8,649	5,426	4,103	3,455
JB	7,581	6,010	5,277	4,794	11,291	7,812	6,288	5,382	4,623	2,783	2,067	1,680
LT	7,681	5,783	4,921	4,335	8,749	5,837	4,582	3,850	2,427	1,366	961	717
MT	11,804	9,522	8,187	7,606	15,603	11,173	9,114	8,183	5,697	3,762	2,910	2,612

La tabla 13, muestra la precisión con 1-gramas de palabras. La precisión más alta es para los autores BT, ER y JB que promedian al menos el 95%, seguidos de LT y MT con resultados que oscilan entre el 73 y el 82%. Los autores restantes apenas logran un 68% de precisión. En este sentido, GM es el autor con menor precisión con 50% en novelas completas (1).

Tabla 13. Precisión con 1-grama de palabras.

Autor	1					2					3					4				
	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo
BT	100	100	100	100	100	100	100	97	100	94	100	98	98	91	94	90	86	83	83	96
CD	67	67	67	67	61	61	61	61	61	61	63	63	63	63	63	64	65	65	65	65
ER	100	100	100	100	100	94	94	92	92	94	96	96	98	94	98	96	96	96	96	94
FM	56	56	56	56	56	58	58	58	58	58	61	61	65	63	61	64	64	65	65	65
GM	44	50	50	50	50	64	64	64	67	64	67	69	70	72	70	61	61	58	60	62
GV	67	72	67	67	67	78	96	67	67	64	74	74	70	69	70	61	68	67	71	68
IM	67	67	67	67	67	64	67	67	67	67	67	67	63	67	65	68	68	68	68	68
JB	100	100	100	100	100	100	97	97	100	97	96	96	94	96	96	93	93	93	93	93
LT	83	83	83	83	83	92	92	92	89	86	89	89	89	89	87	90	90	90	89	89
MT	72	72	72	72	67	75	75	75	75	75	67	69	67	67	67	79	79	79	78	74

La tabla 14, muestra la precisión con 2-gramas de palabras. Al igual que en experimentos anteriores, los autores BT, ER y JB logran hasta un 100% de precisión en novelas completas (1), así como en medias novelas (2). Contrariamente a 1-gramas de palabras, la precisión de los autores CD, GV e IM mejora para superar el 70%, incluso el 90% en tercios (3) y cuartos (4) de novelas. Con 2-gramas de palabras, el autor GM muestra la precisión más baja en los diferentes tamaños de novelas e independientemente del número de características utilizadas. Se destaca que la precisión del autor de LT disminuye de manera significativa a una media del 72% en los diferentes tamaños de novelas, resultado que dista mucho de 1-gramas de palabras de 90%. Por el contrario, el autor MT, mejoró su precisión en novelas completas (1), pasando del 72% en 1-gramas de palabras a 83% en 2-gramas de palabras. En los tamaños restantes de la novela, este autor también mostró una disminución en la precisión. Contrariamente a lo esperado, usar todas las características (**todo**) no garantiza una mejor precisión. Todos los autores muestran una disminución al usar todas las características con respecto a los n-gramas más frecuentes.

Tabla 14. Precisión con 2-gramas de palabras.

Autor	1					2					3					4				
	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo
BT	100	100	100	100	100	94	97	97	97	86	93	94	94	94	89	96	96	94	93	89
CD	83	83	83	83	83	83	83	83	83	83	72	67	69	61	83	67	72	68	65	79
ER	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	99	100	100	99
FM	56	61	61	61	56	58	64	58	58	53	69	70	63	63	61	72	69	71	72	70
GM	50	56	50	56	56	47	50	50	50	53	48	52	54	54	54	51	53	54	54	54
GV	56	72	61	61	50	92	92	92	92	92	96	96	98	98	93	96	93	94	93	90
IM	72	78	78	78	67	61	72	75	75	75	65	67	69	69	72	64	69	71	69	67
JB	89	100	100	100	100	100	100	100	100	100	93	94	94	94	94	94	92	94	92	96
LT	78	72	72	72	67	67	67	64	64	67	80	74	76	76	72	81	72	74	74	74
MT	83	83	83	83	83	67	67	67	67	67	56	57	57	63	56	60	61	62	57	58

La tabla 15 muestra la precisión usando 3-gramas de palabras. La precisión disminuyó con respecto 1-gramas y 2-gramas de palabras (Ver tablas 13 y 14). La precisión de BT disminuyó aproximadamente un 10% en tercios de novelas (3) y hasta un 20% en cuartos de novela (4). En general, para el autor ER la precisión disminuyó un 20%, además de que en cuartos de novela (4) no superó el 70%. Sin embargo, la precisión del autor JB se mantuvo constante en todos los tamaños de novela, con una precisión del 95 al 100%. Así mismo, el autor GV mostró resultados constantes cercanos al 80%. Un valor atípico es el autor LT: la precisión con palabras de 3-gramas es superior a las palabras de 2-gramas hasta en un 10% con novelas completas (1) y medias novelas (2). Por otro lado, la precisión con 3-gramas de palabras es menor que la de 2-gramas de palabras hasta en un 20% en tercios y cuartos de novelas.

Tabla 15. Precisión con 3-gramas de palabras.

Autor	1					2					3					4				
	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo
BT	100	94	94	89	83	94	94	92	92	81	83	83	87	85	81	75	76	76	75	72
CD	61	61	61	67	89	56	61	64	64	75	57	61	67	65	69	56	67	65	69	71
ER	50	72	83	83	83	64	69	69	67	72	67	72	69	70	67	67	67	67	64	64
FM	78	72	67	72	78	69	67	69	69	75	69	67	70	70	67	65	65	65	65	64
GM	67	72	72	72	61	64	64	64	58	58	63	59	67	56	57	57	57	56	57	61
GV	83	83	78	78	83	78	78	78	78	81	80	80	80	80	78	78	81	81	79	79
IM	72	67	67	67	72	78	78	75	75	69	76	76	76	76	69	71	75	74	72	71
JB	100	100	100	100	100	100	100	100	100	100	96	96	94	94	96	96	94	94	94	94
LT	94	89	83	78	56	78	75	67	58	53	56	61	57	57	59	60	56	57	57	54
MT	67	61	61	61	67	56	53	53	53	56	52	56	56	54	54	53	56	57	57	56

3.2. n-gramas de etiquetas POS

La tabla 16, muestra el número total de n-gramas de etiquetas POS para $n = \{1,2,3,4\}$. El número de características disminuye a medida que el texto se divide en bloques más pequeños. Cabe hacer notar que en 1-gramas de etiquetas POS hay solo 33 o 34 características, ya que son las que conforman el conjunto estándar de etiquetas POS: CC=Conjunción coordinadora, CD=número cardinal, DT= determinante, JJ=adjetivo, etc. Para conocer el número de etiquetas y su significado, consulte el proyecto Penn Treebank⁵

Tabla 16. Total de características de n-gramas de etiquetas POS.

Autor	POS 1 - gramas				POS 2 - gramas				POS 3 - gramas			
	1	2	3	4	1	2	3	4	1	2	3	4
BT	33	33	33	33	736	668	640	617	4,606	3,844	3,412	3,147
CD	34	34	34	34	859	823	793	775	7,629	6,646	6,100	5,691
ER	34	34	34	34	658	611	578	567	3,983	3,356	2,984	2,767
FM	33	33	33	33	743	702	680	663	5,612	4,795	4,327	4,031
GM	33	33	33	33	748	712	695	661	5,428	4,540	4,134	3,853
GV	33	33	33	33	697	647	616	602	4,594	3,824	3,431	3,145
IM	35	35	35	35	756	714	693	674	5,719	4,889	4,428	4,148
JB	34	34	34	34	691	636	598	572	4,353	3,679	3,277	3,039
LT	34	34	34	34	693	644	620	597	4,265	3,541	3,204	2,898
MT	34	34	34	34	735	704	674	655	5,607	4,859	4,410	4,157

Debido al reducido número de características en 1-gramas de etiquetas POS, solo se realizaron experimentos con todas las características. La tabla 17, muestra que los autores ER y JB logran una precisión de hasta el 100%, seguidos por los autores BT, FM e IM con una precisión que va del 80 al 90%. Por otro lado, el autor CD apenas alcanza el 61% de precisión en novelas completas (1) y medias novelas (2).

⁵ https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

Tabla 17. Precisión con 1-gramas de etiquetas POS y todas las características.

Autor	1	2	3	4
BT	86	94	87	96
CD	61	61	70	68
ER	100	100	100	100
FM	89	92	89	92
GM	72	50	41	44
GV	67	72	69	71
IM	89	78	81	78
JB	100	100	93	93
LT	72	75	70	67
MT	67	78	72	71

La tabla 18 muestra la precisión con 2-gramas de etiquetas POS. Se observa una tendencia similar a los experimentos con 1-gramas de etiquetas POS: los autores ER, JB, BT e IM son los mejores clasificados, seguidos de CD, FM y MT. Cabe señalar que los autores de GM y LT tienen la precisión más baja, independientemente del número de características y tamaños de la muestra.

Tabla 18. Precisión con 2-gramas de etiquetas POS.

Autor	1					2					3					4				
	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo
BT	89	89	89	89	89	97	97	100	100	97	94	94	94	94	94	92	90	92	93	88
CD	72	72	72	72	72	75	75	75	75	78	85	83	83	83	83	76	76	75	75	75
ER	100	100	100	100	100	100	100	100	97	100	94	98	96	98	100	94	96	96	96	96
FM	78	78	78	78	78	83	83	83	83	83	83	83	83	83	83	85	85	85	85	85
GM	67	67	67	67	67	42	42	42	42	42	50	52	52	52	52	44	43	43	43	43
GV	61	61	61	61	61	69	69	69	69	69	70	69	69	69	69	72	74	74	74	74
IM	83	83	83	83	83	69	69	69	69	69	78	78	78	78	78	72	72	72	72	72
JB	100	100	100	100	100	100	100	100	100	100	91	91	89	91	91	96	93	93	93	93
LT	56	56	56	56	56	67	67	67	67	67	65	65	65	65	65	71	69	69	69	69
MT	72	72	72	72	72	78	78	78	78	78	70	72	70	70	70	67	68	68	68	68

La tabla 19 muestra la precisión con 3-gramas de etiquetas POS. Persiste la tendencia en todos los autores: El autor GM es el peor clasificado ya que en promedio no supera el 55% de precisión. Sin embargo, en estos experimentos el autor LT supera el 70% al utilizar novelas completas (1) y medias novelas (2). Otro punto a destacar es que el autor GV pasó de 74% en 2-gramas de etiquetas POS a un máximo de 92% en cuartos de novelas (4) 3-gramas de etiquetas POS.

Tabla 19. Precisión con 3-gramas de etiquetas POS.

Autor	1					2					3					4				
	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo	1h	2h	3h	4h	Todo
BT	89	94	100	100	100	89	92	97	97	94	87	89	89	89	89	81	81	86	86	85
CD	78	72	78	72	72	92	94	94	92	89	94	96	93	91	91	93	93	92	92	88
ER	100	100	100	100	100	92	92	94	92	97	94	94	94	94	94	94	93	93	93	93
FM	78	78	78	78	78	81	81	81	81	81	83	83	83	83	81	82	82	82	83	83
GM	50	56	56	56	61	47	47	47	50	44	50	54	59	54	57	50	51	49	56	57
GV	61	61	61	61	61	67	67	67	67	67	85	74	70	80	72	90	92	90	83	83
IM	89	89	89	89	89	69	72	72	72	72	76	78	76	74	76	76	74	74	74	74
JB	100	100	100	100	100	92	94	94	94	100	93	85	85	89	81	90	89	92	90	89
LT	72	78	67	67	67	72	75	72	72	72	59	67	67	63	65	64	69	62	74	69
MT	83	83	78	78	78	83	78	81	78	78	76	76	76	69	74	67	68	72	72	65

Discusión.

La forma natural de evaluar el estilo de escritura de un autor es con las palabras que utiliza. Dichas palabras conforman el vocabulario del autor. De esta forma se conoce la frecuencia de uso: aquellas que usa con más frecuencia, o con menos frecuencia, palabras que usa una sola vez o dos veces, entre otras. Sin embargo, un análisis de estilo de escritura basado únicamente en la frecuencia de uso de palabras no es confiable, ya que, en cierto sentido para un autor experimentado, es muy simple manipular las palabras de un texto. Por otro lado, las palabras por si solas aportan indicios del tópico del documento. En consecuencia, en lugar de realizar un análisis de estilo se estaría realizando un análisis de los temas sobre los cuales un autor escribe regularmente.

Todo análisis de estilo de escritura parte de la selección de características estilométricas (o marcadores de estilo). Algunos ejemplos de marcadores de estilo son los caracteres del alfabeto, signos de puntuación, palabras, etiquetas POS entre otros. Un escritor experto dispone, además de las palabras, de otros recursos del lenguaje que se encuentran en el nivel sintáctico y semántico. En el nivel semántico existen las categorías de las palabras: sinónimo, antónimo, homónimo, homófonos), catáforas, anáforas, etc. En el nivel sintáctico dispone de oraciones simples y compuestas, oraciones activas y pasivas, oraciones principales y subordinadas, etc.

A partir de los dos niveles descritos previamente, es posible generar otros tipos de marcadores de estilo, como por ejemplo n-gramas de palabras, n-gramas de etiquetas POS, n-gramas sintácticos. De estos tres tipos n-gramas se generan combinaciones de ellos: palabras-etiquetas POS, etiquetas POS-relaciones de dependencia, palabras-relaciones de dependencia y así sucesivamente. Resulta evidente que diferencia de las palabras, la manipulación consciente de este tipo de marcadores de estilo resulta compleja incluso al autor más experimentado. En consecuencia, estos marcadores permiten análisis de estilo mucho más confiables. Idealmente, se investigan marcadores que identifiquen a un autor sin importar el tipo de documento que este escriba y el tema que aborde.

Para obtener marcadores de estilo robustos, se requieren herramientas computacionales que analizan en lenguaje a nivel de su estructura sintáctica, específicamente a nivel de oraciones. Dicha estructura contiene información muy valiosa sobre la forma en que el autor compone sus oraciones. Eventualmente la estructura (que de forma inconsciente) de las oraciones revela patrones de construcción que no son perceptibles al nivel de los caracteres y las palabras.

Como ejemplos de las herramientas de software para análisis sintáctico se tienen: Stanford Parser, Spacy y Stanza. Sin embargo, es importante recordar que, a pesar de ser herramientas de última generación, incluso entrenadas por medio de aprendizaje profundo, presentan un margen de error que es importante considerar. Particularmente, dichas herramientas presentan detalles en oraciones ambiguas. O bien, que los algoritmos para determinar dónde inicia y donde termina una oración son diferentes entre ellas.

Conclusión.

Se analizaron novelas de 10 autores de habla inglesa; dichas novelas se ordenaron de forma cronológica de acuerdo con la fecha de publicación y se definieron 2 etapas cada una con 3 novelas. Las novelas fueron divididas en oraciones; posteriormente cada novela se dividió en 4 partes proporcionales de acuerdo a la cantidad de oraciones. Una vez generados los textos con distinto número de oraciones, se obtuvieron n-gramas de palabras y etiquetas POS con valores para $n = \{1, 2, 3\}$ con sus respectivas frecuencias de uso. Los n-gramas y sus frecuencias se almacenaron en matrices de dos dimensiones (Modelo Espacio Vectorial). Una vez que los datos están en las matrices, los n-gramas se ordenaron de forma descendente: de los n-gramas más frecuentemente utilizados a los menos frecuentes.

Dichas matrices se utilizaron para entrenar un algoritmo de clasificación supervisada conocido como Regresión Logística. La estrategia para el entrenamiento fue los k marcadores utilizados con mayor frecuencia donde $k = \{100, 200, 300, 400, 500\}$. Adicionalmente se realizaron experimentos con todas las características o marcadores de estilo disponibles en cada configuración.

La etapa del aprendizaje tiene la finalidad de crear un modelo predictivo que sea capaz de determinar a cuál de las dos etapas de un autor pertenece un texto de ejemplo. La métrica para evaluar la eficiencia de los dos tipos de n-gramas y longitudes fue la exactitud (accuracy).

De forma general se encontró que al seleccionar los 100 o 200 n-gramas utilizados con más frecuencia, se obtiene una exactitud que supera a los valores restantes de k . Esta tendencia se observó en los n-gramas de palabras y de etiquetas POS así como en los distintos valores de n . Utilizar todas las características no presentó ninguna mejora significativa con respecto a los valores de k .

Por otro lado, la idea de dividir las novelas en partes es proporcionales tiene dos objetivos: el primero es aumentar de forma artificial el número de textos de cada autor. El segundo es para observar qué ocurre cuando en los textos hay menos cantidad de información: novelas completas (1), medias novelas (2), tercios de novelas (3) y cuartos de novelas (4). Cuando se evalúan novelas completas (1) cada autor cuenta con 6 ejemplos. Mientras que al evaluar cuartos de novelas (4) disponen de 24. Es evidente que a menor cantidad de texto menor es el valor de la métrica exactitud. De acuerdo con los resultados de los experimentos, se sugiere dividir los textos en cuatro bloques de distinto tamaño para observar claramente la tendencia registrada por un autor ante los dos tipos de n-gramas y distintos valores de n .

Finalmente, el cambio de estilo de escritura en algunos autores es más evidente que en otros. En n-gramas de palabras, los autores con la tasa de clasificación más alta fueron Booth Tarkington (BT), Edgar Rice (ER) y John Buchan (JB). La misma tendencia se observó en n-gramas de etiquetas POS. En el otro extremo se encuentran los autores Frederick Marryat (FM) y George Macdonald (GM) particularmente en 1-gramas y 2-gramas de palabras. El mismo caso ocurrió para George Vaisey (GV) pero en n-gramas de etiquetas POS en novelas completas (1) y en medias novelas (2).

Créditos.

Los autores agradecen al Tecnológico Nacional de México por el financiamiento del proyecto 10849.21-P, de la convocatoria de apoyo a proyectos de desarrollo tecnológico e innovación 2021 y las facilidades del Tecnológico Nacional de México campus Tuxtla Gutiérrez para la realización de este trabajo.

Referencias bibliográficas.

- Escalante, H., Solorio, T., & Montes, M. (2011).** *Local histograms of character n-grams for authorship attribution. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 288-298.
- García, V., Mollineda, R., & Sánchez, J. (2009).** Index of balanced accuracy: A performance measure for skewed class distributions. *Iberian Conference on Pattern Recognition and Image Analysis* (págs. 441-448). Springer.
- Gómez-Adorno, H., Posadas-Durán, J., Ríos-Toledo, G., Sidorov, G., & Sierra, G. (2018).** Stylometry-based approach for detecting writing style changes in literary texts. *Computación y Sistemas* 22(1), 47-53.
- Houvardas, J., & Stamatatos, E. (2006).** N-gram feature selection for authorship Identification. *International Conference on Artificial Intelligence: Methodology, Systems, and Applications* (págs. 77-86). Springer.
- Juola, P. (2008).** *Authorship Attribution* vol. 3. Now Publishers Inc.

- Sapkota, U., Solorio, T., Montes, M., Bethard, S., & Rosso, P. (2014).** Crosstopic authorship attribution: Will out-of-topic data help? *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 1228-1237.
- Sebastiani, F. (2002).** Machine learning in automated text categorization. *ACM computing surveys (CSUR)* 34(1), 1-47.
- Sidorov, G. (2013).** México D.F.: Sociedad Mexicana de Inteligencia Artificial.
- Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., & Chanona-Hernández, L. (2014).** Syntactic n-grams as machine learning features for natural language processing. *Expert Systems with Applications* 41(3), 853-860.
- Singhal, A., Salton, G., & Buckley, C. (1995).** Length normalization in degraded text collections. *Technical report, Cornell University*.
- Stamatatos, E. (2009).** A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology* 60 (3), 538-556.
- Toledo, G. R., Sánchez, N., Sidorov, G., & Durán, J. (2019).** Identificación de cambios en el estilo de escritura literaria con aprendizaje automático. *Onomázein: Revista de Lingüística, filología y traducción de la Pontificia Universidad Católica de Chile* (46), 102-128.
- Zuo, C., Zhao, Y., & Banerjee, R. (2019).** Style change detection with feed-forward neural networks. *CLEF (Working Notes)*.

Información de los autores.



Germán Ríos Toledo obtuvo el grado de Doctor en Ciencias de la Computación en 2019 por el Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) en Cuernavaca, Morelos, México. Actualmente, es profesor de tiempo completo en el Departamento de Computación del Tecnológico Nacional de México (campus Tuxtla Gutiérrez, Chiapas) en la Ingeniería en Sistemas Computacionales y en la Maestría en Ciencias en Ingeniería Mecatrónica. Su área de especialización es el Procesamiento del Lenguaje Natural, particularmente en el uso de información sintáctica como una característica para el análisis de estilo de escritura. Otras áreas de su interés incluyen el procesamiento y análisis de imágenes, audio y video por medio de Algoritmos de Aprendizaje Automático y Aprendizaje Profundo. Miembro del Sistema Nacional de Investigadores, nivel C (2024-2021).



Cesar Alejandro Meza Pérez, estudiante de Ingeniería en Sistemas Computacionales en el Instituto Tecnológico de Tuxtla Gutiérrez, sus áreas de interés se relacionan con las ciencias de la computación, desarrollo de software y desarrollo de aplicaciones móviles.



Jonathan Velázquez Trinidad, estudiante de ingeniería en sistemas computacionales en el Instituto Tecnológico de Tuxtla Gutiérrez, sus áreas de interés se relacionan con base de datos, telecomunicaciones, redes de computadoras y multimedia.



Héctor Guerra Crespo es egresado del I. T. de Mérida (Yucatán, México) de la carrera de Ingeniería en Sistemas Computacionales en 1994, es Doctor en Sistemas Computacionales por la Universidad del Sur (Chiapas, México) en 2011. Es profesor en el área de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Tuxtla Gutiérrez y en el área de Licenciatura en Sistemas Computacionales de la Universidad Autónoma de Chiapas, en ambas desde 1995. Es miembro del Claustro Doctoral "Doctorado en Ciencias de la Ingeniería" perteneciente al Programa Nacional de Posgrados de Calidad, I.T. de Tuxtla Gutiérrez desde 2016. Miembro del Sistema Nacional de Investigadores, nivel C (2024-2021).



Galdino Belizario Nango Solís, Es Ingeniero en Sistemas Computacionales por el Instituto Tecnológico de Tuxtla Gutiérrez, Chiapas, en 1996. Es Maestro en Ciencias de la Computación por el Centro de Investigación en Computación del Instituto Politécnico Nacional, en 2001. Es Doctor en Desarrollo Tecnológico por la Universidad de Ciencia y Tecnología Descartes, en 2016. Es profesor con 30 Hrs. en el área de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Tuxtla Gutiérrez, donde imparte clases en las materias de: Programación Lógica y funcional; y Lenguajes y Autómatas. Actualmente es presidente de la Academia de ISC en el ITTG.



Aída Guillermina Cossío Martínez es Maestra en Ciencias en Administración por el Instituto Tecnológico de Tuxtla Gutiérrez en 2002. Es profesora de tiempo completo del área de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Tuxtla Gutiérrez, desde 1994. Se especializa en la formulación y evaluación de proyectos, así como el emprendimiento y desarrollo de planes de negocio, actualmente es perfil deseable y trabaja en la línea de investigación Tecnología de Información y Base de Datos.

Diseño de prototipo de direccionamiento de robot tipo Ackerman aplicado a la desinfección de COVID.

Ackerman-type robot addressing prototype design applied to COVID disinfection.

José Luis Hernández Ameca* (1).

Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la computación, Laboratorio de Sistemas Robóticos (SIRO).

joseluis.hdzameca@correo.buap.mx.

María del Carmen Báez Salazar (2). Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Laboratorio de Sistemas Robóticos (SIRO), maria.baezs@alumno.buap.mx.

Luis Enrique Colmenares Guillen (3). Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Laboratorio de Análisis Forense Digital (LAFD), lecolme@gmail.com.

Elsa Chavira Martínez (4). Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, elsachavira56@hotmail.com.

*corresponding author.

Artículo recibido en diciembre 17, 2020; aceptado en febrero 16, 2021.

Resumen.

En este trabajo se presentan las fases del proceso de diseño de un sistema de direccionamiento tipo Ackerman, que en un futuro formará parte de un robot móvil para desinfección de espacios físicos, a manera de apoyo en situaciones como la que se vive internacionalmente a causa del virus SARS-CoV-2. En la introducción de este trabajo se contextualiza al lector sobre la importancia de los robots de limpieza en situaciones de contingencia sanitaria y se describe el sistema Ackerman. En el método se muestran las etapas que se han seguido para el desarrollo del prototipo. En el desarrollo se inicia con la definición de la idea, se muestra el dibujo del chasis realizado en dos dimensiones (2D), el modelado en tres dimensiones (3D) de las piezas que conforman el sistema Ackerman, ensamblaje de piezas creadas, estudio de movimiento y detección de interferencias del prototipo. En la última sección se presentan los resultados, conclusiones y se propone como trabajo futuro imprimir, ensamblar y probar el modelo del sistema de direccionamiento Ackerman, así como continuar desarrollando los sistemas de control y potencia del robot de desinfección.

Palabras clave: Sistema de Direccionamiento Ackerman, Diseño Asistido por Computadora, Desinfección, COVID-19.

Abstract.

This work presents the phases of the design process of an Ackerman-type addressing system, which in the future will be part of a mobile robot for disinfection of physical spaces, as a support in situations such as the one experienced internationally because of the SARS-CoV-2 virus. In the introduction to this work, the reader is contextualized about the importance of cleaning robots in health contingency situations and the Ackerman system is described. The method shows the stages that have been followed for the development of the prototype. The development begins with

the definition of the idea, the chassis drawing made in two dimensions (2D) is shown, the three-dimensional modeling (3D) of the parts that make up the Ackerman system, assembly of parts created, study of movement and interference detection of the prototype. The last section presents the results and conclusions and it is proposed as future work to print, assemble and test the Ackerman addressing system model, as well as to continue developing the control and power systems of the disinfection robot.

Keywords: Ackerman Addressing System, Computer Aided Design, Disinfection, COVID-19.

1. Introducción.

Los coronavirus son una extensa familia de virus que pueden causar enfermedades tanto en animales como en humanos. En humanos, se sabe que varios coronavirus causan infecciones respiratorias que pueden ir desde el resfriado común hasta enfermedades más graves como el Síndrome Respiratorio de Oriente Medio (MERS) y el Síndrome Respiratorio Agudo Severo (SRAS). El coronavirus que se ha descubierto más recientemente causa la enfermedad COVID-19. Tanto este nuevo virus como la enfermedad que provoca eran desconocidos antes de que estallara el brote en Wuhan (China) en diciembre de 2019 (Organización Mundial de la Salud, 2019).

Actualmente se utilizan robots móviles en la desinfección de habitaciones y/o superficies contaminadas a causa del virus SARS-CoV-2 como se muestra en los párrafos siguientes.

El Robot UVD fabricado por Blue Ocean Robotics es una herramienta de limpieza para hospitales, ayuda en la desinfección de habitaciones y quirófanos, ha sido utilizado durante la pandemia de COVID-19 en el presente año, la empresa asegura que estos robots pueden eliminar al 99.99% de las bacterias y virus que existieran en las habitaciones gracias a la luz ultravioleta que poseen, en tan sólo 10 minutos. Además, es fácil de usar y está diseñado para ser operado por el personal de limpieza todos los días. El robot UVD ha sido clínicamente probado y verificado en el Hospital Universitario de Odense y en laboratorios microbiológicos independientes (Blue Ocean Robotics, 2020).

Algunas compañías tecnológicas están innovando y comercializando robots para la desinfección, el robot Guoxing Leishen 880T es un ejemplo de esto, el cual se conforma de un chasis que le permite optimizar su desplazamiento mediante dos orugas, con una capacidad de carga de entre 100 y 260 kg., cuenta con un tanque para líquido desinfectante con capacidad de 15 L. y un cañón aspersor de 380°, lo que le permite desinfectar un área de 10,000 m², puede ser utilizado en entornos complejos (Guoxing, 2020).

Científicos de Singapur han inventado un robot desinfectante que imita los gestos humanos para ayudar al personal de limpieza que ha sido desbordado desde el inicio de la crisis del coronavirus. El XDBOT (Robot de extrema desinfección) tiene forma de caja rectangular montada sobre ruedas y armada con un brazo articulado. Llega a lugares difíciles de limpiar, como debajo de las camas o mesas. Creado por investigadores de la Universidad Tecnológica de Nanyang, también contiene un tubo para desinfectar grandes superficies. El robot está dirigido desde un ordenador o una tableta, lo que reduce el riesgo para el personal de resultar infectado por el virus (El Comercio, 2020).

Actualmente se utiliza el direccionamiento Ackerman para el control de robots móviles con un alto grado de estabilidad y maniobrabilidad como se reporta a continuación.

El sistema Ackerman más utilizado es el de ruedas delanteras directrices y traseras motrices, actualmente es el principio básico de los vehículos automotrices. La estabilidad del sistema se debe a la geometría de direccionamiento, que posibilita giros independientes y pivoteos independientes a cada rueda. El principio Ackerman permite cerrar el ángulo de la rueda interior con respecto al exterior (King-Hele & S., 2002).

Se presenta el desarrollo de un controlador para conducir de forma autónoma al robot AutoNOMOS tipo Ackerman, manteniéndolo en su carril. El problema se enfoca en calcular el ángulo de dirección del vehículo, para dar

seguimiento a una trayectoria de referencia; manteniendo el robot en el centro del camino y con una orientación apropiada. El ángulo de dirección es controlado por un servomotor de 5 V. A bordo también se encuentra una computadora Odroid XU4 con el sistema operativo Ubuntu 16.04, cámara digital y sensores (O. & L., 2006).

Para mejorar el control de movimiento y seguimiento de un robot móvil omnidireccional de accionamiento completo, se estudió la dirección Ackerman en proceso dinámico. Primero, se establece el modelo cinemático basado en la estructura física del robot. Luego, la relación entre los parámetros del robot durante el movimiento se deriva de acuerdo con las restricciones dadas por el principio de Ackerman, y se dan la relación de velocidad del motor, la ecuación de estima y la ecuación de movimiento para describir el estado del robot. Finalmente, los resultados de la simulación y el experimento de campo verifican la exactitud del modelo (Oscar, M., & M., 2019).

La aplicación del software de Diseño Asistido por Computadora (CAD) en la ingeniería abarca la elaboración de cuadros sinópticos, diagramas de diversos tipos, gráficos estadísticos, representación normalizada de piezas para su diseño y fabricación, representación tridimensional de modelos dinámicos en multimedia, análisis con elementos finitos, aplicaciones en realidad virtual, robótica, etc. El proceso de diseño en software CAD consiste en la identificación del problema, ideas preliminares, refinamiento del diseño, análisis y optimización, decisión y resultados (documentación) (Jie, Sheng, Kang, Qiang, & Xiao, 2020).

El problema que se aborda es, el diseño de un prototipo de direccionamiento Ackerman que brinde una gran estabilidad al chasis de un robot móvil orientado a la desinfección de superficies contaminadas. Para lo cual se utilizó la herramienta CAD SolidWorks que utiliza un entorno gráfico basado en Microsoft Windows. Se trata de un software de diseño mecánico con poderosas herramientas que cubren el proceso de desarrollo (creación de modelos sólidos 3D, ensamblajes y dibujos, validación antes de fabricar, comunicación y gestión) de productos (Autodesk, 2020; Dessault Systèmes SolidWorks Corporation, 2020).

2. Métodos.

El presente trabajo se desarrolló en el Laboratorio de Sistemas Robóticos (SIRO) de la Facultad de Ciencias de la Computación, el instrumento esencial para trabajar con el CAD fue: una computadora con Windows 10, procesador Core i5 de 7.^a generación, tarjeta gráfica integrada, RAM (por sus siglas en inglés Random Access Memory) de 8 gigabytes (GB) y disco duro de 1 Terabyte (TB).

La metodología del proceso para el diseño del sistema Ackerman utilizando la herramienta CAD se muestra en la figura 1, en los párrafos siguientes se describe detalladamente cada una de las etapas que lo conforman. El plano/dibujo del modelo puede ser creado después del modelado 3D o después del ensamble dependiendo de los resultados a documentar.

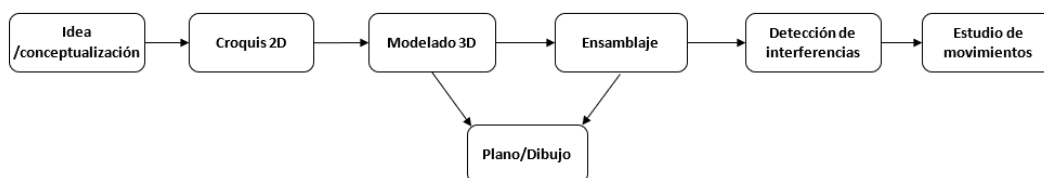


Figura 1. Diagrama del proceso utilizado para el diseño del sistema Ackerman.

3. Desarrollo.

Idea o conceptualización. De acuerdo con la información consultada se definieron los elementos que conforman el sistema de direccionamiento Ackerman: dos manguetas, una varilla de dirección, dos bieletas, dos ruedas, dos brazos, un servomotor, baleros, tuercas y tornillos diseñados parte por parte.

Dibujo o croquis en 2D. Para el desarrollo del croquis en 2D se dibujan los bordes de la figura desde una vista de planta, se crean líneas de apoyo y se agregan las cotas que pueden dimensionarse en milímetros o centímetros de acuerdo con los requerimientos del trabajo, en este caso se optó por trabajar en milímetros, la vista obtenida de croquis 2D se muestra en la figura 2.

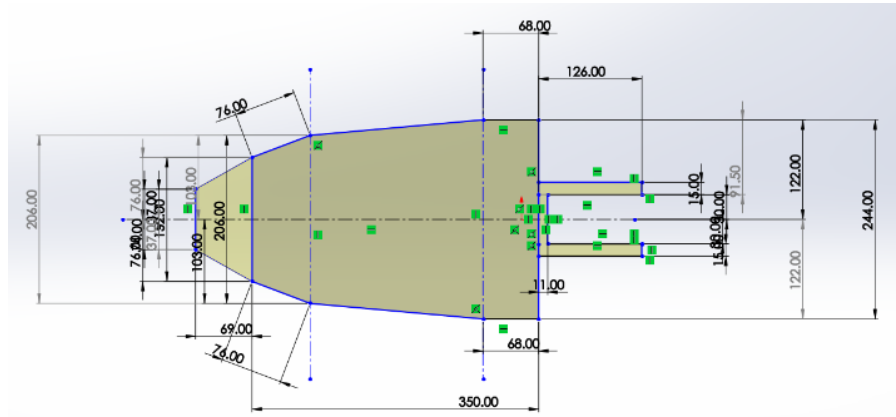


Figura 2. Vista de croquis 2D.

Modelado 3D. En la etapa de modelado en 3D se realizó la extrusión del croquis 2D. Es donde se dan las dimensiones de espesor a cada pieza (en este caso milímetros), en esta etapa es posible realizar cortes, redondeos, vaciado de piezas, entre otras operaciones, los resultados de esto se muestran en la figura 3.

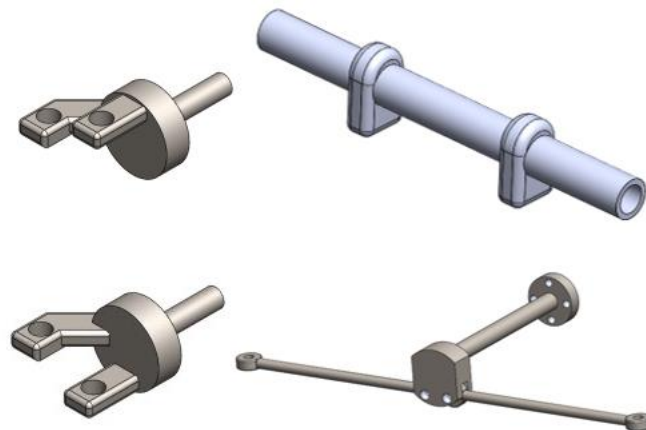


Figura 3. Resultado de modelado en 3D de manguetas (izquierda y derecha), barra estabilizadora, barra de dirección y bieletas.

Ensamble. En esta etapa se obtuvo el “ensamble” realizado al unir todas las piezas que conforman el sistema mecánico y que garantizan su funcionamiento óptimo. Las piezas se agregaron en un archivo de ensamble y se unieron mediante relaciones de posiciones, coincidentes, paralelas, concéntricas y distancias establecidas.

Plano/Dibujo. En esta etapa se obtiene el plano/dibujo del diseño (ensamble) final como parte de la documentación del proyecto, donde se pueden observar las medidas para una mejor visualización y entendimiento de este.

Detección de interferencia. Se realizó el estudio para la detección de interferencias entre los componentes del sistema de dirección. El software cuenta con un complemento que facilita identificar interferencias entre los componentes, proporciona asistencia para examinarlas y evaluarlas. Esta detección de interferencias es útil en ensamblajes complejos, donde es difícil determinar visualmente si los componentes interfieren entre sí.

Estudio de movimiento. Permite simular y analizar con precisión el movimiento de un ensamble a la vez que le incorpora efectos de los elementos de estudio de movimiento para verificar que no existan colisiones entre los componentes.

4. Resultados.

Se obtuvo el ensamble final del sistema de direccionamiento Ackerman, que resulta de la unión de los componentes conceptualizados en la etapa de idea, y creados en la etapa de modelado 3D, el diseño terminado de este sistema se muestra en la figura 4.

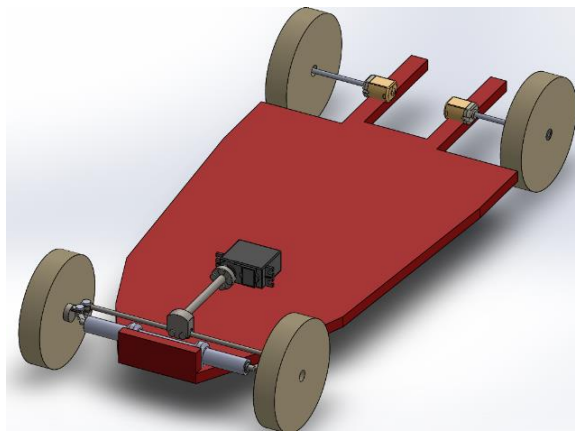


Figura 4. Ensamblaje del sistema de direccionamiento del Robot tipo Ackerman.

Como parte de la documentación se tiene el plano/dibujo del ensamble, el cual nos permite visualizar las dimensiones de este sistema, como se muestra en la figura 5.

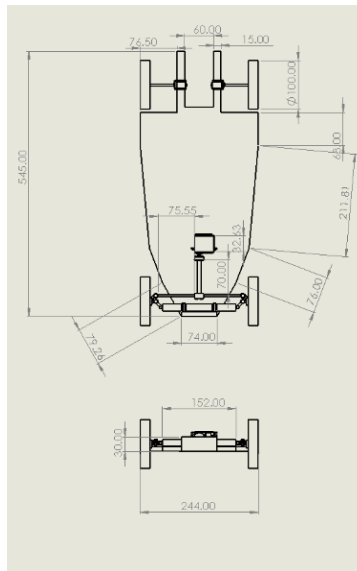


Figura 5. Plano/dibujo del ensamble del sistema de direccionamiento Ackerman.

En la figura 6 se muestran los resultados que arroja el software de la detección de interferencias, muestra automáticamente las colisiones entre los componentes, en la barra lateral del asistente muestra que el sistema no detectó interferencias de ningún tipo entre sus componentes.

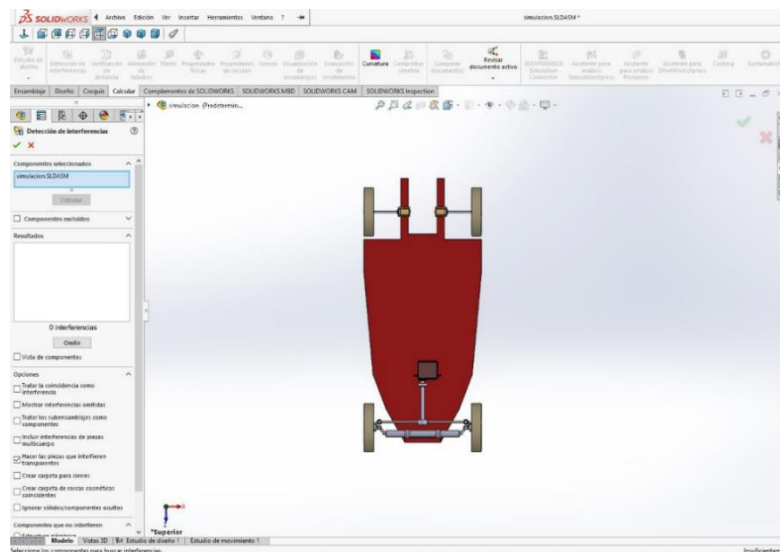


Figura 6. Resultados del estudio de interferencias.

Como resultado del estudio de movimiento, el software simula durante 5 segundos el giro del servomotor que da el movimiento a las ruedas del sistema de direccionamiento Ackerman y detecta la existencia de colisiones entre chasis y ruedas. Como evidencia se presenta la figura 7.

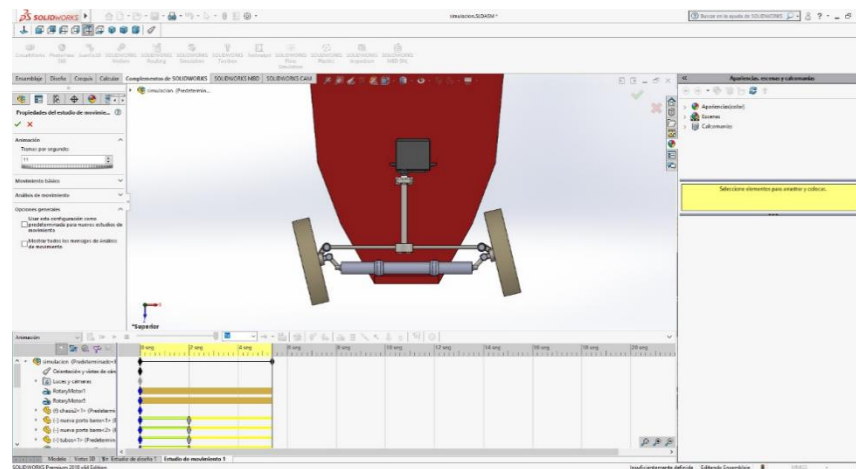


Figura 7. Pantalla de resultado de estudio de movimiento.

Mediante las ecuaciones se muestran a continuación se obtuvo el porcentaje de Ackerman, el cual ayuda a mejorar la estabilidad del robot evitando posibles volcaduras debido al peso y velocidad de desplazamiento.

Los datos propuestos para este prototipo son:

Ángulo de giro de rueda virtual (r_v) = 20° .

Distancia entre base (d_b , distancia entre eje delantero y trasero) = 469.18012678 mm.

Longitud del eje (l_e) = 223.3894144mm.

El cálculo del ángulo de la rueda derecha (r_d) se obtiene con la ecuación 1

$$r_d = \arctan\left(\frac{d_b + \tan r_v}{d_b + 0.5 * l_e * \tan r_v}\right) \quad (1)$$

Sustituyendo en la ecuación 1 los datos propuestos se obtiene el ángulo de la rueda derecha como se muestra a continuación.

$$\begin{aligned} r_d &= \arctan\left(\frac{469.18012678 + \tan 20^\circ}{469.18012678 + 0.5 * 223.3894144 * \tan 20^\circ}\right) \\ r_d &= \arctan\left(\frac{469.18012678 + 0.3639702343}{469.18012678 + 0.5 * 223.3894144 * 0.3639702343}\right) \\ r_d &= \arctan\left(\frac{170.7676007}{509.8336755}\right) \\ r_d &= \arctan(0.334947668) \\ r_d &= 18.52^\circ \end{aligned}$$

El cálculo del ángulo de la rueda izquierda (r_i) se obtiene con la ecuación 2.

$$r_i = \arctan\left(\frac{d_b + \tan r_v}{d_b - 0.5 * l_e * \tan r_v}\right) \quad (2)$$

Sustituyendo en la ecuación 2 los datos del diseño previamente creado se obtiene el ángulo de la rueda izquierda como se muestra a continuación.

$$r_i = \arctan\left(\frac{469.18012678 + \tan 20^\circ}{469.18012678 - 0.5 * 223.3894144 * \tan 20^\circ}\right)$$

$$r_i = \arctan\left(\frac{469.18012678 + 0.3639702343}{469.18012678 - 0.5 * 223.3894144 * 0.3639702343}\right)$$

$$r_i = \arctan\left(\frac{170.7676007}{428.526578}\right)$$

$$r_i = \arctan(0.3984994384)$$

$$r_i = 21.73^\circ$$

El porcentaje de Ackerman se obtiene con la ecuación 3.

$$\text{Porcentaje Ackerman} = 100 * \frac{r_d}{r_i} \quad (3)$$

Sustituyendo en la ecuación 3 los datos previamente obtenidos podemos calcular el porcentaje de Ackerman como se muestra a continuación.

$$\text{Porcentaje Ackerman} = 100 * \frac{18.52^\circ}{26.19^\circ}$$

$$\text{Porcentaje Ackerman} = 100 * \frac{18.52^\circ}{21.73^\circ}$$

$$\text{Porcentaje Ackerman} = 100 * 0.852277957$$

$$\text{Porcentaje Ackerman} = 85.23$$

Conclusiones.

La geometría Ackerman mostró su eficiencia durante la simulación ya que se lograron tener los mismos ángulos de giro de la simulación y los obtenidos en las ecuaciones (rueda exterior con 18.52° e interior con 21.73°) con lo que se corrobora la geometría Ackerman del 85.23%. Estos parámetros ayudan a mantener la estabilidad del sistema, el cual se ha diseñado para transportar 15 Kg., apoyados en la simulación del software se ha corroborado que no se presentan colisiones entre chasis y llantas.

El uso de herramientas tecnológicas de tipo CAD permiten diseñar y simular sistemas mecánicos con un alto porcentaje de efectividad, por lo que en la actualidad se utilizan ampliamente en los sectores industriales, ya que permiten obtener una visión del modelo terminado antes de implementarlo físicamente, lo cual permite ahorrar tiempo en errores que pueden ser detectados gracias a estas herramientas.

En este trabajo en particular, se realizaron algunas correcciones, de acuerdo con los resultados obtenidos durante la simulación hasta llegar al modelo ideal; como las dimensiones de manguetas y bieletas que permitieran girar libremente a las ruedas sin chocar con el chasis. Por lo que se concluye seguir utilizando este tipo de herramientas para continuar este y otros desarrollos relacionados con el área de la robótica y las ciencias de la computación.

Los retos de la investigación son continuar con las etapas del desarrollo de un brazo robótico y sistema de control para que el robot pueda funcionar como una herramienta tecnológica para la desinfección de superficies o habitaciones que hayan estado en contacto con personas con COVID-19.

Agradecimientos.

Agradecemos a los integrantes del laboratorio SIRO, que han brindado sus conocimientos, habilidades y experiencias directa e indirectamente durante el desarrollo de esta investigación. A la facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla por brindarnos un espacio físico e instalaciones adecuadas para realizar esta investigación.

Referencias Bibliográficas.

- Autodesk.** (2020). *SOFTWARE DE DISEÑO CAD*. <https://www.autodesk.mx/solutions/cad-software>.
- Blue Ocean Robotics.** (2020). *UVD Robots revolutionizing disinfection*. <https://www.uvd-robots.com/>.
- Dassault Systèmes SolidWorks Corporation.** (2020). *SolidWorks*. <https://www.solidworks.com/es>.
- El Comercio.** (2020). *Prueban en Singapur un robot desinfectante contra el coronavirus*. <https://www.elcomercio.com/tendencias/prueban-singapur-robot-desinfectante-coronavirus.html>
- Guoxing.** (2020). *Guoxing Leishen 880T desinfección plataforma robot*. <https://spanish.alibaba.com/product-detail/guoxing-leishen-880t-disinfection-robot-platform-62497641931.html?spm=a2700.8699010.normalList.11.2b603a9fwd582X>
- Jie, Z., Sheng, L., Kang, H., Qiang, S., & Xiao, K.** (2020). *Steering Analysis of Omnidirectional Full Drive Mobile Robot in Dynamic Process*. International Conference on Intelligent Human-Machine Systems and Cybernetics, 162-165. ISBN:978-1-7281-6518-9.
- King-Hele, D., & S., F. R.** (2002). *Erasmus Darwin's improved design for steering*. The Royal Society, 41-62, DOI:10.1098/rsnr.2002.0166.
- O., O. L., & L., R. R.** (2006). *Diseño asistido por computadora*. Diseño Y Tecnología. Industrial Data, 7-15, ISSN: 1560-9146.
- Organización Mundial de la Salud.** (2019). Preguntas y respuestas sobre la enfermedad por coronavirus (COVID-19). Obtenido de Organización Mundial de la Salud: <https://www.who.int/es/emergencias/diseases/novel-coronavirus-2019/advice-for-public/q-a-coronaviruses>
- Oscar, G.-M., M., O.-S. S., & M., I.-Z. J.** (2019). Control basado en campos potenciales para un vehículo autónomo usando realimentación visual. Research in Computing Science, 253–262, ISSN 1870-4069.

Información de los autores.



José Luis Hernández Ameca. Obtuvo el título de Ingeniero Electrónico del Instituto Tecnológico de Puebla en 2001. En 2005 recibió el título de Maestro en Ciencias de la Computación por la Benemérita Universidad Autónoma de Puebla (BUAP). En 2017 obtuvo el título de Doctor en Planificación Estratégica y Gestión de la Tecnología de la Universidad Popular Autónoma de Puebla (UPAEP). De 2002 a 2004 trabajó como Ingeniero en el departamento de implementación de proyectos de la empresa digital IUSACEL. Trabaja como Profesor en la facultad de Ciencias de Computación de la BUAP desde 2007, donde su línea de investigación es la robótica y educación.



María del Carmen Báez Salazar. Estudiante de la carrera de Ingeniería en Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla. Actualmente se encuentra realizando su tesis en el laboratorio SIRO. Ha participado con exposición de carteles y conferencias en los congresos: IX Congreso Nacional De Tecnología Aplicada A La Salud (CONTACTS), Congreso Nacional De Ciencias De La Computación (CONACIC 2018 y 2019), Congreso Nacional De Tecnología En La Educación (CONTE 2018, 2019 y 2020). Durante su participación en el Laboratorio SIRO ha participado los concursos “COCOTRON” obteniendo el primer lugar, participo como miembro del comité organizador de VIII Torneo Nacional De Robótica “GUERRA DE DIOSES 2018 y 2019” Por último ha sido miembro del comité organizador del evento “NOCHE DE LAS ESTRELLAS SEDE BUAP 2018 y 2019”.



Luis Enrique Colmenares Guillén, realizó sus estudios de la Licenciatura en Computación en la Benemérita Universidad Autónoma de Puebla (BUAP). La Maestría en la Universidad de las Américas Puebla. El doctorado fue realizado en la Universidad Politécnica de Cataluña de Barcelona España. Actualmente es profesor investigador de la Facultad de Ciencias de la Computación de la BUAP. Pertenecer a un cuerpo académico PRODEP consolidado. Es miembro del Registro CONACYT de Evaluadores Acreditados (RCEA). Pertenecer al Sistema Nacional de Investigadores (SNI) y a dos redes temáticas CONACYT. Pertenecer al padrón de Investigadores y de Consultores de la BUAP. Actualmente tiene publicaciones nacionales e Internacionales, además tiene cuatro patentes registradas y dos títulos de patente. En la Facultad de Ciencias de la Computación ha impartido las cátedras de Sistemas Operativos, Administración de proyectos, Sistemas Distribuidos, Procesamiento Digital de imágenes, Sistemas de tiempo real. Actualmente ha desarrollado algoritmos y sistemas clasificadores para el área de la Inteligencia artificial y reconocimiento de patrones.



Elsa Chavira Martínez, Licenciatura en Física, BUAP, la Maestría en Ciencias en Física del Estado Sólido en el Instituto de Física, BUAP y obtuvo en grado de Doctora en Ingeniería Biomédica UAPEP, Posgrado de Excelencia PNP, CONACYT. Como científica e investigadora ha trabajado en: Física de Superficies, Materiales Semiconductores, es especialista en el crecimiento de Silicio monocristalino por la técnica Czochralski, en el campo de la Microelectrónica, ha diseñado diversos Circuitos Integrados de Aplicación Específica. Actualmente está adscrita a la Facultad de Ciencias de la Computación (FCC) en la BUAP, como Profesora - Investigadora Titular A Tiempo Completo.

Diseño y construcción de un prototipo dispensador automático de alimentos para ganado porcino.

Design and construction of a prototype automatic feed dispenser for pigs.

Hernán Valencia Sánchez* (1).
TecNM/ITTG.
hernan.vs@tuxtla.tecnm.mx.

Roberto Carlos García Gómez. TecNM/ITTG, roberto.gg@tuxtla.tecnm.mx.

Fernando Alfonso May Arriola. TecNM/ITTG, fernando.ma@tuxtla.tecnm.mx.

Pedro Isaac López Hernández. Estudiante TecNM/ITTG, isaac07pedr0@gmail.com.

Amín Vázquez Velázquez. Estudiante TecNM/ITTG, amin981820@gmail.com.

*corresponding author.

Artículo recibido en julio 07, 2021; aceptado en agosto 16, 2021.

Resumen.

El proyecto consiste en el diseño y construcción de un prototipo de máquina que ayude a controlar la cantidad de alimento para ganado porcino, disminuyendo el tiempo y la mano de obra del proceso de alimentación de ganado. Este proyecto es enfocado en la industria de ganado porcino. La carencia de la tecnología, crea la necesidad de innovación e implementación de máquinas, buscando alternativas que ayuden a mejorar el proceso de alimento al ganado, dando cantidades adecuadas en el tiempo exacto para cada etapa del ganado.

Palabras claves: Diseño virtual, arduino, porcino, solidworks.

Abstract.

This project involves the design and construction of a prototype machine to help control the amount of feed for pigs, reducing the time and labor of the livestock feeding process. This Project is focused on the pig industry. The lack of the technology creates the need for innovation and implementation of machines, looking for alternatives that help improve the process of feeding livestock giving adequate amounts in the exact time for each stage of livestock.

Keywords: Virtual design, arduino, porcine, solidworks

1. Introducción.

La industria porcina se considera una de las más aceptadas a nivel mundial gracias al significativo consumo de carne de este tipo de animales, tan solo en 2019, el volumen de producción de carne de porcino en México superó los 1,6

millones de toneladas métricas. Esto representó un incremento de alrededor del 6,6% en comparación con el volumen de producción reportado un año anterior (Padilla M.,2007).

La alimentación representa un papel muy importante, una nutrición adecuada es fundamental para una exitosa producción porcina. “Bajos pesos a la venta que van desde 80 a 90 kilos en cerdos que han llegado a 22 semanas de edad, por lo tanto, no alcanzan el peso óptimo en los cerdos de engorde pueden representar importantes pérdidas económicas” (Cavenco, 2017). Alto índice de conversión alimenticia que es sinónimo de rentabilidad, puesto que es una relación directa entre el alimento que consume un cerdo y la ganancia de peso que tiene. Es decir, un cerdo eficiente debe tener un consumo bajo de alimento y una alta ganancia de peso.

En la ración diaria, es necesario proveer una cantidad adecuada de alimento para obtener una buena ganancia diaria de peso y un buen crecimiento en la producción porcina.

Para la solución de esta problemática se propone en “Diseñar y construir un Prototipo que permita controlar la cantidad necesaria de alimento de manera automatizada para el ganado porcino.”

Las características del dispositivo son las siguientes:

- Motor.
- Tornillo sin fin.
- Tubo de desplazamiento principal.
- Celdas de alimentos.
- Tarjeta de adquisición de datos.
- Servomotores.
- Sensor.

Diseño de las piezas del prototipo usando el software de SolidWorks

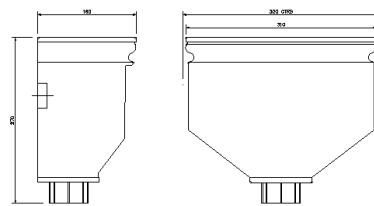


Figura 1. Diseño de tolva.

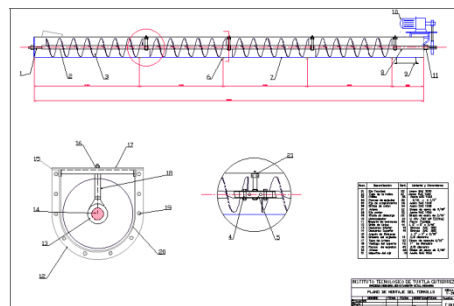


Figura 2. Diseño tornillo sin fin.

2. Métodos.

Cálculos para el transporte del alimento.

Cantidad de alimento que se transportará.

El alimento para cerdos peletizado es un material ligero y poco abrasivo, que tiene como coeficiente de relleno (λ)=0.32.

El área de relleno (S) del canalón que ocupa el material que mueve el transportador, se puede obtener mediante la siguiente expresión.

$$S = \lambda \frac{\pi \times d^2}{4} \dots\dots\dots \text{Ecuación 1.}$$

Por lo tanto.

$$S = 0.32 \frac{\pi \times 0.1142^2}{4} = 0.00327 \text{ m}$$

Velocidad de desplazamiento (v) del transportador.

Es la velocidad con la que desplaza el material en la dirección longitudinal del eje del tornillo. Depende tanto del paso del tornillo como de su velocidad de giro (Budynas, R., 2012).

$$v = \frac{p \times n}{60} \dots\dots\dots \text{Ecuación 2.}$$

n = velocidad de giro.

p = paso del tornillo sin fin.

$$v = \frac{0.0571 \times 120}{60} = 0.1142 \text{ m/s}$$

Capacidad de transporte.

La capacidad de transporte de un transportador de tornillo sin fin viene determinada por la siguiente expresión que calcula el flujo de material transportado.

$$Q = 3600 \times S \times v \times \rho \times i \dots\dots\dots \text{Ecuación 3.}$$

ρ = densidad del material.

i = coeficiente de disminución del flujo de material.

γ = peso específico.

Teniendo.

$$\gamma = \rho \times g \dots\dots\dots \text{Ecuación 4}$$

Por lo tanto.

$$\rho = \frac{\gamma}{g} \dots\dots\dots \text{Ecuación 5}$$

$$\rho = \frac{0.8}{9.81} = 0.0815 \text{ kg/m}^3$$

El transportador no tendrá inclinación, por lo tanto, $i = 1$.

$$Q = 3600 \times 0.003227 \times 0.1142 \times 0.0815 \times 1 = 0.10812 \text{ m}^3/\text{s}$$

Diseño del Dispensador Automático de Alimentos para el Ganado Porcino.

Se seleccionó este diseño por ser, bajo el análisis de transporte de cantidad de alimento, el que cumple con las características buscadas. Éste consiste en una tolva que almacenará alimento seco, montada en una sección que permitirá que un tornillo sin fin, ensamblado a un motor eléctrico, mueva el alimento a través de un tubo diseccionado por tubos de montaje “T”, los tubos “T” tendrán otra ampliación de tubo que estarán conectados a electroválvulas para detener o dejar pasar el alimento.

Diseño del mecanismo de ensamble.

Los diseños fueron creados con el software SolidWorks. Para el diseño del mecanismo se proponen 1 tolva, 2 tubos de entrada, 1 motor, y 1 tornillos sin fin (Gómez S., 2012).

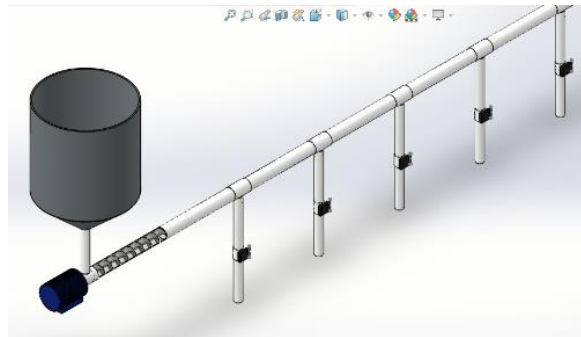


Figura 3. Diseño de mecanismo.

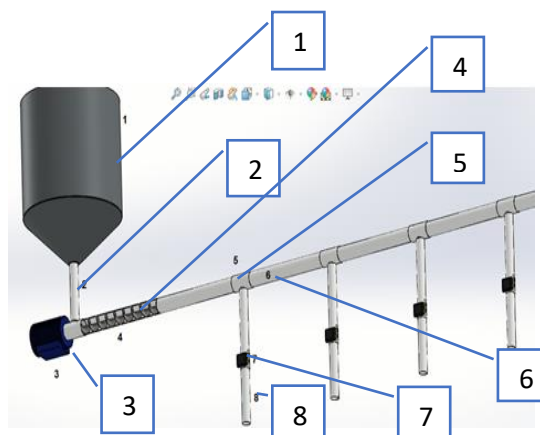


Figura 4. Componentes principales del alimentador.

Este diseño está constituido por los accesorios que se indican en la siguiente tabla.

Tabla 1. Accesorios de máquina.

Número	Material	Cantidad
1	Tolva	1
2	Tubo de entrada	1
3	Motor	1
4	Tornillo sin fin	1
5	Conexión tee	5
6	Tubería principal de distribución	1
7	Unión de la válvula con el tubo	4
8	Tubería de salida	4

Explicación del funcionamiento del dispensador.

El alimento proveniente de la tolva (elemento 1, ver la figura 4) será dirigido hacia el sistema a través de una sección tubular (2), el cual, al caer en la tubería principal (6) será impulsado por el tornillo sin fin (4), éste elemento realizará un movimiento rotatorio generado por el motor eléctrico (3) al que se encontrará conectado. Gracias al continuo movimiento de giro que efectuará el tornillo sin fin el alimento seco podrá desplazarse a lo largo de la longitud del tubo principal (6), cuando llegue a las secciones donde se encuentran las uniones con los tubos “T” (5) el alimento caerá a lo largo de la sección circular que lleva el número 6 en la figura 4, debido a que las electroválvulas (7) estarán cerradas el alimento ocupará el volumen vacío dentro de la tubería (8) colocada en posición vertical, posteriormente las electroválvulas recibirán el impulso eléctrico para que se abran y permitan el descenso del alimento para cerdos, el impulso eléctrico será enviado desde el controlador programable, es decir, la placa de adquisición de datos, según la configuración que el usuario haya ingresado de acuerdo a los horarios de alimentación para los cerdos (Rodríguez, H. s.f.).

Sistema de control del prototipo.

Un sistema de control es un tipo de sistema que se caracteriza por la presencia de una serie de elementos que permiten influir en el funcionamiento del sistema.

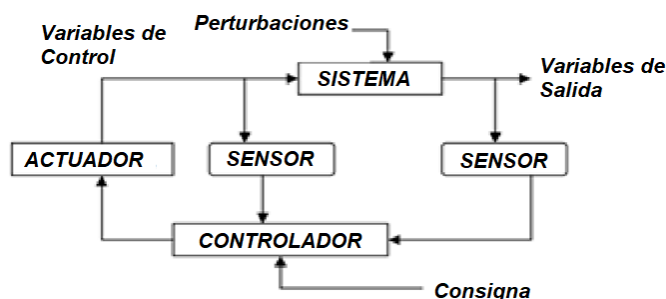


Figura 5. Elementos de un sistema de control.

La finalidad de un sistema de control es conseguir, mediante la manipulación de las variables de control, un dominio sobre las variables de salida, de modo que estas alcancen unos valores prefijados (consigna). Los elementos básicos que forman parte de un sistema de control más utilizados en el sector industrial.

Sensores. Permiten conocer los valores de las variables medidas del sistema.

Controlador. Utilizando los valores determinados por los sensores, son almacenados en una tarjeta de adquisición de datos que nos permite calcular la acción de acuerdo a la consigna impuesta, calcula la acción que debe aplicarse para modificar las variables de control en base a ciertos parámetros requeridos en el proceso.

Actuador. Es el mecanismo que ejecuta la acción calculada por el controlador y que modifica las variables de control. Toda producción liderada por ingeniería requiere de este proceso para lograr objetivos determinados. La función de este sistema es la de gestionar o regular la forma en que se comporta el sistema para así evitar fallas (Uraný, P., 2015).

Sistema de lazo Abierto: es aquel sistema en el cual la acción de control es, en cierto modo, independiente de la salida. Este tipo de sistemas por lo general utiliza un regulador o actuador con la finalidad de obtener la respuesta deseada.

La capacidad que tales sistemas tienen para ejecutar una acción con exactitud depende de su calibración. En general, los sistemas de lazo abierto están regulados por base de tiempo.

Controlador: es aquel instrumento que compara el valor medido con el valor deseado, en base a esta comparación calcula un error y actuar a fin de corregir este error. Entonces la función del controlador es mantener la variable controlada dentro de los criterios previamente establecidos.

```
motor_on_off
//Declaro las variables llamadas Boton y Motor
int boton=2; //El botón se conecta en el pin 2 del Arduino.
int motor=6; //El Motor se conecta en el pin 13 del Arduino.
int estado; //Esta variable la usaremos para saber el estado del botón, HIGH si alguien pulsa el botón, LOW si el botón no es pulsado.

void setup()
{
  pinMode(motor,OUTPUT); //Se configura el pin digital 13 como salida, esto es así porque la variable LED vale 13.
  pinMode(boton,INPUT); //Se declara el pin 2 como entrada, esto es así porque la variable botón vale 2
}

void loop()
{
  estado = digitalRead(boton); //Estado toma el valor que tenga el botón, = es para asignar el valor.
  if(estado==HIGH)           //Esta línea compara (pregunta) si el estado es igual a HIGH, así sabemos si el botón está pulsado o no.
  {
    digitalWrite(motor,HIGH); //Escribimos en el pin digital 13 un estado en alto (motor encendido).
  }
  else //Cuando la condición del if no se cumple entramos al else.
  {
    digitalWrite(motor,LOW); //Ahora establecemos el pin digital 13 en bajo (motor apagado).
  }
}
```

Figura 6. Motor principal código de programación "arduino".

En la figura 6, presentamos la programación que se utilizó para nuestro motor principal, el cual tenía como objetivo el movimiento del tornillo sin fin.

Como podemos ver la programación se definió para añadir un botón de apagado y encendido, dentro del mismo programa se define de forma clara el proceso de funcionamiento.

```

alimentador_de_ganado_sensor_de_presion_y_servo_motor-modificado
//incluimos la libreria del servo y declaramos los pines a utilizar
#include <Servo.h>
int servo=4;
Servo myservo;
int entrada=0;
void setup() {
  // se declara al sensor de presion como variable de entrada, ligada al servo
  myservo.attach(2);
  pinMode(sensor,INPUT);
}
void loop() {
  // se inicia el ciclo del servo con la condicion del sesor de presion
  entrada=digitalRead(sensor);
  //si el sensor detecta presion se enciende el motor con un giro de 180 grados y espera 20 segundos, si no hay presion suficiente el motor no enciende y se cierra el ciclo
  if (entrada==HIGH)
  {
    myservo.write(180);
    delay (20000)
  }
  else
  {
    myservo.write (0);
  }
}

```

Figura 7. Servo Compuerta Código de programación "arduino".

En la figura 7, encontramos la programación para el servo motor que se encuentra en las tuberías verticales (ver figura 4 (8)), el objetivo de esta programación es determinar el tiempo óptimo para que las tuberías sean vaciadas y servir el alimento al ganado.

El tiempo que nosotros definimos como 20 segundos fue basado al tiempo de llenado de una sola tubería, pero el programa puede ser modificado de acuerdo a las necesidades que se tengan

3. Resultados.

Prototipo del dispensador automático de alimentos para el ganado porcino.

Con el fin de corroborar las condiciones de diseño dispuestas para el equipo, se construyó un prototipo que permitió caracterizar el comportamiento del ensamble simulando el paso del alimento de ganado por medio del tornillo y de la tubería propuesta.



Figura 8. Prototipo del proyecto.

Para este prototipo se utilizó un alambre galvanizado para transmitir el movimiento mecánico del motor, se construyó sobre una caja de 600 cm^3 , dentro de la cual está conectada una tubería de pvc con un diámetro de 1/8 in, sobre el cual sería conectado nuestro accesorio de tubería T, para que posteriormente pase a nuestra celda de alimento. Se usó un motor de engranaje para la tarjeta de adquisición de datos al cual fue ensamblado nuestro tornillo sin fin.

El mecanismo del prototipo realizó el desplazamiento del alimento que entró por la tubería principal, pasó al tornillo y este pudo desplazar el alimento por el resto de la tubería.

Construcción del dispensador.

Una vez realizado el desplazamiento del alimento, se construye el proyecto final correspondiente al diseño hecho en SolidWorks, en el cual se utilizó: una tarjeta de adquisición de datos para programación del movimiento y un servomotor para el movimiento de la celda de alimento (Gómez G., 2012)

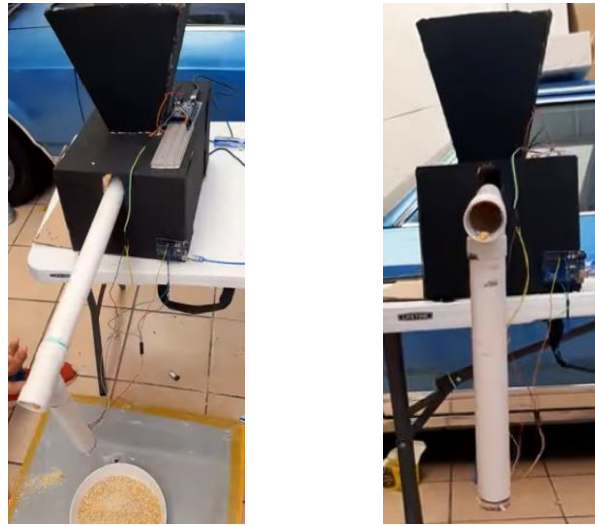


Figura 9. Construcción del alimentador de ganado.

De acuerdo con la investigación que se realizó, los resultados que obtuvimos de la ejecución de esta etapa son las siguientes.

- El prototipo realiza el desplazamiento de alimento de acuerdo al análisis de diseño.
- El prototipo se le programa el horario de alimentación del porcino así como la cantidad de tiempo para que las válvulas accionen las compuertas y proporcionen los alimentos.
- En su mayoría los sistemas de alimento para ganado porcino, están ampliamente aplicados en la industria a gran escala más que para un productor a mediana escala.

Conclusiones.

Al llevar a cabo el “Diseño y construcción de un dispositivo que permite controlar la cantidad necesaria de alimento de manera automatizada para el ganado porcino” podemos concluir que es factible y de mucha utilidad para pequeñas y medianas granjas porcinas, ya que la estructura y funcionamiento del sistema es sencillo, no requiere de un proceso de fabricación complicado, los materiales y accesorios para su elaboración son de fácil acceso.

Para la fabricación de este prototipo se deben realizar prueba en campo y optimizar los materiales. De acuerdo al diseño, es apto para los pequeños productores a los que nos referimos, además de proporcionar aumentos en el crecimiento del ganado y en lo posterior teniendo un impacto económico positivo, porque al implementar un sistema automatizado, promueven una dosificación exacta en tiempo y forma para nuestro ganado, mejorando su calidad.

Comparando los datos obtenidos del análisis por elementos finitos con los valores teóricos, se encuentra que los porcentajes de error son relativamente pequeños, lo que indica la consistencia del método.

Referencias bibliográficas.

Budynas, R. N. (2012). Diseño en Ingeniería Mecánica de Shingley. En R. N. Budynas, *Diseño en Ingeniería Mecánica de Shingley* (págs. 125-235). Mexico : Mc Graw Hill.

Cavenco (2017). Granjas llave en mano. Zaragoza: Cavenco.

Gómez González Sergio (2012). “SolidWorks”, Ed. Alfaomega México, 12da.

Padilla Pérez, M. (2007). CIAP (Centro de información de actividades porcinas). San José, Costa rica: MAG. Recuperado el 11 de Noviembre de 2020, de <http://www.ciap.org.ar/Sitio/Archivos/MANUAL%20DE%20PORCICULTURA.pdf>

Rodríguez, H. (s. f). Ingemecánica. Recuperado el 29 de Septiembre de 2020, de <https://ingemecanica.com/tutorialsemanal/tutorialn143.html>

Urany, P. (2015). Urany P.R. Recuperado el 19 de Septiembre de 2020, de https://urany.net/stock-de%20productos?gclid=CjwKCAiAtej9BRAvEiwA0UAWXl8bIwMMfRg45NIefL3HW3FMF1lhcKHkxmFCeCPvKRve-pX56uobmRoCYAUQAvD_BwE

Información de los autores.



Hernán Valencia Sánchez, Licenciatura en Ingeniería Mecánica y con un Posgrado en Maestría en Ciencias en Ingeniería Mecatrónica Egresado del Instituto Tecnológico de Tuxtla Gutiérrez, Reconocimiento al perfil deseable Jefe de Proyectos de Investigación del Departamento de Metal-Mecánica, Docente del Departamento de Metal-Mecánica del TecNM/Instituto tecnológico de Tuxtla Gutiérrez.



Roberto Carlos García Gómez, Licenciatura en Ingeniería Mecánica, con tres posgrados: Especialidad en Mecatrónica, Maestría en Ciencias en Ingeniería Mecánica, Doctorado en Procesos de Manufactura. Egresado del Instituto Tecnológico de Tuxtla Gutiérrez, Jefe de Proyectos de Vinculación del Departamento de Metal-Mecánica, Docente del Departamento de Metal-Mecánica del TecNM/Instituto tecnológico de Tuxtla Gutiérrez.



Fernando Alfonso May Arriola, Licenciado en Ingeniería Mecánica por el Instituto Tecnológico de Tuxtla Gutiérrez, Maestro en Energías Renovables por la Universidad Politécnica de Chiapas y Doctorante en Ingeniería Aplicada en el Colegio de Formación Educativa Tenam. Profesor de Carrera Enseñanza Superior en el Departamento de Metal-Mecánica del TecNM/Instituto Tecnológico de Tuxtla Gutiérrez. Perfil Deseable desde 2016.



Pedro Isaac López Hernández. Alumno de 8° Semestre de Ingeniería Mecánica del Instituto Tecnológico De Tuxtla Gutiérrez.



Amín Vázquez Velázquez. Alumno de 8° Semestre de Ingeniería Mecánica del Instituto Tecnológico De Tuxtla Gutiérrez.

Sistema de ubicación y reconocimiento de objetos para robot autónomo.

Object recognition and location system for autonomous robot.

Alvaro Hernández Sol* (1).
Tecnológico Nacional de México campus Tuxtla Gutiérrez.
alvaroht@tuxtla.tecnm.mx.

Manuel de Jesús Cancino Escobar (2). Estudiante del Tecnológico Nacional de México campus Tuxtla Gutiérrez,
manuelcancino883@outlook.com.

Raúl Moreno Rincón (3). Tecnológico Nacional de México campus Tuxtla Gutiérrez, rmoreno@tuxtla.tecnm.mx.

José Ángel Zepeda Hernández (4). Tecnológico Nacional de México campus Tuxtla Gutiérrez,
jzepeda@tuxtla.tecnm.mx.

Aldo Esteban Aguilar Castillejos (5). Tecnológico Nacional de México campus Tuxtla Gutiérrez,
aaguilarc@tuxtla.tecnm.mx.

Osbaldo Ysaac García Ramos (6). Tecnológico Nacional de México campus Tuxtla Gutiérrez,
ogarcia@tuxtla.tecnm.mx.

*corresponding author.

Artículo recibido en septiembre 27, 2021; aceptado en octubre 13, 2021.

Resumen.

El presente artículo describe el diseño de un sistema de reconocimiento y ubicación de objetos usando sensores ultrasónicos aplicados a un robot móvil autónomo. Al aplicar este sistema en el prototipo se le proporciona la capacidad de reconocer y ubicar la posición de dicho objeto con la finalidad de poder evadirlo y generar mapas de posición de obstáculos en ambientes semiestructurados. En el reconocimiento de los objetos se utiliza una matriz de nueve sensores ultrasónicos, que incrementa la cantidad de datos a 27 por lectura, controlando su dirección. Estos datos son usados como entradas a una red neuronal artificial "Deep learning", entrenada para realizar el reconocimiento de los diferentes objetos a seleccionar. En pruebas de laboratorio se obtuvieron buenos resultados en el reconocimiento de objetos ubicados hasta una distancia máxima de 50 cm. Al implementar la red en el prototipo robótico este se deberá desarrollar en un ambiente controlado en el cual pueda identificar qué tipo de objeto tiene cerca y así poder ubicarlo dentro de un mapa de entorno.

Palabras clave: Robot, redes neuronales artificiales, reconocimiento de patrones, sensor ultrasónico.

Abstract.

This article describes the design of an object recognition and location system using ultrasonic sensors applied to an autonomous mobile robot. By applying this system in the prototype, it is provided the ability to recognize and locate the position of said object in order to be able to evade it and generate obstacle position maps in semi-structured environments. In the recognition of the objects, a matrix of nine ultrasonic sensors is used, which increases the

amount of data to 27 for reading, controlling its direction. These data are used as inputs to an artificial neural network "Deep learning", trained to recognize the different objects to be selected. In laboratory tests, good results were obtained in the recognition of objects located up to a maximum distance of 50 cm. When implementing the network in the robotic prototype, it must be developed in a controlled environment in which it can identify what type of object is nearby and thus be able to locate it within an environment map.

Keywords: Robot, Artificial Neural Networks, Pattern Recognition, Ultrasonic sensor.

1. Introducción.

La robótica móvil autónoma es un tema de investigación fascinante, por muchas razones. Primero, cambiar un robot móvil de una computadora con ruedas que simplemente es capaz de sentir algunas propiedades físicas del medio ambiente a través de sus sensores en un agente inteligente, capaz de identificar características, detectar patrones y regularidades, aprender de la experiencia, localizar, construir mapas y navegar requiere la aplicación simultánea de muchas disciplinas de investigación (Rios, 2015).

Rodríguez (2015) menciona que “La autonomía de un robot se ve definida por su capacidad de reaccionar ante los acontecimientos que surgen a su alrededor, para ello se debe tener un sistema estable y robusto el cual resuelva de la mejor manera tales circunstancias”. Para el desarrollo de este artículo se basó en un sistema el cual mediante redes neuronales artificiales detecta características específicas de objetos físicos como cuadrados y cilindros, para lograr esto el prototipo robótico cuenta con una serie de sensores ultrasónicos que le permiten “ver” su entorno, estos sensores funcionan como datos de entrada para el prototipo robótico.

Una red neuronal (también llamada red neuronal artificial) siendo un sistema adaptativo semejante al cerebro humano, que aprende mediante el uso de nodos (neuronas) interconectados en una estructura estratificada. Las redes neuronales son especialmente adecuadas para realizar el reconocimiento de patrones para clasificación de objetos o señales en los sistemas de control, visión y habla (García, 2017).

Una de las técnicas de machine learning para llevar a cabo la clasificación de características físicas de objetos presentados a un robot son los sistemas de aprendizaje supervisado. Estos toman un conjunto de datos de una colección de ejemplos etiquetados $\{(x_i, y_i)\}^{N_i} = 1$. Usados para producir un modelo que, a partir de un vector de entrada x_i , consiga deducir una etiqueta y_i correspondiente al vector de características (Gonzalo, 2019).

El aprendizaje profundo (Deep Learning) es una técnica de aprendizaje automático que enseña a las computadoras a hacer lo que es natural para los humanos: aprender con el ejemplo. El aprendizaje profundo es una tecnología clave detrás de los automóviles sin conductor, que les permite reconocer una señal de alto o distinguir a un peatón de una farola. Los modelos de aprendizaje profundo pueden lograr una precisión de vanguardia, a veces superando el rendimiento a nivel humano. Los modelos se entrenan mediante el uso de un gran conjunto de datos etiquetados y arquitecturas de redes neuronales que contienen cientos de capas (Valencia, 2016).

2. Métodos.

Deep Learning.

En el aprendizaje profundo, un modelo de computadora aprende a realizar tareas de clasificación directamente a partir de imágenes, texto o sonido. Los modelos de aprendizaje profundo pueden lograr una precisión de vanguardia, a veces superando el rendimiento a nivel humano. Los modelos se entrenan mediante el uso de un gran conjunto de datos etiquetados y arquitecturas de redes neuronales que contienen muchas capas, para el desarrollo de una red neuronal artificial se utilizó el Toolbox de Matlab “Deep Learning”. (The Mathworks, 2021).

Como menciona Goodfellow, los modelos de Deep learning utilizados están basados en las redes profundas de retro propagación, también llamadas “redes neuronales de retro propagación”. La finalidad de estas redes es aproximar un función f^* por medio de un clasificador $y = f^*(x)$ convirtiendo una entrada x a una categoría y definiendo esta transformación como $y = f(x; \theta)$.

Como podemos representar una red neuronal como una composición de funciones diferentes, dependiendo de la cantidad de funciones dependerá la profundidad de la red neuronal.

$$f(x) = f^{(3)}(f^{(2)}(f^{(1)}(x)))$$

Siendo $f(1)$ la capa de entrada, $f(2)$ la segunda capa y $f(3)$ la capa de salida. Cada una de estas funciones las podemos representar como un vector.

Un algoritmo de hipótesis final como el presentado por Abu-Mostafa (2012), nos menciona que para una función objetivo desconocida (como fórmula aproximada de manera ideal).

$$f : x \rightarrow y$$

Y teniendo un grupo de ejemplos históricos de objetos caracterizados.

$$(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$$

Generamos un algoritmo de aprendizaje A , basado en un grupo de reglas hipotéticas H , con las cuales el algoritmo genera un fórmula de aprendizaje como resultado de la aplicación como entrada f .

$$g \approx f$$

Basado en un proceso de optimización que se utiliza para encontrar los valores óptimos w^* y b^* minimizando la expresión de la función objetivo.

$$\frac{1}{N} \sum_{i=1 \dots N}^N (f_{w,b}(x_i) - y_i)^2$$

Siendo la pérdida promedio, para un modelo, la media de las penalizaciones obtenidas aplicando el modelo al grupo de entrenamiento (Burkov, 2019).

Por esa razón si considerando lo que Sandoval (2018) menciona “si se observa que hay entre un 80% a 90% de respuestas correctas, podemos decir que hay un buen grado de aprendizaje y poder utilizar ese algoritmo”, se toma esta premisa para realizar la selección de la red neuronal a utilizar, la de mejor porcentaje de acierto para la implementación en el prototipo robótico. Con esta herramienta se entrenaron 41 redes neuronales distintas,

Para llevar a cabo este entrenamiento se construyó un robot móvil con ruedas de movimientos independientes. También se montó un sistema de recolección de datos, usando para esto 9 sensores ultrasónicos SRF08.

Para elaborar el chasis se construyó un marco con perfiles de aluminio en forma de C de $1 \times 25 \times 1$ agujeros los cuales tienen una medida de 32 cm de largo y 3 cm de ancho. Quedando un marco con dimensiones de 32 cm de largo y 32 cm de ancho. Se ubicó un motor NEMA17 en la parte central del móvil para colocar una caja de engranajes reductora de tipo planetario, la cual se imprimió, usando PLA, ver figura 1.



Figura 1. Montaje de la caja reductora de engranes helicoidales

Para la sistema de “visión” se construyó una base giratoria de 31cm de diámetro y un grosor de 6 mm del material ABS plástico, siendo esta sujeta por medio de un piñón dentro de la caja de engranes. En esta base se ubicaron los 9 sensores ultrasónicos SRF08, formando un semicírculo en las orilla, cada sensor se colocó con un Angulo de separación de 22.5° , esto con la finalidad de cubrir los 180° de visión que se necesita. Para el montaje de cada sensor se utilizó acrílico transparente cortado a medida, ver figura 2.

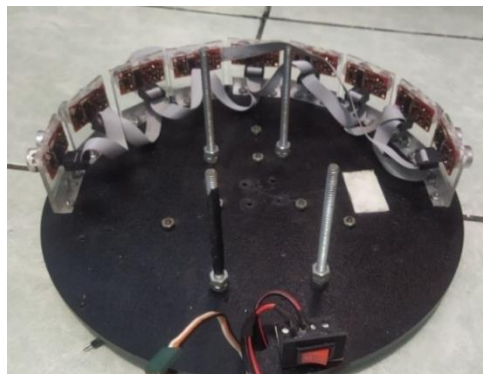


Figura 2. Montaje de base giratoria de los sensores ultrasónicos SRF08

Cuando se monta el sistema de visión sobre la estructura del móvil, se agrega la electrónica de control y visualización gráfica, teniendo un sistema como se muestra en la figura 3

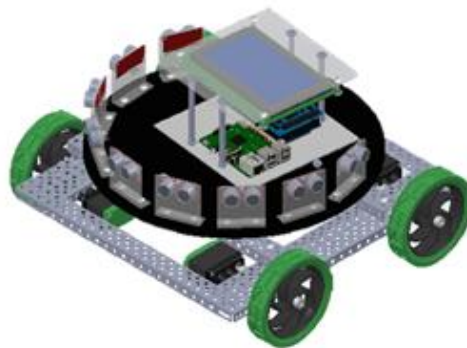


Figura 3. Diseño CAD del prototipo robótico

3. Resultados.

Para llevar a cabo el entrenamiento del sistema, se realiza la lectura de 27 muestras por elemento a analizar. El objeto a entrenar se ubica en varias posiciones alrededor del sistema de “visión”, a no más de 50 cm de distancia. Las lecturas se hace en tres rondas, ya que solo se tiene 9 sensores, cada ronda consta de 9 lecturas, guardando estas en memoria, para posteriormente rotar el sistema 7.5°, leer de nuevo y volver a rotar de nuevo los 7.5° y al final de las 27 lecturas se rota en sentido contrario los 15°. Esto se hizo con objetos cilíndricos y cuadrados, De toda la información almacenada se utilizó el 80% para entrenar a las redes neuronales, utilizando el 20% de muestras restante como datos de verificación de eficiencia del clasificador de objetos. Durante el proceso de aprendizaje se entrenaron 42 diferentes redes neuronales de retro propagación, variando las diferentes características (tipo de señal de salida, cantidad de capas y cantidad de neuronas por capa). Buscando incrementar el porcentaje de aciertos en la verificación de la red entrenada. Este incremento se puede observar en la figura 4.

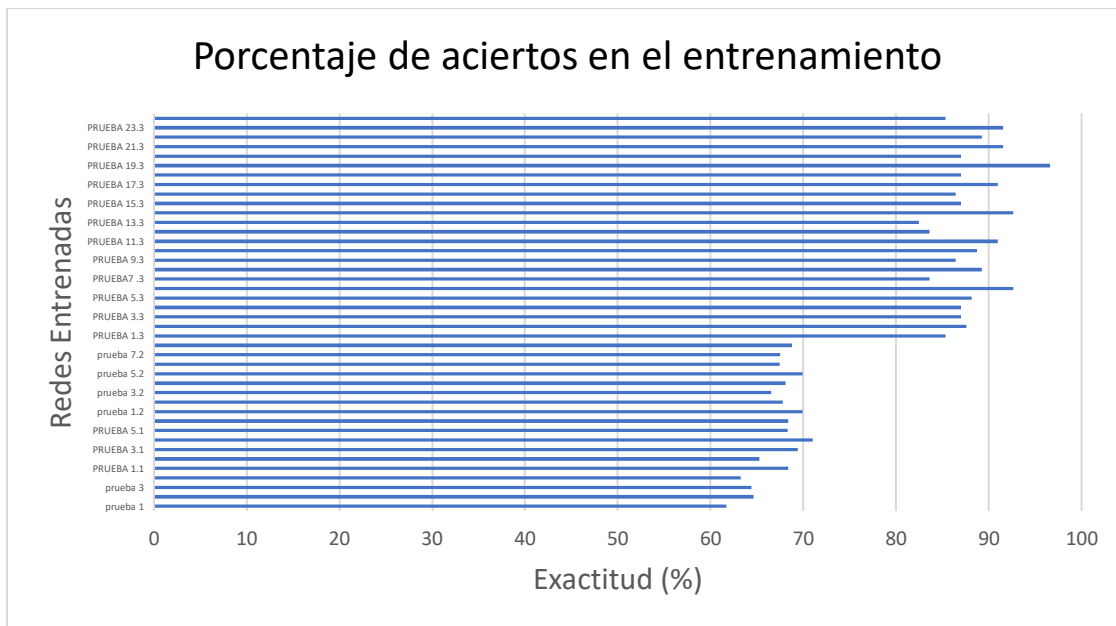


Figura 4. Comparativo de exactitud de todas las redes entrenadas.

Con los resultados obtenidos de la red neuronal artificial que obtuvo el mejor porcentaje de aciertos en el entrenamiento, se programó en el sistema de ubicación y reconocimiento para comprobación de resultados frente a datos de entrada diferentes. Para evaluar correctamente la red neuronal artificial que se había entrenado, se recolectaron 59 nuevos datos para cada objetos a identificar, obteniendo un porcentaje de acierto para cada objeto del 96.6% y que se encuentra muy aproximado al porcentaje que Matlab encuentra, ver figura 5.

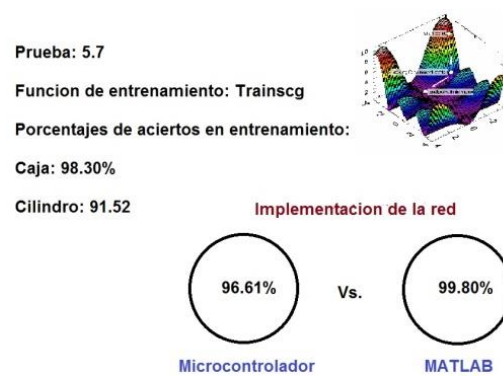


Figura 5. Comparación de la red neuronal (sistema programado autónomo vs Matlab).

Al tener resultados positivos con la red neuronal, se decidió hacer un recorrido marcando cuatro obstáculos los cuales tendrían que ser identificados por el prototipo y graficados en una interfaz gráfica desarrollada en Matlab. Para la lógica del recorrido, se programó un Atmega 2560 con las instrucciones de navegación, usando un controlador PID considerando la dirección a seguir controlada por medio de un giroscopio MPU 6050.



Figura 6. Matrices de confusión de la red neuronal artificial entrenada (prueba 5.7)

Todos los datos leídos por el sistema de visión se enviaron vía Bluetooth al programa de Matlab. Como resultado se obtuvo en el 75% de objetos presentados se logró la identificación por la red neuronal artificial. Ver figura 7.

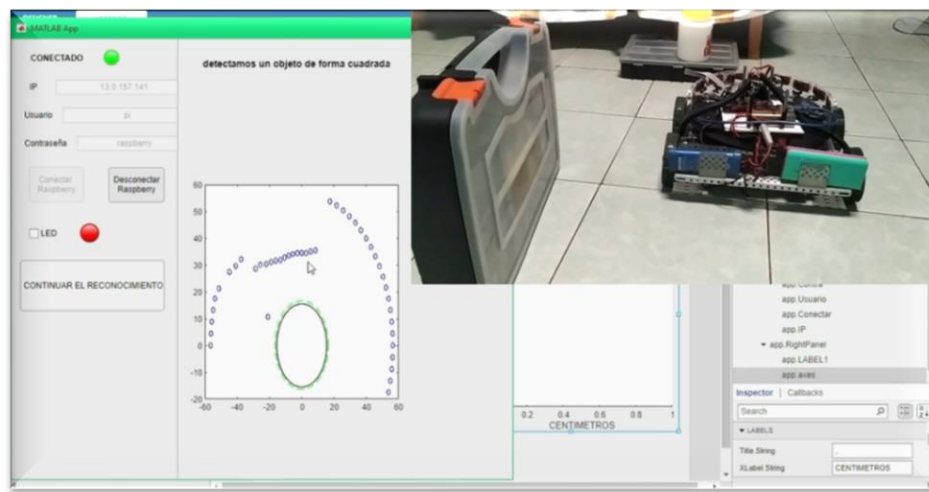


Figura 7. Identificando un objeto cuadrado en Matlab.

Discusión.

Se diseñó y entreno redes neuronales artificiales para la clasificación de objetos diversos que se le presentaban a un prototipo robótico, el cual mediante una base giratoria y nueve sensores ultrasónicos SFR08 detecta la posición del objeto en un semicírculo de 190°. Se realizaron múltiples pruebas a cada red neuronal entrenada en el sistema de cómputo numérico MATLAB. Con todos los datos recabados y almacenados en formato *.xls se eligió la red neuronal artificial con el mejor porcentaje de acierto y con ello se implementó dicha red en el prototipo robótico. Logrando implementar una red neuronal artificial en la que Matlab le atribuye un porcentaje del 99.8%, sin embargo, mediante pruebas con datos no utilizados en el entrenamiento se logró un porcentaje de acierto real del 96%.

El sistema a pesar de tener un excelente desempeño en la detección de obstáculos y el reconocimiento de patrones, tiene cierto porcentaje de incertidumbre generado por el error que introduce el uso de los sensores ultrasónicos que su método de detección está basado en un volumen frontal cónico que ocasiona estos errores al momento de procesar los vectores de entrada para la red neuronal artificial.

Créditos.

Los autores agradecen al Tecnológico Nacional de México por el financiamiento del proyecto 11162.21-P, de la convocatoria de apoyo a proyectos de desarrollo tecnológico e innovación 2021 y las facilidades del Tecnológico Nacional de México campus Tuxtla Gutiérrez para la realización de este trabajo.

Referencias bibliográficas.

- bu-Mostafa, Y. Magdon Ismail, H. Lin. 2012.** *Learning from Data: A Short Course*, United States: AMLBook
- Burkov, A. 2019.** The Hundred-Page Machine Learning Book, United States: Recuperado de: <http://themlbook.com/> Julio
- García, C. (2017).** “Localización activa de robots móviles”. Tesis de licenciatura no publicada. Universidad de la Laguna, Santa cruz de Tenerife, España.

Gonzalo, P. (2019). Aplicación de algoritmos de Machine Learning para la predicción del beneficio por cliente a partir de métricas de google analytics. Tesis de doctorado. Universidad de Valladolid, España.

Nehmzow U. (2003). Introducción. En: Robótica móvil: una introducción práctica. Springer, Londres. https://doi.org/10.1007/978-1-4471-0025-6_1.

The Mathworks (1970-2021). Redes Neuronales . MathWorks.: <https://www.mathworks.com/discovery/neural-network.html>

The Mathworks (1970-2021). Deep Learning . MathWorks.: <https://www.mathworks.com/discovery/deep-learning.html>

Ríos, A. J. (2015). “Reconstrucción de mapas para el recorrido de un vehículo sobre ruedas mediante un dispositivo de cómputo móvil”. Programación Matemática y Software. 7 (2). 41-52. Recuperado de <http://riaa.uaem.mx/xmlui/bitstream/handle/20.500.12055/118/progmat722015Reconstruccion.pdf?sequence=1&isAllowed=y>

Rodríguez, A., & Cortés, J. (2015). “Robot Oruga detector y esquivador de obstáculos con Fuzzy Logic IA”. IV congreso internacional de ingeniería mecánica y automatización: CIIMA 2015. Recuperado a partir de <https://revistas.eia.edu.co/index.php/mem/article/view/849>.

Sandoval L. (2018). Algoritmo de aprendizaje automático para análisis y predicción de datos. Revista tecnológica, vol. 11. 36-40. Recuperado de: http://www.redicces.org.sv/jspui/bitstream/10972/3626/1/Art6_RT2018.pdf

Valencia, J. C. Y., & MESA, M. A. F. (2016). “PMITO-Plataforma móvil para investigación de técnicas de odometría”. Tesis de doctorado no publicado. Universidad Tecnológica de Pereira. Pereira, Colombia.

Información de los autores.



Álvaro Hernández Sol, es Ingeniero en electrónica, egresado del Instituto Tecnológico de Tuxtla Gutiérrez ITTG, en Tuxtla Gutiérrez, Chiapas. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez y es investigador desde 1997. Certificado en SolidWorks Associate. Jefe de proyectos de investigación de ingeniería electrónica desde el 2001. Fundó y asesora el “Club de robótica del ITTG”. Colabora en la línea de investigación de “Robótica” de Ingeniería electrónica. Dirige el área de trabajo en “Robótica” y es parte del cuerpo académico “sistemas de control inteligentes”. Ha realizado investigaciones en el área de los sistemas alternativos de comunicación y en sistemas robóticos, Así como en sistemas traductores de lenguaje.



Manuel de Jesús Cancino Escobar es Ingeniero Electrónico egresado del Instituto Tecnológico de Tuxtla Gutiérrez. Durante su etapa académica fue presidente del Club de Robótica “Engineer Bots” del I.T.T.G., siendo parte del equipo de diseño para los concursos del torneo de VEX ROBOTIC temporada 2016-2019. Certificado en SolidWorks Associate. En la actualidad es alumno de tiempo completo de la Maestría en Ciencias de Mecatrónica del Tecnológico Nacional de México Campus Tuxtla Gutiérrez. Se especializa en el desarrollo de aplicaciones y programación en Python y Matlab.



Raúl Moreno Rincón, Ingeniero en Comunicaciones y Electrónica, egresado de la ESIME-IPN, en la ciudad de México, D.F. Maestro en Ciencias en Ingeniería Electrónica egresado del Instituto Tecnológico de Toluca. Maestro en Educación Superior por la Universidad Autónoma de Chiapas. Certificado en SolidWorks Associate. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez y Jefe de Proyectos de Docencia de Ingeniería Electrónica. Es investigador desde 1999 y Colabora en la línea de investigación “Robótica” de Ingeniería electrónica y es parte del cuerpo académico en formación “sistemas de control inteligentes”. Ha realizado proyectos como: Sistema de alarma para personas con deficiencia auditiva basado en XBEE, robot de cafetería, araña hexápoda, sistema de control de animatronic, entre otros.



José Ángel Zepeda Hernández, es Ingeniero Industrial en Eléctrica y Maestro en Ciencias en Ingeniería Mecatrónica, egresado del Instituto Tecnológico de Tuxtla Gutiérrez ITTG. Es profesor de tiempo completo en el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez e investigador desde 1999, Imparte cátedra en el área de Ingeniería Electrónica y la Maestría en Ciencias en Ingeniería Mecatrónica en el ITTG, donde imparte las asignaturas de Diseño con Transistores y Electrónica Básica respectivamente. Líder del cuerpo académico en formación “sistemas de control inteligentes”.



Aldo Esteban Aguilar Castillejos, obtuvo el grado de Maestro en Ciencias en Ing. Mecatrónica en el Instituto Tecnológico de Tuxtla Gutiérrez (ITTG) en el 2011 y el título de Ingeniero en Electrónica en el mismo Instituto, con especialidad en Instrumentación y Control. Certificado en SolidWorks Associate y SolidWorks Professional Avanzado, con Diplomado en Competencias para la enseñanza de las ciencias, Diplomado en Unreal Engine. Actualmente es parte del cuerpo académico en formación “sistemas de control inteligentes” del I.T.T.G. y parte del grupo de investigación en la Universidad del Valle de México, con desarrollos de proyectos financiados por CONACyT y empresas privadas. Con campos de interés en: dispositivos opto-mecatrónicos, manufactura, sensores de fibra óptica, instrumentación y control.



Osbaldo Ysaac García Ramos, Maestro en Ciencias en Ingeniería Mecatrónica egresado del Instituto Tecnológico de Tuxtla Gutiérrez. Académico de Tiempo Completo en UVM desde 2005. Tiene 12 años de experiencia docente. Tiene Cursos especialización en Mecatrónica en Universidad de Esslingen Alemania, en la empresa Emco Salzburgo Austria, en la empresa Festo Estados Unidos y Festo México. Responsable técnico del proyecto de investigación “Consolidar un centro de investigación y desarrollo de vehículos eléctricos funcionales y confortables e impulsados por energía limpia” junto con la empresa Invemex S.A de C.V., 2015. Ha desarrollado proyectos de investigación como ‘Sistema de medición de hélices de barco para la secretaria de marina financiado por Conacyt y Semar’, “Diseño y caracterización de Biomateriales a partir de macromicetos como alternativa al uso del unicel” en el año 2014 (financiados por Conacyt). Tiene participaciones en congresos internacionales como congreso Tecno láser Habana Cuba 2013 con artículo y ponencia ‘Método de registro automático de imágenes de rango tridimensionales para restauración de piezas arqueológicas’.

Sistema inteligente para tutores del bachillerato tecnológico del Estado de Morelos, México.

Smart system for tutors of the technological high school of Morelos State, Mexico.

Rodolfo Villa Noriega* (1).
Centro de Bachillerato Tecnológico Industrial y de Servicios No. 136
rodolfo.villa.noriega@gmail.com.

Francisco Javier Cartujano Escobar (2). Tecnológico Nacional de México / Instituto Tecnológico de Zacatepec,
francisco.ce@zacatepec.tecnm.mx.

Leticia Santa Olalla Ocampo (3). Tecnológico Nacional de México / Instituto Tecnológico de Zacatepec,
leticia.so@zacatepec.tecnm.mx.

Claudia Noguerón González (4). Tecnológico Nacional de México / Instituto Tecnológico de Zacatepec,
claudia.ng@zacatepec.tecnm.mx.

*corresponding author.

Artículo recibido en diciembre 21, 2020; aceptado en febrero 16, 2021.

Resumen.

La baja eficiencia terminal y bajo aprovechamiento académico son unos de los principales retos que la Educación Media Superior tiene que afrontar. En respuesta a esta problemática la Secretaría de Educación Pública en conjunto con la Educación Media Superior, ha creado desde el año 2009 el Sistema Nacional de Bachillerato, que en el marco de la Reforma Integral de la Educación Media Superior se ha implementado un programa de acciones tutoriales que atienden esta problemática a nivel operacional. El presente trabajo describe el desarrollo de un sistema inteligente que sirve como herramienta a los tutores en la etapa de análisis de información y toma de decisiones. Para extraer conocimiento de la información generada por la institución educativa y los tutores se utilizaron y evaluaron diferentes técnicas de minería de datos mediante WEKA. Posteriormente se utilizó el conocimiento generado para desarrollar una herramienta que permita predecir si un estudiante tiene altas probabilidades de desertar o tener un bajo aprovechamiento académico. Estas proyecciones son utilizadas por un Sistema Experto construido mediante JESS para sugerir a los tutores las acciones tutoriales a implementar para cada uno de los estudiantes.

Palabras clave: Minería de Datos Educacional, Sistemas Expertos, Sistemas Inteligentes, Tutoría.

Abstract.

Low terminal efficiency and low academic achievement are one of the main challenges that High School Education has to face. In response to this problem, La Secretaría de Educación Pública in conjunction with La Educación Media Superior has created since 2009 El Sistema Nacional de Bachillerato, which within the framework of La Reforma Integral de la Educación Media Superior has implemented a program of actions tutorials that address this problem at an operational level. This work describes the development of an intelligent system that serves as a tool

for tutors in the information analysis and decision-making stage. To extract knowledge from the information generated by the educational institution and the tutors, different data mining techniques were used and evaluated using WEKA. Subsequently, the knowledge generated was used to develop a tool that allows predicting if a student has a high probability of dropping out or having a low academic achievement. These projections are used by an Expert System built by JESS to suggest to the tutors the tutorial actions to implement for each of the students.

Keywords: Educational Data Mining, Expert Systems, Intelligent Systems, Tutoring.

1. Introducción.

Entre los retos más complejos del sistema educativo nacional se encuentran el abandono escolar, el bajo aprovechamiento académico y bajos índices de eficiencia terminal. En la educación media superior, estos retos, son particularmente atendidos ya que, en muchos casos, truncan el desarrollo y crecimiento personal de los estudiantes generando repercusiones negativas en la sociedad como lo son la desigualdad entre las y los jóvenes, estancamiento de la movilidad social, incremento de la pobreza y marginación, trabajos poco remunerados entre otros, que disminuyen el nivel competitivo del país (Subsecretaría de Educación Media Superior, 2014). Dentro del Sistema Nacional de Bachillerato (en adelante SNB), en el contexto de la Reforma Integral de Educación Media Superior (en adelante RIEMS), la acción tutorial es considerada como un mecanismo de apoyo para la operación y gestión del proceso enseñanza aprendizaje bajo el enfoque por competencias. Se enfoca al acompañamiento de los estudiantes durante su bachillerato con la finalidad de contribuir en su formación integral mediante actividades de orientación, asesoría y apoyo (Martínez et al., 2010).

Desafortunadamente, según lo reportado por (Santizo, 2017), existen diferencias importantes entre el diseño del programa de tutorías y su puesta en práctica, independientemente del subsistema en el que se analice. Algunas de las diferencias son: la tutoría se imparte a todo un grupo reduciendo el tiempo de atención personalizada a los estudiantes que así lo requieren, los tutores no tienen el perfil para detectar y atender las problemáticas de los estudiantes, los tutores no cuentan con una herramienta de apoyo para el análisis de datos y generan reportes en papel con un control estadístico pobre y no hay forma de utilizar esa información para implementar acciones tutoriales adecuadas.

Se ha demostrado que la forma en que se aplica el programa de tutoría si repercute en el aprovechamiento de los estudiantes y en la probabilidad de desertar (Corral, 2014) por lo que es de suma importancia buscar mecanismos que disminuyan los riesgos que puedan afectar su efectividad. Entre los riesgos tenemos tutores mal capacitados, falta de información con la que cuentan los tutores de los estudiantes, pero, sobre todo, la falta de tiempo para atender los problemas de una manera más particular (Santizo, 2017).

Al proporcionar una solución sustentada en las tecnologías de la información que facilite el diagnóstico automático de problemas académicos de los estudiantes, se ayudará a resolver en gran medida la problemática expuesta anteriormente, de tal manera que se beneficia a los estudiantes al recibir una atención tutorial en función de las variables que inciden en su deserción y aprovechamiento. El resultado del presente proyecto será posible replicarlo en planteles inscritos en el SNB como un apoyo a sus programas de tutoría institucional, independientemente del subsistema, lo cual puede aumentar el alcance de su implementación.

2. Métodos.

2.1. Minería de datos educacional.

La minería de datos es un campo de las ciencias de la computación referido al proceso que intenta descubrir patrones en grandes volúmenes de conjuntos de datos (Hernández, 2016). La minería de datos comparte una gran cantidad de técnicas con el Aprendizaje Automático por lo que resulta difícil realizar una distinción entre ambos conceptos (Berlanga, 2016). El objetivo general del proceso de minería de datos consiste en extraer información de un conjunto

de datos y transformarla en una estructura comprensible para su uso posterior. La tarea de minería de datos real es el análisis automático o semiautomático de grandes cantidades de datos para extraer patrones interesantes hasta ahora desconocidos, como los grupos de registros de datos (análisis clúster), registros poco usuales (la detección de anomalías) y dependencias. Una aplicación de minería de datos podría identificar varios grupos en los datos que luego pueden ser utilizados para obtener resultados más precisos de predicción por un sistema de soporte de decisiones. Ni la recolección de datos, preparación de datos, ni la interpretación de los resultados y la información son parte de la etapa de minería de datos, pero pertenecen a todo el proceso KDD (Knowledge Discovery in Databases) como pasos adicionales.

La Minería de Datos Educativa (en adelante EDM por sus siglas en inglés de Educational Data Mining) es la aplicación de las técnicas de la Minería de Datos a datos generados en contextos de la educación (Khare et al., 2018). Es un puente donde convergen principalmente dos disciplinas: la educación por un lado y las ciencias de la computación por el otro, donde la Minería de Datos y el Aprendizaje Automático son subáreas de las ciencias de la computación que permiten encontrar nueva información en los datos educativos (Bakhshinateght et al., 2018).

2.1.1 Knowledge Discovery in Databases.

La extracción de conocimiento está principalmente relacionada con el proceso de descubrimiento conocido como *Knowledge Discovery in Databases* (Hernández, 2016), que descubre conocimiento e información potencialmente útil dentro de los datos contenidos en algún repositorio de información. Es un proceso repetitivo que explora volúmenes muy grandes de datos para determinar relaciones y extraer información de calidad que puede usarse para encontrar modelos dentro de los datos (Deepashri et al., 2017).

Las etapas del proceso KDD descritas en la Figura 1, se dividen en 5 fases (Fayyad & Stolorz, 1997) y son:

- **Selección de datos.** En esta etapa se determinan las fuentes de datos y el tipo de información a utilizar.
- **Preprocesamiento.** Esta etapa consiste en la preparación y limpieza de los datos extraídos desde las distintas fuentes de datos en una forma manejable, necesaria para las fases posteriores.
- **Transformación.** Consisten el tratamiento preliminar de los datos, transformación y generación de nuevas variables a partir de las ya existentes con una estructura de datos apropiada.
- **Minería de datos.** Es la fase de modelamiento propiamente en donde métodos inteligentes son aplicados con el objetivo de extraer patrones previamente desconocidos, válidos, nuevos, potencialmente útiles y comprensibles y que están contenidos u “ocultos” en los datos.
- **Interpretación y evaluación.** Se identifican los patrones obtenidos y que son realmente interesantes, basándose en algunas medidas y se realiza una evaluación de los resultados obtenidos.

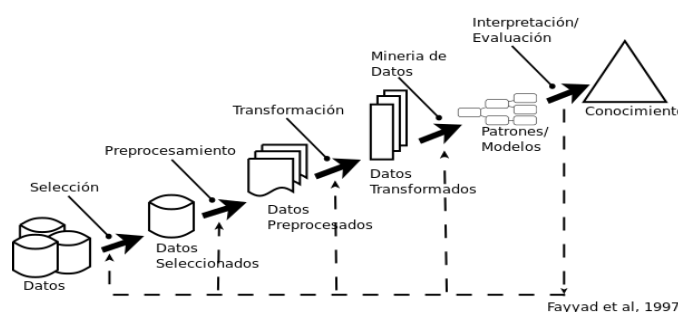


Figura 1. Proceso KDD.

2.1.2 Técnicas de Minería de Datos.

Las técnicas de la minería de datos provienen de la inteligencia artificial y de la estadística. Dichas técnicas no son más que algoritmos más o menos sofisticados que se aplican sobre un conjunto de datos para obtener unos resultados. Las técnicas más representativas según Hernández Cedano (2016) son:

- **Redes neuronales:** Son un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas en una red que colabora para producir un estímulo de salida.
- **Regresión lineal:** Es la más utilizada para formar relaciones entre datos. Rápida y eficaz pero insuficiente en espacios multidimensionales donde puedan relacionarse más de 2 variables.
- **Árboles de decisión:** Un árbol de decisión es un modelo de predicción utilizado en el ámbito de la inteligencia artificial. Dada una base de datos se construyen estos diagramas de construcciones lógicas, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que suceden de forma sucesiva, para la resolución de un problema.
- **Agrupamiento o clustering:** Es un procedimiento de agrupación de una serie de vectores según criterios habitualmente de distancia; se tratará de disponer los vectores de entrada de forma que estén más cercanos aquellos que tengan características comunes.

2.2 Sistemas inteligentes basados en reglas.

Los sistemas basados en reglas constituyen un campo de estudio importante dentro de la Inteligencia Artificial ya que nos ayuda a capturar la experiencia humana en la resolución de problemas, con el fin de alcanzar decisiones consistentes y repetibles. En ellos la representación del conocimiento se identifica por medio de la heurística o formas de proceder de los expertos. Son interesantes, especialmente en aquellos dominios en donde escasean los expertos como por ejemplo medicina, ingeniería, etc. ya que proporcionan un medio eficaz para difundir ampliamente razonamientos escasos y específicos (Ramírez et al., 2016).

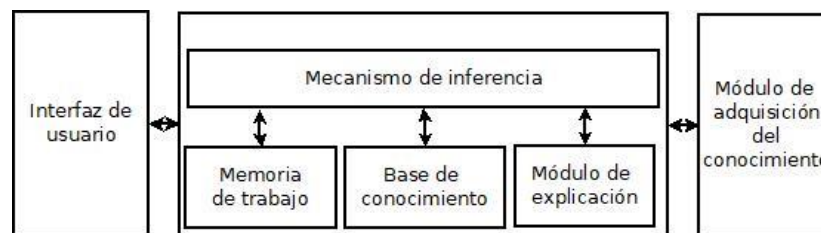


Figura 2. Arquitectura de un sistema experto.

Un sistema basado en reglas según (Ferrer et al., 2015), es un sistema que contiene los elementos mostrados en la Figura 2 y se describen a continuación:

- **Una base de conocimiento (BC):** contiene el conocimiento extraído del experto en un dominio específico. Para realizar la representación del conocimiento se han desarrollado diversas formas como lo son las reglas de producción, marcos, las redes semánticas, entre otras.
- **Mecanismo de inferencia (MI):** selecciona las reglas que se pueden aplicar y las ejecuta, con el fin de obtener alguna conclusión, es decir, realizar un procesamiento o interpretación del conocimiento controlando el proceso de razonamiento. Su objetivo es encontrar conclusiones aplicando el conocimiento almacenado en la base de conocimiento a los datos de entrada.
- **Base de hechos (BH) o memoria de trabajo:** acumula un conjunto de hechos establecidos, que se usan para determinar qué reglas puede aplicar el mecanismo de inferencias.
- **Interfaz de usuario:** es el puente entre el sistema experto y el usuario no experto. Permite al usuario describir al sistema el problema al ingresar información en forma de hechos, por otro lado, permite también al sistema presentar los resultados de forma que el usuario pueda entenderlos.
- **Módulo de adquisición del conocimiento:** permite al experto modificar la base de conocimiento ya sea actualizando el conocimiento ya existente o agregando nuevo conocimiento.
- **Módulo de explicación:** es utilizado para que el sistema experto describa el procedimiento que utilizó para llegar a una conclusión. Es útil cuando se requieren tomar decisiones basados en los resultados del sistema experto.

Existen implementaciones de sistemas expertos que no contienen base de conocimiento, estos son conocidos como *Shells* y permiten desarrollar sistemas expertos constituyendo una herramienta de apoyo en el proceso de creación de bases de conocimiento y facilitar la utilización de los sus componentes al usuario (Prada, 2016).

Jess es un motor de reglas para la plataforma Java. Para usarlo, se especifica la lógica en forma de reglas utilizando uno de dos formatos: el lenguaje de reglas Jess o XML. También proporciona algunos de sus propios datos para que las reglas operen. Las reglas pueden crear nuevos datos, o pueden hacer cualquier cosa que pueda hacer el lenguaje de programación Java.

3. Desarrollo.

3.1. Arquitectura del sistema.

En la Figura 3 se muestra la arquitectura general de la aplicación donde se definen las tecnologías que se utilizaron y los componentes principales del sistema.

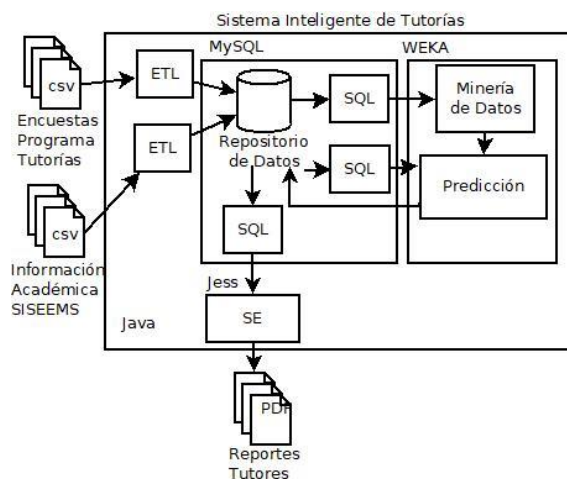


Figura 3. Arquitectura del sistema.

El sistema utiliza una base de datos relacional en MySQL donde se depositan los datos de las fuentes de información. Los datos son cargados por un módulo de la aplicación que permite seleccionar los archivos de datos para procesarlos, limpiarlos, transformarlos (en caso necesario) e insertarlos en la base de datos. Posteriormente se diseñaron consultas SQL que permiten obtener *DataSets* para los procesos de minería de datos y para el sistema experto.

El proceso de minería de datos es realizado con Weka. Este proceso permite obtener un modelo el cual es utilizado para obtener una predicción de cada uno de los estudiantes, dicha predicción también es almacenada en el repositorio de datos.

El sistema experto se desarrolló con el *shell* Jess. Utiliza los datos de los estudiantes, así como la predicción realizada por Weka. Este módulo genera listas de estudiantes para cada una de las acciones tutoriales.

Por último, se desarrolló una interfaz gráfica que permite automatizar cada uno de los procesos descritos anteriormente.

3.2 Minería de datos.

Para la generación del modelo de predicción se utilizó la técnica de redes neuronales siguiendo las fases del proceso KDD, descritas a continuación:

- **Selección de datos.** En esta etapa de seleccionaron las fuentes de información. Para este proyecto se determinaron dos fuentes de información:
 1. Datos de estudiantes que obtiene el programa de tutorías del bachillerato tecnológico. Los tipos de datos incluidos en esta fuente de información son personales, socioeconómicos, de hábitos de estudios, estilos de aprendizaje e información acerca de la convivencia entre los estudiantes. Para este estudio se utilizaron dos muestras de información. la primera con 240 registros con 143 atributos con información personal y socioeconómica del semestre febrero – junio del 2018. En la segunda muestra se obtuvieron 458 registros con 143 atributos del semestre agosto – diciembre del 2018.
 2. Datos académicos. Esta fuente de información incluye calificaciones y asistencias parciales por materia y, datos personales obtenidos del CURP de estudiante. En total se cuentan con 13752 registros producto del proceso de evaluación del semestre enero – junio del 2018 con 25 atributos.
- **Preprocesamiento.** Se desarrollo un proceso ETL que consta con dos componentes:
 1. Una base de datos implementada en MySQL que almacena dos datos preprocesados de las dos fuentes de información.
 2. Una aplicación desarrollada en Java la cual procesa las fuentes de información evalúa los datos obtenidos de las evaluaciones del programa de tutorías y carga la información a la base de datos.
- **Transformación.** Se obtuvo el género y la edad del CURP de los estudiantes para complementar los datos ya existentes. El conjunto de datos presentó el problema de clases desbalanceadas. Para evitar un sobre aprendizaje en el proceso de la construcción del modelo de predicción se balancearon las clases mediante el uso del filtro *Resample* de Weka, que generó una submuestra de los datos de forma aleatoria Villa-Noriega (2019). Por último, se seleccionaron 35 atributos descritos en la Tabla 1, con los cuales se formó el conjunto de datos para la siguiente fase. El total de registros generado en esta fase fue de 4239 registros.

Tabla 1. Atributos del conjunto de datos.

N	Atributo	N	Atributo	N	Atributo
1	edad	13	trabaja	25	resultadoActitudClase
2	genero	14	ingresoFamiliar	26	resultadoCondicionesExternas
3	tipoEscuela	15	tipoVivienda	27	resultadoMetodoPersonal
4	promedioSecundaria	16	viviendaPropia	28	resultadoProgramacionPersonal
5	razonIngreso	17	tieneComputadora	29	resultadoConvivencia
6	padeceEnfermedad	18	tieneInternet	30	resultadoNuevoIngreso
7	vivePadre	19	tiempoTraslado	31	carrera
8	escolaridadPadre	20	gastoTransporte	32	materia
9	viveMadre	21	resultadoAuditivo	33	semestre
10	escolaridadMadre	22	resultadoVisual	34	turno
11	vivePadres	23	resultadoKinestesico	35	tipoAcreditacion
12	sostenEconomico	24	resultadoVAK		

- **Minería de datos.** Para la generación del modelo de predicción se utilizaron redes neuronales mediante el algoritmo *MultiLayerPerceptron* de Weka y una validación cruzada con 10 particiones para verificar su precisión.
- **Interpretación y evaluación.** El algoritmo generó un modelo de predicción con el 97.42% de precisión con un total de 4130 instancias clasificadas correctamente y 109 incorrectas. En la Tabla 2 se muestran los resultados de la validación del modelo donde se detalla el número de instancias que se clasificaron de cada una de las clases.

Tabla 2. Matriz de confusión de la red neuronal.

CLASES	A	NA	NP	Predicción Correcta	Predicción Incorrecta
A	1325	78	10	93.8%	0.7%
NA	19	1392	2	98.5%	0.3%
NP	0	0	1413	100%	0.04%

3.2 Aplicación del modelo predictivo.

Uno de los objetivos del presente trabajo es generar una herramienta para el apoyo a la toma de decisiones por parte del docente tutor. Esta herramienta deber realizar un análisis de estudiantes de manera automatizada, obteniendo una predicción de su deserción. Para lograr el objetivo anteriormente mencionado, es necesario realizar la predicción aplicando el modelo generado en Weka a los datos del estudiante. Posteriormente almacenar dicha predicción en el repositorio de datos para su posterior utilización. Para ello se generó otro módulo en la aplicación Java que permite obtener los datos de los estudiantes, evaluar a cada estudiante con el modelo y finalmente guardar la predicción.

En el Algoritmo 1 podemos observar el código Java donde se aplica el modelo predictivo a cada uno de los registros de los estudiantes. Se obtiene el modelo generado en Weka en un objeto *Classifier* con la finalidad de poder realizar la clasificación de forma masiva. Posteriormente se obtiene el modelo de datos de un archivo *ARFF* con los datos de los estudiantes idéntico al *DataSet* con el que se generó el modelo. Se leen los datos de los estudiantes para clasificarlos con el método *classifyInstance()* de la clase *Classifier*. Por último, se guarda la predicción de cada instancia en la base de datos para su utilización en el futuro.

Algoritmo 1. Aplicación del modelo de predicción en Java

```
Classifier cls = (Classifier) weka.core.SerializationHelper.read("Clasificador.model");
```

```
model = new Instances(  
    new BufferedReader(  
        new FileReader("Archivo.ARFF")));
```

```
for (int i = 0; i < model.numInstances(); i++)  
{  
    Instance inst=model.instance(i);  
    double pred = cls.classifyInstance(inst);  
    data[i][45]=inst.classAttribute().value((int) pred);  
    CalificacionesSiseemsPredictions().save(data);  
}
```

3.3. Implementación de un sistema inteligente.

En el programa de tutorías del bachillerato tecnológico no existe un Plan de Acción Tutorial. La Subsecretaría de Educación Media Superior ha proporcionado documentos base donde se establece que es responsabilidad del docente tutor definir las acciones tutoriales a partir del diagnóstico de los estudiantes y la identificación de las necesidades particulares de cada estudiante. Para definir las acciones tutoriales del sistema se realizaron mesas de trabajo con los docentes tutores. En estas mesas de trabajo los docentes tutores definieron 137 acciones tutoriales.

3.3.1. Reglas del sistema experto en Jess.

Continuando con las mesas de trabajo de los docentes tutores, una vez que definieron las acciones tutoriales, determinaron los valores de las características de los estudiantes para cada una las acciones tutoriales formando

reglas de producción. Estas reglas se transformaron a reglas en el lenguaje CLP que es el que utiliza el *shell* Jess. En la Figura 4 podemos observar un fragmento de la plantilla con las características de los estudiantes y sus posibles valores en lenguaje CLP. Al final del análisis se llegó a un total de 47 slots definidos para utilizarse en el sistema experto.

```
(deftemplate Estudiante
  (slot numeroControl)
  (slot generacion) ;{2016-2019,2015-2018,2017-2020,2018-2021}
  (slot edad) ;numeric
  (slot genero) ;{H,M}
  (slot tipoEscuela) ;{Pública,Privada}
  (slot promedioSecundaria) ;numeric
  (slot razonIngreso) ;{Convicción,Padres,Amigos,Otro}
  (slot periodo) ;{'SEMESTRAL 2 - 2017 ','SEMESTRAL 1 - 2018'}
  (slot padeceEnfermedad) ;{No,Si}
  (slot vivePadre) ;{Si,No}
  (slot escolaridadPadre) ;{Secundaria,'Preparatoria o bachillerato',Primaria,Licenciatura,'Ma
  (slot viveMadre) ;{Si,No}
  (slot escolaridadMadre) ;{'Preparatoria o bachillerato',Primaria,Secundaria,'Sin escolaridad
  (slot vivePadres) ;{Si,No}
  (slot sostenEconomico) ;{Ambos,'Padre o tutor','Madre o tutora',Yo,'Un familiar'}
  (slot trabaja) ;{No,Si}
  (slot ingresoFamiliar) ;{'$2500 - $5000','Menos de $2500','$7500 - $10000','$5000 - $7500','
  (slot tipoVivienda) ;{Casa,Apartamento,'Vivienda rural'}
  (slot viviendaPropia) ;{No,Si}
  (slot tieneComputadora) ;{Si,No}
  (slot tieneInternet) ;{Si,No}
  (slot tiempoTraslado) ;numeric
```

Figura 4. Definición de la plantilla Estudiante en Jess.

Posteriormente se definieron las reglas con el lenguaje CLP en el cual se utilizaron las características de los estudiantes para definir cada una de las acciones tutoriales a realizar. Una de las tareas medulares del docente tutor es realizar una intervención cuando el estudiante presenta deficiencias en su rendimiento académico el cual puede ser de una determinada materia. En la Figura 5 se observa un ejemplo de la definición de una acción tutorial de canalización académica de la asignatura de Álgebra. La regla define las características que los estudiantes deben de cumplir para que el sistema recomiende la asesoría académica como acción tutorial. El estudiante debe de cursar la carrera “Componente básico y propedéutico”, cursar la materia de “Álgebra”, tener “Insuficiente” por lo menos en uno de los resultados de la actitud en clase, condiciones de estudio externas, método de estudio y programación personal, además haber sido clasificado como “Riesgo moderado” o “Riesgo Alto” en el perfil de deserción de nuevo ingreso. Por último, si el estudiante en la evaluación del primer parcial obtuvo menos de ocho o si en los valores obtenidos por los modelos de predicción en la acreditación es posible que repruebe o deserte entonces se canaliza a asesorías académicas de álgebra.

```
(defrule asesoria-algebra
  "Asesoría de Álgebra"
  (Estudiante {carrera == "COMPONENTE BASICO Y PROPEDEUTICO"
    && materia == "ÁLGEBRA"
    && (resultadoActitudClase == "INSUFICIENTE"
      || resultadoCondicionesExternas == "INSUFICIENTE"
      || resultadoMetodoPersonal == "INSUFICIENTE"
      || resultadoProgramacionPersonal == "INSUFICIENTE")
    && (resultadoNuevoIngreso == "Riesgo moderado" || resultadoNuevoIngreso == "Riesgo alto")
    && (call < 8 || (tipoAcreditacionP == "NA" || tipoAcreditacionP == "NP"))
  })
  =>
  (printout t "Canalizar a las asesorias de Álgebra" crlf))
```

Figura 5. Regla para la acción tutorial de canalización académica.

Otro de los objetivos de la tutoría en el bachillerato tecnológico es realizar el acompañamiento del estudiante desde que inicia sus estudios hasta que los finaliza. En la regla de la Figura 6 se define una acción tutorial de canalización a

asesorías académicas de una materia de especialidad. La diferencia entre esta regla y la anterior es que esta última pide que el estudiante pertenezca a la especialidad de “Programación”, y que no se utilizan los resultados del test de perfil de ingreso. Esto es debido a que esta materia pertenece al tercer semestre, por lo que el estudiante pudo haber obtenido un resultado en el perfil de ingreso que no describe al estudiante en el presente.

```
(defrule asesoria-programacion-orientada-eventos
  "Asesoría de APLICA LA METODOLOGÍA DE DESARROLLO RÁPIDO DE APLICACIONES CON PROGRAMACIÓN ORIENTADA A EVENTOS"
  (Estudiante (carrera == "PROGRAMACIÓN"
    && materia == "APLICA LA METODOLOGÍA DE DESARROLLO RÁPIDO DE APLICACIONES CON PROGRAMACIÓN ORIENTADA A EVENTOS"
    && (resultadoActitudClase == "INSUFICIENTE"
      || resultadoCondicionesExternas == "INSUFICIENTE"
      || resultadoMetodoPersonal == "INSUFICIENTE"
      || resultadoProgramacionPersonal == "INSUFICIENTE")
    && (call < 8 || tipoAcreditacionP == "NA" || tipoAcreditacionP == "NP")
  ))
=>
(printout t "Canalizar a las asesorías de Programación con RAD Orientada a Eventos" crlf))
```

Figura 6. Regla para la acción tutorial de canalización académica de carrera.

Las reglas y las plantillas se almacenaron en un archivo de texto plano CLP con la finalidad de utilizarlas con el motor de inferencia de JESS.

3.3.2 Implementación del sistema experto en Java.

Para la implementación del sistema experto se utilizó Jess como API. Jess recibe la lista de los estudiantes con cada una de sus características que obtenemos de la información proporcionada por los estudiantes y la información que generó el sistema aplicando los modelos predictivos obtenidos con la ayuda de WEKA. Esta lista es obtenida mediante consultas SQL y contienen los datos de los estudiantes de un grupo en específico en un determinado momento por lo que está definido por el “Grupo”, “Periodo de Evaluación”, “Turno” y “Especialidad”. Estos datos son ingresados mediante un formulario en Java mostrado en la Figura 7.

Figura 7. Formulario del sistema experto.

Cada uno de los registros obtenidos en la consulta es llamado instancia. Cada instancia es enviada a Jess como un hecho. Se obtienen las instancias JESS desde la base de datos a partir de los datos anteriormente mencionados, para posteriormente evaluar cada hecho con la ayuda de una clase que contiene el método *afirmarHecho*.

La estructura principal del método *afirmarHecho* se muestra en el Algoritmo 2. Este método crea un hecho y utiliza un objeto *RETE* llamado *engine*, ambos proporcionados por el API de JESS. Mediante el objeto *engine* aceptamos hechos a la memoria de trabajo de JESS y con el método *run()* evaluamos el hecho.

Algoritmo 2. Método *afirmarHecho*

```
public void afirmarHecho(JessInstance instance) throws JessException
{
    Fact fact = new Fact("Estudiante", this.engine);

    fact.setSlotValue("numeroControl", new Value(instance.getNoControl(), RU.STRING));
    fact.setSlotValue("generacion", new Value(instance.getGeneracion(), RU.STRING));
    fact.setSlotValue("edad", new Value(instance.getEdad(), RU.INTEGER));
    ...
    fact.setSlotValue("calP", new Value(instance.getCalP(), RU.INTEGER));
    fact.setSlotValue("tipoAcreditacion", new Value(instance.getTipoAcreditacion(), RU.STRING));
    fact.setSlotValue("tipoAcreditacionP", new Value(instance.getTipoAcreditacionP(), RU.STRING));

    engine.assertFact(fact);
    engine.eval("(facts)");
    engine.run();
    engine.clear();
}
```

Por último, se utilizó un manejador de eventos de JESS donde especificamos las acciones que se realizarán cada que una regla es aceptada. En el Algoritmo 3 se muestran las acciones que definen si una regla es aceptada. Si es así, obtiene los datos del estudiante para posteriormente enviar la acción tutorial a la base de datos.

Algoritmo 3. Evento de Jess que guarda la acción tutorial

```
if (JessEvent.DEFRULE_FIRED)
{
    Activation actRule = (Activation) je.getObject();
    String nameRule = actRule.getRule().getName();

    JessInstance estudiante = new JessInstance(engine.findFactByTemplateName("MAIN::Estudiante"));
    estudiante.setAccionTutorial(nameRule);
}
```

3.3.3. Reportes.

Como se explicó en la metodología, cuando se acepta una regla del sistema experto, el sistema genera un registro en la base de datos con la acción tutorial de cada uno de los estudiantes. Este conjunto de registros son los que conforman la propuesta de plan de acción tutorial. Para comprobar la generación del plan tutorial se introdujeron hechos al sistema. Estos hechos de prueba pertenecen a los estudiantes de un grupo completo de la asignatura de Álgebra adscritos al componente básico y propedéutico. En la Figura 8 se observa la respuesta de Jess al ingresar los datos de un estudiante de este grupo. Se puede observar que el estudiante con número de control 19317051360414 está adscrito al componente básico y propedéutico, se han detectado problemas en los resultados de los test de habilidades académicas y se obtuvo “NA” (No Acreditado) en la predicción de la materia de Álgebra. En la calificación del primer parcial el estudiante obtuvo una calificación de 7 y en el perfil de nuevo ingreso obtuvo un resultado de “Riesgo Moderado”.

f-1 (MAIN::Estudiante	(resultadoVisual 65.0)
(numeroControl "19317051360414")	(resultadoKinestesico 71.66999816894531)
(generacion "2019-2022")	(resultadoVAK "AUDITIVO")
(edad 16)	(resultadoActitudClase "SUFICIENTE")
(genero "H")	(resultadoCondicionesExternas "BIEN")
(tipoEscuela "Pública")	(resultadoMetodoPersonal "INSUFICIENTE")
(promedioSecundaria 8.0)	(resultadoProgramacionPersonal "INSUFICIENTE")
(razonIngreso "Otro")	(resultadoConvivencia "DETECTADO")
(periodo "SEMESTRAL 1 - 2019")	(resultadoNuevoIngreso "RIESGO MODERADO")
(padeceEnfermedad "No")	(carrera "COMPONENTE BASICO Y PROPEDEUTICO")
(vivePadre "No")	(materia "ÁLGEBRA")
(escolaridadPadre "Preparatoria o bachillerato")	(semestre 1)
(viveMadre "Si")	(turno "VESPERTINO")
(escolaridadMadre "Preparatoria o bachillerato")	(cal1 7)
(vivePadres "Si")	(cal1P 0)
(sostenEconomico "Madre o tutora")	(cal2 0)
(trabaja "Si")	(cal2P 0)
(ingresoFamiliar "\$2500 - \$5000")	(cal3 0)
(tipoVivienda "CASA")	(cal3P 0)
(viviendaPropia "Si")	(cal 7)
(tieneComputadora "Si")	(calP 0)
(tieneInternet "Si")	(tipoAcreditacion "A")
(tiempoTraslado 40)	(tipoAcreditacionP "NA")
(gastoTransporte 30.0)	
(resultadoAuditivo 73.33000183105469)	

Figura 8. Datos introducidos en Jess.

Estas características coinciden con las características necesarias para que el sistema recomiende canalizar a este estudiante a asesorías académicas de Álgebra generando un registro en la base de datos de acciones tutoriales. Esta regla es mostrada en la Figura 4.

Una vez que el sistema ha evaluado a los estudiantes, es posible obtener las acciones tutoriales recomendadas mediante un formulario de Java mostrado en la Figura 9. Los datos necesarios para obtener las acciones tutoriales son: el periodo de evaluación, turno, especialidad y grupo.

Figura 9. Formulario para obtener acciones tutoriales.

El resultado es un archivo PDF como el mostrado en la Figura 10. Este reporte contiene una lista de acciones tutoriales con el número de control y nombre completo de los estudiantes con la finalidad de que el docente tutor

realice la implementación de dichas acciones tutoriales. En el último registro del reporte podemos observar el número de control del estudiante descrito en la Figura 8 por lo que el sistema nos sugiere que se canalice a asesorías de Álgebra. Cabe mencionar que en la figura mostrada se han ocultado parte de los datos, debido a que se utilizaron datos reales para el proyecto.

Acciones tutoriales			Periodo
			SEMESTRAL 1 - 2019
CANALIZACIÓN ASESORÍA ACADÉMICA ÁLGEBRA			
NoControl	Apellido Paterno	Apellido Paterno	Apellido Paterno
19317051360684	C	S	ABIE
19317051360434	G		YES
19317051360593	G	B	JUAN
19317051360431	M		ROG
19317051360415	L	H	ELVI
19317051360420	M	F	JESS
19317051360422	M	M	JOSE
19317051360428	M	M	LUIS
19317051360429	M	C	MARI
19317051360425	P	K	KEILI
19317051360706	R	V	BRA
19317051360592	R	V	JOSL
19317051360414	S	A	EDW

Figura 10. Estudiantes canalizados a asesoría académica.

Conclusiones.

En el presente trabajo se realizó un análisis de los datos que el programa de tutorías del bachillerato tecnológico genera. Se encontraron dos fuentes principales de datos; la generada por el programa de tutorías y la generada por el departamento de servicios escolares. Estas fuentes de datos se unieron y se consolidaron en una base de datos mediante un repositorio de datos.

Con la información almacenada y consolidada en la base de datos se generó un modelo de predicción mediante redes neuronales. El modelo generado fue utilizado para realizar un módulo de predicción de deserción de estudiantes. La predicción se almacenó en la base de datos del sistema con la finalidad de utilizar esta nueva información en la construcción de un sistema experto que genera el plan de acción tutorial para cada estudiante.

Se realizaron pruebas en un ambiente real obteniendo los siguientes resultados:

- El sistema experto permite al usuario generar acciones tutoriales a partir de los datos de los estudiantes y las predicciones de la deserción estudiantil.
- El sistema propone acciones tutoriales personalizadas para cada estudiante las cuales son consideradas como un plan de acción tutorial. Este plan es generado en documentos electrónicos para facilitar al tutor su implementación.
- El sistema reduce el tiempo de análisis de datos y propuesta de acciones tutoriales lo que permite implementar acciones tutoriales pertinentes y a tiempo.
- Cada acción tutorial propuesta por el sistema también es almacenada en la base de datos, logrando de esta manera obtener los resultados de cada ejecución en cualquier momento.

- El sistema realiza un análisis de estudiantes cuando sea necesario, inclusive cuando los datos del estudiante cambian a través del tiempo, como lo son las calificaciones parciales registradas por los docentes o actualización de los datos manejados por el programa de tutorías.
- El sistema cumple con los lineamientos de programa de tutorías de los planteles inscritos al Sistema Nacional de Bachillerato por lo que su implementación puede realizarse en más planteles dentro del Estado e inclusive del País.

Los resultados de este trabajo de investigación pueden sentar las bases para realizar trabajo futuro o mejoras como las mencionadas a continuación:

- Existen más técnicas de minería de datos que pueden aplicarse al conjunto de datos. Dichas técnicas permitirán generar nuevos modelos predictivos y descriptivos que pueden aportar más conocimiento del fenómeno del aprovechamiento académico y la deserción estudiantil.
- Es posible el desarrollo de un sistema integral que incorpore el proceso de generación de modelos de predicción al sistema mediante el API de Weka.
- El sistema experto puede ser mejorado agregando nuevas características, como lo son el encadenamiento hacia atrás y el módulo de explicación. Esta mejora permitiría mayor interacción de parte de los tutores por lo que se podrían agregar nuevas reglas o mejorar las ya existentes.
- Por último, debido a la importancia que los datos tienen en la aplicación de técnicas de minería de datos, es recomendable la implementación de un sistema que permita análisis de datos donde se obtenga más información del dominio del problema. En este proyecto se generó un repositorio de datos mediante una base de datos. Este componente puede ser mejorado con una herramienta más adecuada como lo son los Data Warehouse. Esta herramienta permite aplicar operaciones complejas a los datos que después pueden ser utilizados en la aplicación de técnicas de minería de datos.

Agradecimientos.

Especial agradecimiento al Centro de Bachillerato Tecnológico, Industrial y de Servicios No. 136 por permitir utilizar sus datos para la realización del proyecto de investigación reportado en este artículo.

Referencias Bibliográficas.

- Bakhshinategh, B., Zaiane, O. R., ElAtia, S., & Ipperciel, D. (2018).** *Educational data mining applications and tasks: A survey of the last 10 years.* Education and Information Technologies, 23(1), 537–553. <https://doi.org/10.1007/s10639-017-9616-z>.
- Berlanga, A. (2016).** *El camino desde la Inteligencia Artificial al Big Data.* Índice. Revista de estadística y sociedad, 68 (9–11). ISSN: 1697-2325.
- Corral, C. M. (2014).** *El rendimiento académico de los jóvenes en relación a diferentes estilos de acción tutorial de un Centro de Bachillerato Tecnológico* (Tesis de maestría). Tecnológico de Monterrey, México.
- Deepashri, Kamath, A. (2017).** *Survey on Techniques of Data Mining and its Applications.* International Journal of Emerging Research in Management & Technology, 6(2), 198-201. ISSN: 2278-9359.
- Fayyad, U., Stolorz, P. (1997).** *Data mining and KDD: Promise and challenges.* Future Generation Computer Systems, 13(2–3), 99–115. [https://doi.org/10.1016/S0167-739X\(97\)00015-0](https://doi.org/10.1016/S0167-739X(97)00015-0).
- Ferrer, R., Jiménez, K., Argüelles, D. & Montes, A. (2015).** *Sistema experto para la elección del tipo de recuperación en canteras de materiales de construcción.* Revista Cubana de Ciencias Informáticas, 9(3), 33–48.

- Hernández, J. Á. (2016).** *Modelo de minería de datos para identificación de patrones que influyen en el aprovechamiento académico* (Tesis de maestría). Instituto Tecnológico de La Paz, México.
- Khare, K., Lam, H., & Khare, A. (2018).** *Educational Data Mining (EDM): Researching Impact on Online Business Education*. On the Line: Business Education in the Digital Age, 37–53. <https://doi.org/10.1007/978-3-319-62776-2>.
- Martínez, A. L., Virgen, I.J., Prieto, L. N., Alcalde, L. A. (2010).** *La Tutoría en el Sistema de Educación Media Superior*. Universidad de Guadalajara. Retrieved from [http://www.sems.udg.mx/sites/default/files/BGC/Tutoria en el SEMS UDG Nov 2010.pdf](http://www.sems.udg.mx/sites/default/files/BGC/Tutoria%20en%20el%20SEMS%20UDG%20Nov%202010.pdf).
- Prada, D. (2016).** *Sistema Experto aplicado al análisis de crédito y la gestión de cartera de la empresa Masterkey LTDA* (Tesis de pregrado). Fundación Universitaria Católica Lumen Gentium, Colombia.
- Ramírez, N.V., Azuara, G., Corona, F. J., Ramirez, J. R. & Martínez, H. S. (2016).** *Sistema Experto Determinista Para El Diagnóstico De Enfermedades Eritemato-Escamosas En Prolog*. Pistas educativas, México (119), 205–216.
- Santizo, E. (2017).** *La función de la tutoría en educación media superior: Retos entre su diseño e implementación*. Reflexiones, análisis y experiencias sobre la tutoría en la educación media superior, México: Ediciones del Lirio. ISBN 978-607-28-0978-9.
- Subsecretaría de Educación Media Superior. (2014).** *Yo no abandono 9 Manual para ser un mejor tutor en planteles de educación media superior*. (9). Retrieved from https://www.gob.mx/cms/uploads/attachment/file/706/yna_manual_9.pdf.
- Villa-Noriega, R. & Cartujano-Escobar, F. J. (2019).** *Determinación del perfil de deserción en estudiantes de bachillerato mediante técnicas de predicción de minería de datos*. Congreso internacional en tecnología, innovación y docencia, 2(2), 201–208. ISSN: 2594-0465.

Información de los autores.



Rodolfo Villa Noriega, Maestro en Ciencias de la Ingeniería e Ingeniero en Sistemas Computacionales, ambos títulos otorgados por el Instituto Tecnológico de Zacatepec. Se ha desempeñado en el sector privado como desarrollador de sistemas. Actualmente docente de las carreras técnicas de Ofimática y Programación en el Centro de Bachillerato, Industrial y de Servicios No. 136, en Jojutla, Morelos.



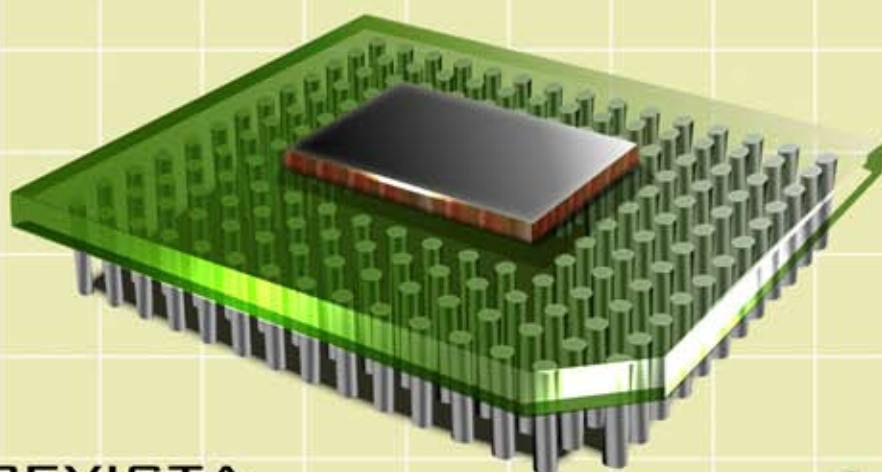
Francisco Javier Cartujano Escobar, Doctor en Administración con Especialidad en Sistemas de Información egresado del ITESM, Campus Ciudad de México; Maestro en Ciencias de la Computación e Ingeniero en Sistemas Computacionales, ambos títulos otorgados por el ITESM, Campus Cuernavaca. Obtuvo mención honorífica al terminar la carrera de Ingeniería. Se ha desempeñado en el sector privado como gerente de sistemas y como profesor investigador del Departamento de Computación del ITESM, Campus Ciudad de México. Ha pertenecido al SNI del CONACYT. Actualmente catedrático de la carrera de Ingeniería en Sistemas Computacionales y de la Maestría en Ingeniería en el Instituto Tecnológico de Zacatepec.



Leticia Santa Olalla Ocampo graduada como Licenciada en Informática en 1992 por el Instituto Tecnológico de Zacatepec (ITZ), graduada como Maestra en Ciencias de la Computación en 2008 por el Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET). Actualmente catedrática de la carrera de Ingeniería en Sistemas Computacionales y de la Maestría en Ingeniería en el Instituto Tecnológico de Zacatepec. Área de interés Ingeniería de Software.



Claudia Noguero González, Maestra en Ciencias Computacionales, egresada del Centro nacional de Investigación y Desarrollo Tecnológico (CENIDET) obtuvo el grado en diciembre de 1996, obtuvo su Licenciatura en Informática en el Instituto Tecnológico de Zacatepec en septiembre de 1993. Ha sido miembro del Sistema Estatal de Investigadores. Actualmente catedrática de la carrera de Ingeniería en Sistemas Computacionales en el Tecnológico Nacional de México campus Zacatepec. Áreas de interés todo lo relacionado a bases de datos.



REVISTA
TECNOLOGÍA
DIGITAL

www.revistatecnologiadigital.com