APLICACIONES DE LAS CIENCIAS COMPUTACIONALES EN SISTEMAS INTELIGENTES Y CIBERSEGURIDAD

María del Carmen Santiago Díaz

APLICACIONES DE LAS CIENCIAS COMPUTACIONALES EN SISTEMAS INTELIGENTES Y CIBERSEGURIDAD

APLICACIONES DE LAS CIENCIAS COMPUTACIONALES EN SISTEMAS INTELIGENTES Y CIBERSEGURIDAD

María del Carmen Santiago Díaz
Gustavo Trinidad Rubín Linares
Ana Claudia Zenteno Vázquez
Judith Pérez Marcial
Pedro García Juárez
(Editores**)**

María del Carmen Santiago Díaz
(Coordinador)

María del Carmen Santiago Díaz, Gustavo Trinidad Rubín Linares, Ana Claudia Zenteno Vázquez, Judith Pérez Marcial, Pedro García Juárez
(editores BUAP)

María del Carmen Santiago Díaz
(coordinador BUAP)

María del Carmen Santiago Díaz, Gustavo Trinidad Rubín Linares, Ana Claudia Zenteno Vázquez, Yeiny Romero Hernández, Judith Pérez Marcial, José Luis González Compeán, Alejandro Galaviz, Heidy Marisol Marín Castro, Abelardo Gómez Andrade, Paola Eunice Rivera Salas, , Ana Bertha Ríos Alvarado, María Concepción Landa Arnaiz, Osslan Osiris Vergara Villegas, Carina Toxqui Quitl, Juan Manuel González Calleros, Miguel Morales Sandoval, Abel Alejandro Rubín Alvarado, María Eugenia Narcisa Sully Sánchez Galvez, Jaime Julián Cid Monjaraz, Oleg Starostenko Basarab, Carlos Soubervielle Montalvo, Victor Manuel Morales Rocha, Germán Ardul Muñoz Hernández, Lorna Verónica Rosas Téllez, Fernando Reyes Cortes, Luis Carlos Altamirano Robles, Abraham Sánchez López, María Josefa Somodevilla García, Maya Carrillo Ruíz, Luis Enrique Colmenares Guillén, Josefina Guerrero García , Meliza Contreras González, Gabriel Juárez Díaz, José de Jesús Lavalle Martínez, Ivo Humberto Pineda Torres, Roberto Contreras Juárez, Héctor David Ramírez Hernández, Nelva Betzabé Espinoza Hernández, Rogelio González Velázquez, Pedro García Juárez, Beatriz Beltrán Martínez, Ana Luisa Ballinas Hernández, Nicolás Quiroz Hernández, Luz del Carmen Reyes Garcés, Alba Maribel Sánchez Gálvez, Bárbara Emma Sánchez Rinza, José Andrés Vázquez Flores, Hermes Moreno Álvarez, Raúl Antonio Aguilar Vera, Julio Cesar Diaz Mendoza, Juan Pablo Ucán Pech, Francisco Marroquín Gutiérrez, Jéssica Nayeli López Espejel, Eden Belouadah
(revisores)

Primera edición: 2022
ISBN: 978-607-8857-28-9

Montiel & Soriano Editores S.A. de C.V.
15 sur 1103-6 Col. Santiago Puebla, Pue.

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

Rectora:
Dra. María Lilia Cedillo Ramírez

Secretario General:
Mtro. José Manuel Alonso Orozco

Vicerrector de Investigación y Estudios de Posgrado:
Dr. Ygnacio Martínez Laguna

Directora de la Facultad de Ciencias de la Computación:
M.I. María del Consuelo Molina García

Contenido

Prefacio

El avance vertiginoso de la ciencia y la tecnología han generado un gran abanico de soluciones a diversos problemas, sin embargo, en nuestra sociedad nos encontramos inmersos en un sistema que ya no brinda el soporte para una población que ha crecido de forma tan acelerada. Por ello es imprescindible resolver problemas propios de una sociedad en constante crecimiento, a fin de generar mejores condiciones de vida a nuestra población. Aunque hay identificados a nivel nacional y local los problemas que requieren solución inmediata, casi en cualquier área se requiere realizar innovaciones tecnológicas, algunas muy sofisticadas y complejas y otras no tanto, pero finalmente innovaciones, es decir, aplicar ideas y conceptos para solucionar problemas utilizando las ciencias computacionales, que aunque en muchos casos no se requiere una solución que implique años de investigación, si se requiere que la solución esté plenamente enfocada a un problema en particular. Muchas de estas soluciones en general no están a la vista sin embargo, resolver problemas ambientales, de vialidad, de producción de alimentos, etc., representan en sí aplicar la tecnología de forma innovadora ya que aunque en nuestro país tenemos un enorme retraso tecnológico, lo que no se quiere en universidades e institutos de investigación es tener este retraso en la aplicación de la tecnología que se genera, por ello se cuenta con diversos programas internacionales, nacionales y locales para apoyar la innovación tecnológica. En nuestra universidad como en muchas otras en México y en el mundo se cuentan con programas específicos de innovación tecnológica para que laboratorios de investigación generen y apliquen tecnología propia. Pero estos esfuerzos no son suficientes, se requiere de una concientización colectiva para que desde el aula se socialice la necesidad de resolver toda clase de problemas aplicando el conocimiento impartido en clases, y no se debe esperar a una asignatura o catedra de emprendedurismo o innovación, sino desde cualquier tópico que se aborde en ciencias e ingeniería, ya que además de abstraer del mundo real el problema, se deben plantear las diversas alternativas de solución, las implicaciones tecnológicas para llevarlas a cabo, la importancia de la implementación, pero sobre todo, resolver el problema quizá por etapas o versiones hasta llegar a la solución óptima.

María del Carmen Santiago Díaz
Gustavo Trinidad Rubín Linares

Haciendo seguro el Big Data

Karen Aislinn Gonzalez Lopez, Yeiny Romero Hernández, María del Carmen Santiago Díaz, Gustavo Trinidad Rubín Linares
Benemérita Universidad Autónoma de Puebla,
Av. San Claudio y 14 sur
C.P. 72000.Puebla, Pue., México
{yeiny.romero, maricarmen.santiago, gustavo.rubin}@correo.buap.mx

Resumen. La seguridad en Big Data es de suma importancia hoy en día dado que se manejan grandes cantidades de datos en cualquier servidor e incluso en la nube que es él lugar de almacenamiento más usado en la actualidad, por ende, se deben encriptar estos datos para hacerlos seguros y manejables en el internet, este proyecto desarrollo una prueba de dos algoritmos de cifrado para probar la encriptación de grandes volúmenes de datos, el primer algoritmo de cifrado es simétrico y será programado en Java y el segundo algoritmo de cifrado será asimétrico programado en Python, con la finalidad de comprobar cuál es mejor para el big data, obtendremos un estudio sobre la efectividad de cada algoritmo. Es importante recalcar que para obtener una alta seguridad en los datos existen muchos procedimientos que se pueden llevar a cabo, sin embargo, debemos de utilizar cada uno para proteger bases de datos, correos electrónicos, archivos, etc.

1 Introducción

En esta investigación abordaremos la seguridad del Big Data, sabemos que el Big Data se encuentra conformado por enormes conjuntos de datos que hacen imposible la tarea de poder gestionarlos de forma común, por lo cual se recurren a otros métodos para poder hacerlo, dichos datos deben estar seguros para proteger la información que estamos manejando, para ello se requiere de la seguridad informática y sus procedimientos de encriptación y desencriptación, por lo que se explicará algunos de estos métodos de dichos procedimientos usados en big data para una protección integrada de los datos.

1.1 ¿Qué es el Big Data?

El término "big data" se refiere a un conjunto de datos con una gran variedad y se presentan en volúmenes crecientes y a una mayor rapidez. Esto hace difícil o imposible procesarlos con los métodos tradicionales [2], La importancia del big data no gira en torno a la cantidad de datos que tienes, sino en lo que haces con ellos. Puedes tomar datos de

cualquier fuente y analizarlos para encontrar respuestas que permitan 1) reducir los costos, 2) reducir el tiempo, 3) desarrollar nuevos productos y optimizar las ofertas, y 4) tomar decisiones inteligentes [3].

1.2 Sistemas de análisis del Big Data

Ejemplos de sistemas de análisis de presupuestos de Big Data:

- 1010data.
- Apache Chukwa.
- Apache Hadoop.
- Apache Hive.
- Apache Pig.
- Jaspersoft.
- LexisNexis Risk Solutions HPCC Systems.
- MapReduce.
- Revolution Analytics.

Hadoop: Software de código abierto que permite en los últimos años, se utiliza como analizador de datos con la mayoría de las empresas, en la actualidad casi todos los medios modernos de análisis de Big Data proporcionan un medio para la integración de Hadoop. Los desarrolladores actúan como empresas de crecimiento y empresas mundialmente conocidas.

1.3 Protección de datos, privacidad y seguridad

Dos principios básicos de la protección de datos:

- Prevención de acumulación de Big Data y minimizar los datos.
- Mercado contraste con capacidad de Big Data para facilitar el seguimiento del movimiento, el comportamiento y las preferencias de personas con gran precisión.

Un problema de ciberseguridad es que se necesita una reevaluación de amenazas y riesgos en vista de Big Data y adaptación correspondiente de soluciones de técnicas, es necesario reconsiderar la política en el ámbito de las leyes de seguridad de la información, la confidencialidad y protección de datos.

2 Descripción de algoritmos utilizados

Descripción general de los métodos de protección de datos:

- Medios de almacenamiento de protección contra corrupción de datos no autorizados o accidentales, por distorsión entender, eliminar, modificar o introducir información de terceros en el bloque de datos protegidos.

- La confidencialidad significa que el acceso de datos privados puede haber usuarios autorizados, el acceso de usuarios no autorizados debe ser excluidos.
- La identificación y autenticación permite a las partes involucradas en el intercambio de datos para identificarse a sí mismos.
- Fiabilidad.

2.1 Cifrado

Un cifrado es un par de algoritmos que implementan cada una de estas transformaciones, este algoritmo se aplica a los datos con claves, las claves para cifrar y descifrar. El descifrado hace que sea inaccesible para conocimientos no autorizados, el cifrado hace que sea imposible el ingreso de datos falsos. Los primeros métodos de cifrado usaban las mismas claves, clave privada o simétrica, esto cambió en 1976 ya que los algoritmos usaban distintas claves, clave pública o asimétrica. Esto para mantener claves confidenciales.

Fig. 1. Muestra el proceso de cifrado y descifrado de datos mediante un diagrama. Imagen tomada de pag. 51 de [1].

2.2 Hash

Una función Hash criptográfica es una clase especial de una función que tiene propiedades adecuadas para su uso en la criptografía. Es un algoritmo matemático que muestra datos de tamaños arbitrarios con una cadena de bits de tamaño fijo que sería un hash y esta es una función unidireccional, es una función inviable para invertir.
Una función hash criptográfica ideal tiene cinco propiedades básicas:

- Con el mismo mensaje determinista siempre resulta en el mismo hash.
- Es rápido para calcular el valor hash para cualquier mensaje dado.
- Es inviable crear un mensaje a partir de un valor hash, excepto tratando todos los mensajes posibles un pequeño cambio en el mensaje debe cambiar el valor hash tan extensamente que el nuevo valor hash aparece correlacionado con el antiguo valor hash es inviable encontrar dos mensajes diferentes con el mismo valor hash.

La función hash criptográfica tiene una gran cantidad de aplicaciones en aplicaciones de seguridad, en particular en la firma digital, códigos de autenticación de mensajes (MPC), y otras formas de autenticación.

En términos de seguridad de la información, los valores hash criptográficos a veces se llaman huellas digitales (digitales), sumas de comprobación, o sólo valores hash, incluso si todas estas condiciones valen funciones más generales con propiedades y propósitos muy diferentes.

Fig. 2. Muestra el proceso de la función hash mediante un diagrama. Imagen tomada de pag. 52 de [1].

2.3 MAC

MAC (código de autenticación de mensajes) es una herramienta de protección en protocolos de autenticación de mensajes con participantes de confianza, está diseñado para asegurar la integridad y autenticación de origen de los datos con un conjunto especial de caracteres que se agregan al mensaje.

MAC es utilizado para garantizar la integridad y protección de la información transmitida y va en contra de la manipulación de datos.

El verificador de integridad del mensaje en el lado del emisor, se agrega un valor de función hash de ese mensaje al mismo mensaje, al igual que en el lado del receptor.

Protección contra la falsificación MAC del mensaje, se utiliza una imitación elaborada que contiene una clave conocida sólo por el remitente y el receptor. Si el remitente no tiene la clave secreta del código hash se genera de forma incorrecta.

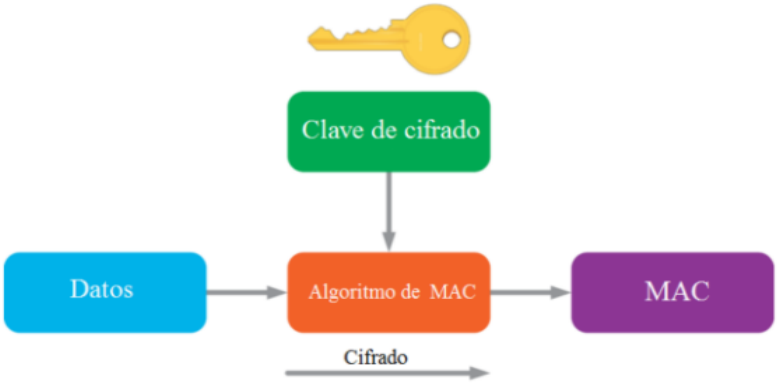


Fig. 3. Muestra el proceso del código de autenticación de mensaje mediante un diagrama. Imagen tomada de pag. 53 de [1].

2.4 Las firmas digitales

Las firmas digitales permiten la autenticación de mensajes, para probar la autenticidad, la firma digital funciona de manera como una firma no digital, esta se puede identificar al remitente. Cuando un cifrado de firma digital se utiliza asimétrico, la clave privada y el descifrado son utilizados para cifrar un mensaje abierto.

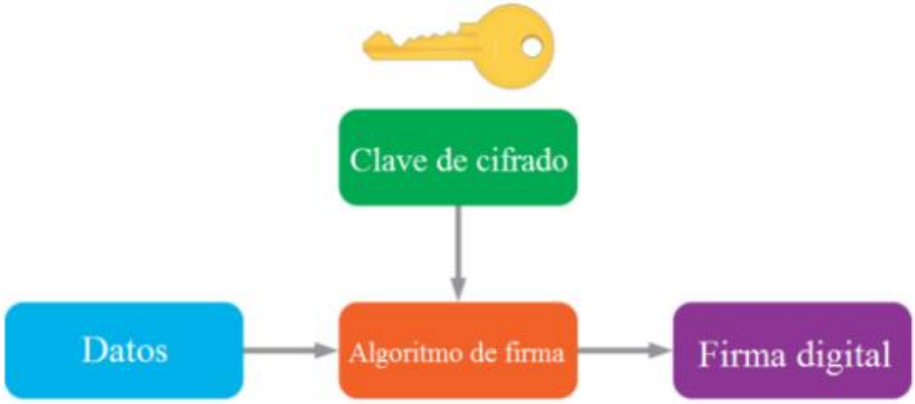


Fig. 4. Muestra el proceso de la firma digital de datos mediante un diagrama. Imagen tomada de pag. 54 de [1].

3 Metodología

Los algoritmos utilizados en esta ocasión es el cifrado simétrico y cifrado asimétrico, el cifrado simétrico se realizó con el lenguaje de programación Java en el cual se utilizaron librerías como “crypto y security”, así como otras que veremos en el código, al igual que en el cifrado asimétrico que este fue programado con Python donde también se utilizó una librería llamada “Crypto” y esta se tuvo que instalar, para hacer estos algoritmos mencionados tuve que hacer la instalación del simulador VMware Workstation 16 Player con licencia gratuita, descargue el sistema operativo CentOS 8 de Linux para poder trabajar sobre ese sistema, realice la descarga de Python 3 y Java para poder trabajar sobre lo cual requería el curso de verano.

3.1 Algoritmo Cifrado Simétrico

Para empezar, utilizaremos las siguientes librerías para poder realizar la parte de la codificación del programa, la “crypto” es para hacer encriptado y desencriptado del archivo de texto que se tomará, la “security” es para el uso de las llaves, la “io” son para las excepciones que se pueden generar, “util” en este se encuentra el Base64 que es un codificador y decodificador de texto, así también como los arreglos.

```
// Función para encriptar el texto y el cual recibe el
parámetro de la clave para poder generar la llave
private void encriptar (String clave){
try{
      Cipher cifrado = Cipher.getInstance("AES");
      this.generar_llave(clave);
      cifrado.init(Cipher.ENCRYPT_MODE, this.llave);
      byte[]cifradoBytes
      cifrado.doFinal(this.texto.getBytes());
      this.textoBase64.getEncoder().encodeToString(cifra
      doBytes);

      System.out.println("Texto Cifrado: " + this.texto);
      }
          catch (Exception ex){
                      System.out.println(ex);
                }
          }
// Función para desencriptar el texto ya encriptado
private void desencriptar (){
try{
          Cipher cifrado = Cipher.getInstance("AES");
          cifrado.init(Cipher.DECRYPT_MODE, this.llave);
       byte[]cifradoBytescifrado.doFinal(Base64.getDecod
       er().decode(this.texto));
          this.texto = new String(cifradoBytes);
```

```
        System.out.println("\nTexto   descifrado:   "   +
        this.texto);
                }
                catch (Exception ex){
                      System.out.println(ex);
                }
          }
// Función que hace la acción de cifrar el archivo de
texto plano, el cual cuenta con dos parámetros que son la
ruta del archivo de texto y la clave elegida para la
generación de la llave.
        private void cifrar (String direccion, String
          clave){
        this.abrir_archivo(direccion);
        this.encriptar(clave);
        this.guardar_llave("llave.dat");
        this.guardar_archivo("texto_encriptado.txt");}

        private void descifrar (String direccion){
        this.abrir_archivo_encriptado(direccion);
        this.abrir_llave("llave.dat");
        this.desencriptar();
        this.guardar_archivo("texto.desencriptado");
```

Con base a la documentación de las librerías utilizadas pude realizar un algoritmo donde se utiliza un cifrado simétrico con el lenguaje de programación de Java con el cual puedes cifrar y descifrar archivos de texto, al momento de encriptar el texto genera un llave con la cual se va poder desencriptar el texto y generar archivos ya sea con el texto cifrado o descifrado y de la llave.

3.2 Algoritmo Cifrado Asimétrico

Para algoritmo se utilizó la librería “Crypto” que se instala con (pip install pycryptodome - es la librería actualizada de “Crypto” pero se importa de la misma manera que la anterior), en este algoritmo se utilizó el lenguaje de programación Python que es muy conocido debido a que es un lenguaje de alto nivel y de los más utilizados, ya que puede utilizarse en muchos sectores.

```
# Función para encriptar un archivo que tiene como parámetro la ruta del archivo a
encriptar
        def encriptar_archivo (self, rutaArchivo):
        super().abrir_archivo_texto(rutaArchivo)
```

```
            super().encriptado()
            super().guardar_archivo_encriptado()
            super().guardar_llave_privada()

    # Función de desencriptación de un archivo de texto, con parámetros de la ruta del
    archivo encriptado y la ruta de la llave privada generada
            def desencriptar_archivo (self, rutaArchivo, rutaLlave):
            super().abrir_llave_privada(rutaLlave)
            super().abrir_archivo_encriptado(rutaArchivo)
            super().desencriptado()
            super().guardar_archivo_desencriptado()

            accion = Accion()
            accion.encriptar_archivo("texto.txt")
            accion.desencriptar_archivo("texto_encriptado.bin", "llave_privada.pem")
```

Este algoritmo fue diseñado en base a la documentación de la librería que fue utilizada para poder hacer el proceso de cifrado y descifrado asimétrico y así poderlo combinar con la función de generar archivos de texto encriptado y desencriptado y está hecho en Python para que fuera distinto al cifrado simétrico que fue hecho en Java.

3.3 Comparación de algoritmos

El algoritmo de cifrado asimétrico cuenta con una llave general, un generador aleatorio de bytes, una llave pública, una llave privada y genera una etiqueta y un token al momento de cifrar el texto plano, únicamente la llave privada se utiliza al momento de querer descifrar el texto cifrado, al momento de querer guardar archivos encriptados con utf-8 se tiene una función aparte de como guardar un texto desencriptado.
El algoritmo de cifrado simétrico tiene una única llave que se genera a través de una clave que decidas, el codificar y decodificar se hace a través de base64 esto para que no se dañe la matriz generada, los archivos de texto se abren distintas funciones, solo el de guardar es el mismo para ambos, la llave si tiene una función aparte para abrir y guardar, aparte que se tiene que generar una nueva cadena al momento de desencriptar el texto.

4 Resultados

En estos algoritmos se pueden encriptar archivos de texto con extensión “.txt” y datasets “csv”.

4.1 Cifrado simétrico

En la primera prueba se puso un archivo con formato ".txt" donde contiene poco texto y pocos saltos de línea, el resultado fue satisfactorio.

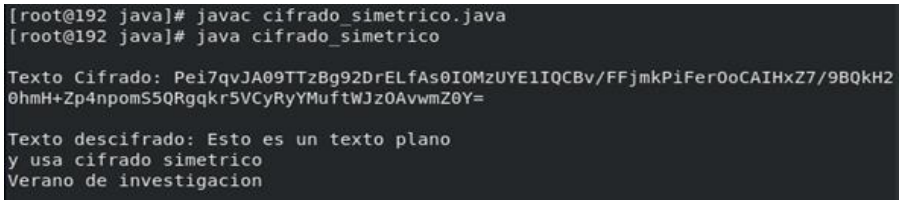


Fig. 5. Probando el código con un texto de formato ".txt", que contiene 3 líneas de texto y el resultado fue muy rápido.

En esta segunda prueba utilizamos un dataset con formato ".csv" donde contiene un peso de 191 KB, ya que aumentaría el contenido del texto a encriptar y desencriptar.

Fig. 6. Probando el código con un texto de formato ".csv" llamado Places Near Harlem que es un dataset de Kaggle y el resultado fue bastante rápido.

Fig. 7. Se muestra un poco del texto descifrado del dataset que utilizamos para esta prueba del algoritmo.

La tercera prueba fue aumentando a 2 MB con otro dataset descargado de Kaggle y se trata sobre la venta de videojuegos con calificaciones.

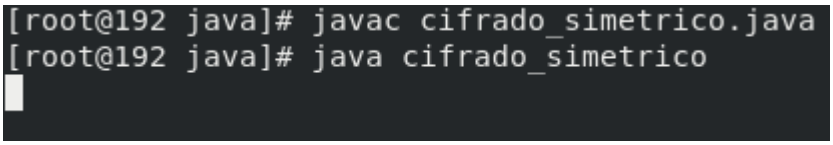


Fig. 8. La prueba con este peso en el algoritmo hizo que se tardara varios minutos en poder encriptar y desencriptar el texto.

Al algoritmo le tomó aproximadamente 5 min para poder obtener la encriptación y desencriptación del texto, fue de un gran tamaño que la consola no pudo visualizar todo el contenido del dataset.

Fig. 9. Es una parte de lo que solo alcanzo a visualizar la terminal.

Se puso a prueba un dataset que fuera menor al anterior, pero mayor que el primer dataset, en este caso fue de 734 KB que trata sobre datos de préstamos en Kaggle.

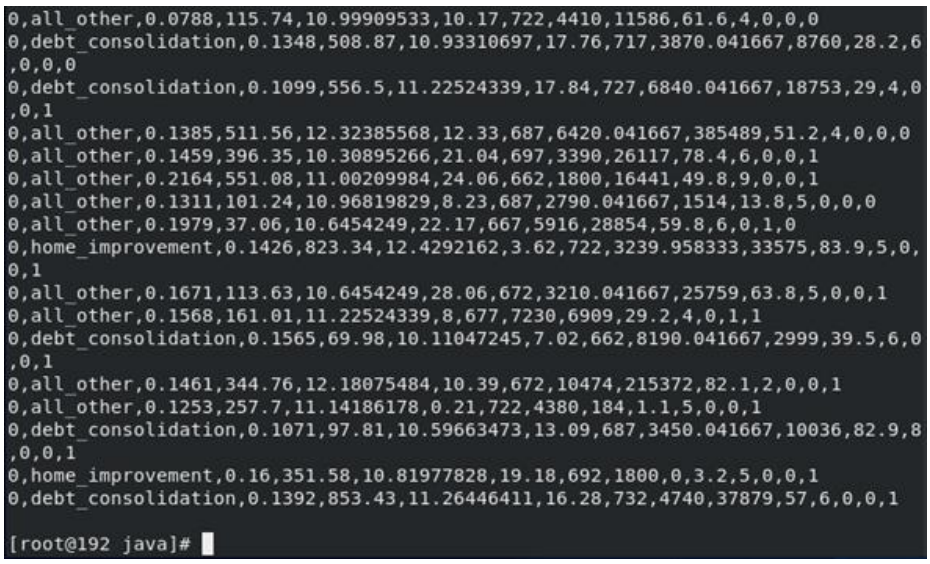


Fig. 10. Es una parte de lo que solo se alcanzó a visualizar la terminal de un dataset de Kaggle.

4.2 Cifrado asimétrico

En esta primera prueba el algoritmo fue muy veloz al procesar esto, en encriptar y desencriptar el texto de un archivo.

```
(big_data) [root@192 seguridad]# vi cifrado_asimetrico.py
(big_data) [root@192 seguridad]# python cifrado_asimetrico.py

Mensaje Cifrado:  b'\x0e\x0e\x135\xd6/\x1c\xa4\xdb\xfe\x1c\x80\x18\xdc\x062";q\x
ea\xef\xe3\x954\x0c|v\x19o\x843l\x04\xebT/\xcf\xc3\xf9\x03\x15Y\xacNG\x0f\xf7\xb
4\x96tu\x99\xc9\xb7\xcc\xe25(\xc4\xccg+yl5\xca\xed\xf5{\xf1\x03\xa9}\xfe0\xb6\xf
d-\x11\x1f\x9d\x80V\xc1\x18\x97Zr\xb8\x9f\xba\xd6[\x14T\xcc\xb3'

Mensaje Decifrado:  Hola esto es un texto plano
Y esta con cifrado asimetrico
con seguridad
Verano de investigación
```

Fig. 11. Se muestra el resultado obtenido en la primera prueba que es con texto plano y formato ".txt".

La segunda prueba es cuando le ponemos el primer dataset, que ya habíamos probado en el algoritmo de Java que contiene un peso de 191 KB.

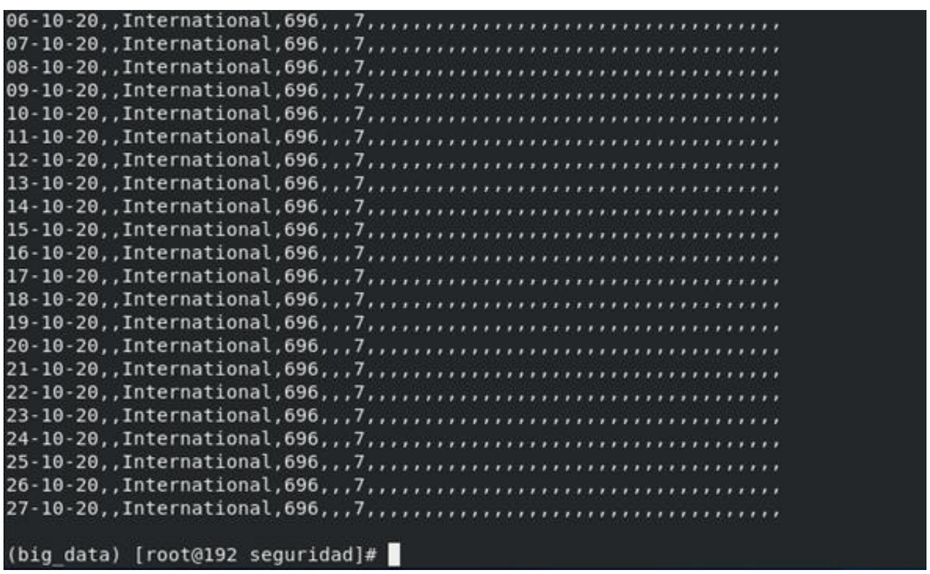
```
(big_data) [root@192 seguridad]# python cifrado_asimetrico.py

Mensaje Cifrado:  b'\xb9>\x00\x1c\xdag\x95\xe8\x9f$V\xbb\xbe\xcf\xd5Gi\xf1\xa4\x
9d=\xef1<\xa8 b\xf8:\\b\x9f/\xf3\x96S\xbcS\x84\x16\xde\xb2M7\xf5\xb7R\xbb\x97\x0
7\xff\xd6\x9fr]qebu\xd9\x92\xbc\xce\x9d\x0c\xba~\xb5[\x00\x8albi\xb8\xb7\xb5\xd1
\xcd\x95^\x9a\xb6\'\xc7\x9c\x04J\xce|\xb3=\x8b\xfd\x18\xf5\xfb\x9dM\x82;\x91S\xc
d\x88\no[`\x00\xb4t\x9d\x96\x81\x18:\xe5z\x80P\x0e\xf2\x11^\x1fY\xd8\xebC\xd7\x9
5\xcdG\xf7\x16\x1f\x97\x1b\xb2\x1an\xab{\xbc\xfdV\xae\x01\xf3\xc04 H!\x8cW\xc0\x
f8\xc3Gg\xac,\x9c\t\x18\xdf\xa8\x01\xaf)\xebA\x9bG\xeeX\xef\xaf\x16\xc5rXf\xeb\x
de1\xad\xf7NN\x13\xe4\xdd\xb1\x91@x\x03\x98\x04*\xa9\xe2\x1b\xd5\xcdi~\xc68\xff\
x9aj\xffb\x93\x1a0\x17:\xf3\xb3.\x9d\x98\xc3\xa8\xbfYa\x0f\x1a\xac\x85\xa30\x96\
xd4\xb0\xfd\x1a\x05Zr\xb9\xcc\xa6\xb2\xef{\xcd\xc8\xe8z\xdaH\x94\xc9\x15\x91\xd1
\xf5\xc7m\x06\xe9\xea\x1d\xba\x8e4\xfe\xf4\xf3\xc6[>\x06\xc0`\x1eL]\r\x85\xf40\x
e0\x05m:\x9eL\xa7\x90*73\n\x14\xf5C\x8f\xa9b\x97\xb1\xbbq\xfa\x97\x84\xea\xf5\x9
4\xec\n\xd2\x88&\xe8\xa6\x9a}mZ\xfflY]\xf0=hZ\x8e\x12\xfb\xea\xaba\xc5b,\xb1\x0b
v\xab\x851\xb0B\xfd\x9cy\xca[e\x0f\xc2;\xf4\x82PA\xca7r\xb54\xb4\xef\xea\x9e\x10
\xc9!\xcb\x8bs\xd4;r\xf8\xa1\xb4\xa7\x99\x05\xfa%\xa9\xbfOp\x1c\xde\xca\x14\xbb!
\x0c\xaf\xb53\x02g\xd3Ci\xbf^\x13\x84\x0b\x99Io\xcc\x90\xf9\x94T\xb7\x86\x9e\xda
\x8cH\xfe\x0f<\x91\xd2\x19\xf6~\x0c\x90\xdf\xbfoS\x97\xe0\t\xfal7\xa3\xbd"w^[m\x
f7z\x043\xda\x02\xe9\x11l\x89u\xe1\x02"\xb0\x92W\xa3\xd1\xce\x1b\x07v\xc0\xc4d\x
b4\xcciJ\x19\xb5\x97N@\x84\x04\xafM{\xd1\x166>\x99\xdd\xc6\xb4\x81s \x02#PHt\xaa
\xc1\xbe\x1d\\\xac@\x03\t?\xbb\xe6\x88~:\x0fC\t+\xc4~]\xc0\x05`\xbb\r\x9d\xf4\xe
1,>\x05\x89u.R7Ye#\x85\x1dQ\xaa\xb6\x8dlV\xad/F.d\x05,<g:\xbb\xa0\xaf\x07\xe9\xc
7\xf1\x97~R\xc2J5OL\x80\x12\x85\x04\xd9\xb3\x8c)Y\xf5` ! \x9c\xfa)\xeaH\xba\xf5\x
```

Fig. 12. Probando el código con un texto de formato ".csv" llamado Places Near Harlem que es un dataset de Kaggle y el resultado fue muy rápido.

```
Mensaje Decifrado:      name    origin start_time       summary end_time       c
oords   address id      url
0       Community Food & Juice  gmaps  ['Monday: 9:00 AM – 9:00 PM', 'Tuesday: 9
:00 AM – 9:00 PM', 'Wednesday: 9:00 AM – 9:00 PM', 'Thursday: 9:00 AM – 9:00 PM'
, 'Friday: 9:00 AM – 9:00 PM', 'Saturday: 9:00 AM – 9:00 PM', 'Sunday: 9:00 AM –
 9:00 PM'] I live nearby and have been here several times, the food is good and
consistent as well as the service. The price is reasonable also. A decent go-to
place for me.   ['Monday: 9:00 AM – 9:00 PM', 'Tuesday: 9:00 AM – 9:00 PM', 'Wed
nesday: 9:00 AM – 9:00 PM', 'Thursday: 9:00 AM – 9:00 PM', 'Friday: 9:00 AM – 9:
00 PM', 'Saturday: 9:00 AM – 9:00 PM', 'Sunday: 9:00 AM – 9:00 PM']     (-73.965
7833, 40.8059222)       2893 Broadway, New York, NY 10025, United States       C
hIJnSR09Dv2wokRz_B9gF6v96c      https://www.google.com/maps/place/2893 Broadway,
 New York, NY 10025, United States
1       Junzi Kitchen   gmaps   ['Monday: 11:30 AM – 9:00 PM', 'Tuesday: 11:30 A
M – 9:00 PM', 'Wednesday: 11:30 AM – 9:00 PM', 'Thursday: 11:30 AM – 9:00 PM', '
Friday: 11:30 AM – 9:00 PM', 'Saturday: 11:30 AM – 9:00 PM', 'Sunday: 11:30 AM –
 9:00 PM'] "June 29, 2021
Junzi Kitchen was delish... however ... there was a few downsides. First and for
emost there were no drinks to choose from that could be drank in full cups.  The
y had canned drink and juice boxes. In order to drink the juice box I had to pok
e a hole at the top and drizzle it over a cup of ice in a sample cup. Wow...Ther
e were no chips that could have been interesting to indulge in that were differe
nt interpretations of Lays.  Also I asked about a picture of something on the we
```

Fig. 13. Se muestra una parte del texto del dataset descifrado.

La tercera prueba es con un dataset de Kaggle que es sobre el covid-19 en 214 países, tiene un peso de 10 MB y fue poco rápido al procesar.



```
06-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
07-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
08-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
09-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
10-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
11-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
12-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
13-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
14-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
15-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
16-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
17-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
18-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
19-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
20-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
21-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
22-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
23-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
24-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
25-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
26-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
27-10-20,,International,696,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,

(big_data) [root@192 seguridad]#
```

Fig. 14. Parte final del proceso, que muestra una parte del texto desencriptado.

La cuarta es un dataset de peso de 35 MB en el cual se tardó algunos segundos más en procesar, pero no pasó de minutos y es sobre reseñas de Universal Studios de Kaggle.

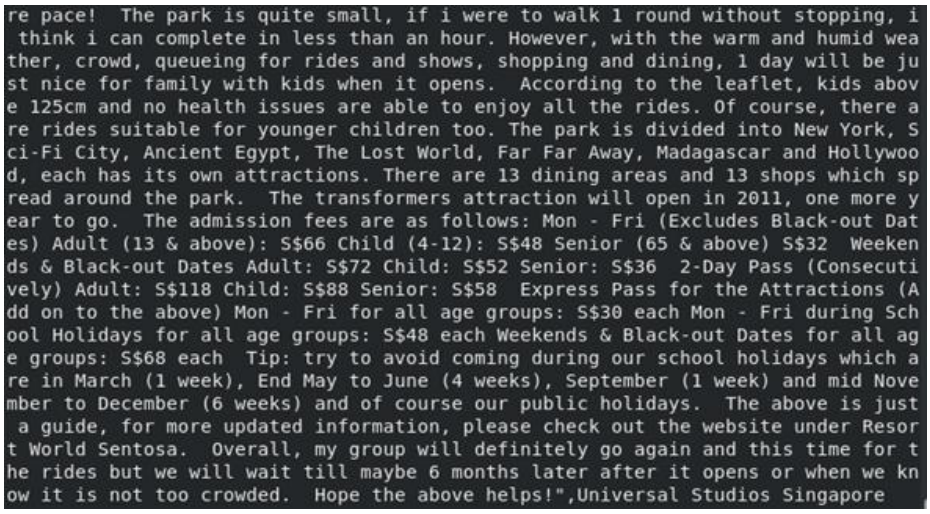


Fig. 15. Parte final del proceso, que muestra una parte del texto descifrado con un dataset que es un poco grande.

5 Conclusiones

El algoritmo más útil fue el de Python ya que soporto más texto en su algoritmo al momento de encriptar y desencriptar, apenas con 36 MB de un dataset se puso lento por unos segundos, lo cual quiere decir que soportará más peso antes de que se atrase por minutos, el de Java solo soporto 2 MB para alentarse por minutos, lo cual una vez hecho estas pruebas puedo concluir que para este algoritmo ocuparía poner máximo 1 MB para que el retardo sea de segundo, por otro lado en Python podrían ser hasta 40 MB.
En lo personal, recomendaría más el algoritmo asimétrico de Python para la seguridad de las bases de datos o correos, ya que brinda un mejor rendimiento y mayor seguridad, aparte que es mucho más fácil su programación, ya que es más didáctico en la cuestión que aparte de ser orientado a objetos, es un lenguaje de alto nivel y no tienes que preocuparte por declarar variables de tipos, si no que se adaptan a lo que necesites; En Java tuve algunas complicaciones más debido que es más reglamentario y están más dispersas las librerías, tuve un problema al querer guardar la llave en este lenguaje, pero lo solucione buscando otro método donde guardas el objeto.

Referencias

1. Blokhin, Ilia.: Mecanismos de seguridad para Big Data basados en circuitos criptográficos. Universidad de Granada. Tesis Doctorales. (2020) 37–54.
2. SAS. (s.f). Big Data. (s.f) Sitio web: https://www.sas.com/es_mx/insights/big-data/what-is-big-data.html
3. ViewNext. (s.f). La influencia del big data en la seguridad. Sitio Web: https://www.viewnext.com/bigdata-ciberseguridad/
4. Bech. (s.f). La importancia del Big Data en la Ciberseguridad. Sitio Web: https://bes-h.com/es/big-data-ciberseguridad/
5. Gustin, Dwi. (2021). Reviews of Universal Studios. Kaggle Sitio web: https://www.kaggle.com/dwiknrd/reviewuniversalstudio
6. Mad, Ammaraah. (2021). covid-19 cases in 214 countries. Kaggle Sitio web: https://www.kaggle.com/ammaraahmad/covid19-cases-in-214-countries
7. Suru, Its. (2021). Loan Data. julio 30, 2021, de kaggle Sitio web: https://www.kaggle.com/itssuru/loan-data
8. Kirubi, Rush. (2016). Video Game Sales with Ratings. Kaggle Sitio web: https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings
9. Kaggle, J. (2021). Places Near Harlem (Manhattan). Kaggle Sitio web: https://www.kaggle.com/jaycram/places-near-morningside-heights.

Optimización de Despacho Económico con Algoritmo de Luciérnagas

Julián Antoni Díaz Ayón y Maya Carrillo Ruiz

Benemérita Universidad Autónoma de Puebla,
4 Sur 104 Centro Histórico C.P. 72000 Puebla, Pue., México
julian.diazayon@viep.com.mx, maya.carrilloruiz@viep.com.mx

Resumen. Los generadores termoeléctricos queman combustibles fósiles como carbón, gas, o petróleo, convirtiendo el calor en energía eléctrica. Estos combustibles son recursos no renovables y contaminantes. Debido a esto, optimizar la planeación y operación de los generadores termoeléctricos es un problema muy importante en la industria eléctrica. El concepto de optimización se refiere a encontrar los valores que maximizan o minimizan una función matemática. Al problema de minimizar el costo operativo de un sistema de generadores termoeléctricos mientras se satisface la demanda de energía, se le conoce como despacho económico. El despacho económico es un problema de optimización con restricciones, en donde se busca minimizar el costo total de los combustibles consumidos por los generadores. En el presente trabajo se propone la solución del problema de despacho económico empleando el algoritmo de luciérnagas. Los resultados obtenidos por este algoritmo son equiparables a los reportados en trabajos que emplean otras metaheurísticas.

Palabras Clave: Metaheurísticas, Algoritmo de Luciérnagas, Optimización, Investigación de Operaciones, Despacho Económico.

1 Introducción

Los generadores termoeléctricos queman combustibles fósiles como carbón, gas, o petróleo, convirtiendo el calor en energía eléctrica. Estos combustibles son recursos no renovables y contaminantes. Debido a esto, la planeación y operación eficiente de los generadores termoeléctricos es un problema muy importante en la industria eléctrica.

La cantidad de combustible que se necesita quemar anualmente para satisfacer las necesidades energéticas de un país son tan grandes, que cualquier ahorro que se pueda ganar representa millones de dólares.

Al problema de minimizar el costo operativo de un sistema de generadores termoeléctricos mientras se satisfacen la demanda de energía y las restricciones del sistema, se le conoce como despacho económico (ED, por sus siglas en inglés). El problema de ED se ha resuelto con distintas técnicas de optimización tales como:

multiplicadores de Lagrange, programación dinámica, programación lineal, programación cuadrática y programación no lineal. La formulación original del problema de ED, en la cual se debe optimizar una función cuadrática bajo restricciones lineales, puede ser resuelta usando multiplicadores de Lagrange o programación cuadrática. Sin embargo, esta formulación es limitada ya que no se consideran muchos aspectos importantes como: las pérdidas en la transmisión, opción de múltiples combustibles, zonas prohibidas de operación, sistemas de almacenamiento, otras fuentes de energía, entre otros. Por esta razón, es importante agregar restricciones adicionales para garantizar la seguridad del sistema, previniendo su colapso bajo condiciones imprevistas [1]. Debido a que considerar estas restricciones hace que el problema sea mucho más complicado, a las técnicas clásicas de optimización se les dificulta resolver el ED, por lo que una buena alternativa de solución son las metaheurísticas, las cuales son técnicas generales de optimización global capaces de resolver satisfactoriamente problemas altamente no lineales y con muchos óptimos locales. En el presente trabajo se propone solucionar el ED de un sistema de generadores empleando una metaheurística llamada algoritmo de luciérnagas (FFA), la cual es eficiente y fácil de implementar. Nuestra aportación consiste en investigar la capacidad de FFA para resolver el ED con restricciones operativas. Hasta donde sabemos, FFA ha sido poco investigado para esta aplicación en particular.

Este trabajo está dividido como sigue: la segunda sección introduce el concepto de optimización y luego define el problema de optimización con restricciones y el problema de ED. La tercera sección menciona trabajos relacionados con el sistema que se resuelve aquí. La cuarta sección explica el algoritmo de luciérnagas el cual es el método que usamos para resolver el problema. La quinta sección presenta los datos del sistema de prueba y los datos experimentales del algoritmo. Finalmente, la sexta sección concluye este trabajo y menciona como se puede extender.

2 Formulación del Problema

El concepto de optimización está fuertemente relacionado con el análisis de muchos problemas de decisión. Al enfrentar un problema de decisión el cual consiste en encontrar los valores de un cierto número de variables, usualmente nos fijamos en una sola función objetivo la cual es una función matemática diseñada para cuantificar el desempeño y la calidad de la decisión. Esta función objetivo puede ser minimizada o maximizada, dependiendo de la formulación del problema, y está sujeta a las restricciones que limitan la selección de los valores de las variables de decisión [2].

2.1 Optimización con Restricciones

El problema general de optimización con restricciones puede ser expresado como

$$\text{Min } f(x)$$
$$\text{Sujeto a } h_i(x) = 0,\ i = 1,\ 2, \dots,\ m$$
$$g_j(x) \leq 0,\ j = 1,\ 2, \dots,\ p$$
$$x\ \in S$$

En esta formulación, $x = (x_1,\ x_{2,}\ \dots,\ x_n) \in \mathbb{R}^n$; f , h_i y g_i son funciones: $f: \mathbb{R}^n \to \mathbb{R}$, $h_i: \mathbb{R}^n \to \mathbb{R}$, $i = 1, \dots, m$; $g_j: \mathbb{R}^n \to \mathbb{R}$, $j = 1, \dots, p$; y $S \subseteq \mathbb{R}$.

Por simplicidad en la notación, definamos $h = (h_1,\ h_2, \dots,\ h_m)$ y $g = (g_1,\ g_2, \dots,\ g_m)$, entonces el problema de optimización con restricciones se puede escribir como

$$\text{Min } f(x)$$
$$\text{Sujeto a } h(x)\ =\ 0$$
$$g(x) \leq 0$$
$$x \in S$$

2.2 Despacho Económico

El problema de ED es un problema de optimización con restricciones, en donde se busca minimizar el costo total de los combustibles consumidos por los generadores termoeléctricos.

En el análisis de ED es necesario conocer la cantidad o el costo del combustible consumido en función de la potencia generada para cada generador, esta función se conoce como característica de la unidad termoeléctrica. La suma de las características de las unidades del sistema es la función objetivo del problema de optimización. El problema se expresa como

$$\text{Min } F_T = \sum_{i=1}^{n} F_i(P_i)$$
$$\text{Sujeto a } \sum_{i=1}^{n} P_i = P_D$$
$$P_{i,min} \leq P_i \leq P_{i,max},$$

Donde P_D es la potencia total que debe satisfacerse, n es el número de generadores, $F_i(P_i)$ es la característica de la unidad i , P_i su potencia de entrega, y $P_{i,min}$ y $P_{i,max}$ la potencia mínima y máxima con la cual puede trabajar. P_i tiene unidades de megavatios

(MW), y $F_i(P_i)$ tiene unidades de $\$/h$, es decir, el costo del combustible consumido por hora.

La forma clásica de una característica termoeléctrica es cuadrática, es decir

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i$$

Las unidades termoeléctricas grandes tienen varias válvulas que son abiertas en secuencia para satisfacer la demanda. Al considerar el efecto de estas válvulas el ED clásico se convierte en el problema de despacho económico con efecto de punto de carga de válvulas (EDVPL, por sus siglas en inglés). Si además consideramos las pérdidas en la transmisión del sistema, el problema es expresado como

$$\begin{aligned} \text{Min } F_T = \sum_{i=1}^{n} & \left\{ a_i P_i^2 + b_i P_i + c_i + \left| d_i \sin\left[e_i \left(P_{i,min} - P_i \right) \right] \right| \right\} \\ \text{Sujeto a } \sum_{i=1}^{n} & P_i = P_P + P_D \\ & P_{i,min} \leq P_i \leq P_{i,max}, \end{aligned}$$

Donde P_P representa la pérdida del sistema, y se calcula como

$$P_P = \sum_{i=1}^{n} \sum_{j=1}^{n} P_i \, B_{ij} P_j + \sum_{i=1}^{n} B_{i0} \, P_i + B_{00}$$

donde B_{ij}, B_{i0} y B_{00} son los coeficientes de pérdida. Específicamente, B_{ij} son los coeficientes de la matriz de pérdida B , B_{i0} los coeficientes del vector de pérdida B_0, y B_{00} el coeficiente constante de pérdida [3].

3 Trabajo Relacionado

En la sección de resultados experimentales se resuelve el problema de EDVPL de un sistema de tres generadores ignorando las pérdidas. Este sistema de tres generadores ha sido resuelto con distintos métodos. Estos son: Walters D.C. y Sheble G.B. [7] resolvieron el EDVPL con un algoritmo genético (GA). Victoire T.A.A. y Jeyakumar A.E. [6] resolvieron este EDVPL con un algoritmo híbrido de enjambre de partículas y programación cuadrática secuencial (PSO-SQP), enjambre de partículas (PSO), programación evolutiva (EP) y programación evolutiva con programación cuadrática secuencial (EP-SQP). Labbi et al. [5] empleó la colonia artificial de abejas (ABC) para

resolver este EDVPL. Azmi A.M. et al [9] emplearon un algoritmo híbrido que combina el optimizador de lobo gris (GWO) con el algoritmo de escalamiento de colina β (β HC). Secui y Rancov. [10] usaron un algoritmo híbrido que combina seno coseno (SCA) y polinización de flores (FPA). Azmi A.M. et al [11] emplearon una hibridación que combina seno-coseno con escalamiento de colina β , usando β HC como un operador dentro de SCA. Az,o A.M. empleó el algoritmo de búsqueda armónica basada en islas (iHS) el cual divide la población en un conjunto de poblaciones y se aplica el algoritmo de HS original a cada isla.

Tabla 1 Se muestran la lista de los algoritmos metaheurísticos usado para resolver el ED. Se muestran el año de publicación, el algoritmo empleado y el número de veces que se ha citado el artículo correspondiente.

Año	Algoritmo	Número de citas
1993	Walters y Sheble emplearon algoritmos genéticos (GA)	747
2013	Victoire y Jeyakumar emplearon un algoritmo híbrido que combina optimización de enjambre de partículas con programación cuadrática secuencial (PSO-SQP)	379
2014	Li et al. emplearon una versión mejorada del algoritmo de evolución diferencial (DE)	0
2014	Labbi et al. emplearon el algoritmo de colonia artificial de abejas (ABC)	24
2017	Abbas et al. Usaron una versión modificada de optimización de enjambre de partículas	67
2020	Azmi A.M. et al. emplearon un híbrido que combina lobo gris (GWO) y escalamiento de colina β (β HC)	28
2021	Azmi A.M. empleó el algoritmo de búsqueda armónica basada en islas (iHS)	6
2022	Azmi A.M. et al. Usaron una combinación del algoritmo de seno-coseno y escalamiento de colina β (SCA-HC)	2
2022	Secui y Rancov usaron un híbrido que combina seno-coseno y polinización de flores (FPA-SCA)	0

4 Método de Solución

Las luciérnagas emiten luz en patrones rítmicos en un proceso de bioluminiscencia; los dos principales usos de estos patrones son: atraer posibles parejas o como advertencia para depredadores. Aun cuando existen más de 2,000 especies de luciérnagas, cada una tiene su propio patrón de iluminación. Algunas especies pueden incluso sincronizar sus patrones de tal manera que forman un sistema autoorganizado.

La intensidad de la luz I de una fuente a una distancia r es inversamente proporcional al cuadrado de r, esto es, $I \frac{\propto 1}{r^2}$. Adicionalmente, el aire absorbe parte de la luz, lo cual hace que se vuelva más débil a distancias más grandes. Estos dos factores hacen que una luciérnaga tenga una distancia límite a la cual puede ser vista [4].

Lo anterior inspiró a Xin-She Yang en 2007 para crear el algoritmo de luciérnagas (FFA). Una solución a un problema de maximización es representada por una luciérnaga y el brillo representa el valor de la fuente objetivo. Cada luciérnaga verá cada una de las demás luciérnagas y percibirá sus brillos, acercándose a estas si son más brillantes que ella misma. El paso que da la luciérnaga para acercarse a las más brillantes es proporcional a la intensidad de la luz que percibe.

Algoritmo 1: Algoritmo de Luciérnagas

Entrada: función objetivo: $f(x)$, tamaño de población: $psize$, número de iteraciones: $niter$, coeficiente de absorción: γ , coeficiente de atracción: β_0, tamaño de paso aleatorio: α_0, constante de reducción de paso aleatorio: θ .
Salida: mejor solución: x_{best}.

```
1. Generar la población inicial de luciérnagas X = (x^(1), x^(2), ..., x^(psize))
2. Determinar la mejor luciérnaga x^(best)
3. for t = 1:niter
4.           for  i = 1:psize
5.                  for j = 1:psize
6.                        if f(x^(1)) < f(x^(2))
7.                              x^(i) = x^(i) + β0 e − γ r_ij^2 (x^(j) − x^(i)) + α0 θ^(−t) ε
8.                              if f(x^(i)) > f(x^(best))
9.                                    x^(best) = x^(i)
10.                       end if
11.                 end if
12.          end for
13.    end for
14. end for
15.
16. return x^(best)
```

Donde $r_{ij} = \sqrt{\left|\left|x^{(j)} - x^{(i)}\right|\right|_2}$ y ε es un vector aleatorio con distribución uniforme. Existen otras variantes de FFA que se pueden consultar en [4].

Hasta el momento no tenemos noticia de que el sistema de prueba que se describe en la quinta sección haya sido resuelto con FFA. Por esta razón, proponemos este método de solución.

4.1 Restricciones

Las restricciones se trataron con el método de penalización [4].
De la formulación del problema

$$\begin{gathered}\text{Min } f(x) = \sum_{i=1}^{n} F_i(P_i) \\ \text{Sujeto a } P_D - \sum_{i=1}^{n} P_i = 0 \\ P_{i,\min} \leq P_i \leq P_{i,\max}\end{gathered}$$

Donde $x = (P_1, \ldots, P_n)$.

La restricción $P_{i,\min} \leq P_i \leq P_{i,\max}$ se impone al crear la población inicial y al corregir una solución nueva si esta se sale de los límites.
Para la restricción $P_D - \sum_{i=1}^{n} P_i = 0$, primero se debe notar que la demanda total debe ser satisfecha obligatoriamente por lo que podemos reescribir esta restricción como $\psi(x) = P_D - \sum_{i=1}^{n} P_i \leq 0$, y definimos la función objetivo penalizada como

$$pf(x) = f(x) + \nu H[\psi(x)]\psi(x)^2,$$

donde ν es un valor muy grande en términos de la escala de problema (en nuestro caso se utilizó 10^9) y $H[\cdot]$ es una función índice: $H[\psi(x)] = 0$, si $\psi(x) \leq 0$, $H[\psi(x)] = 1$, si $\psi(x) > 0$.

Luego, para encontrar el minimizador de $pf(x)$ se maximiza $-pf(x)$ debido a que esta variante de FFA resuelve problemas de maximización.

5 Lenguaje de Programación: Julia

Julia es un lenguaje de programación multi paradigma de propósito general, aunque fue originalmente diseñado para el cómputo numérico/científico. Julia usa un compilador *justo antes de tiempo* (JIT), compilando todo el código a código máquina antes de

ejecutarlo. Julia además, cuenta con muchas librerías eficientes para el cómputo numérico. Sus principales características son:

1. El despacho múltiple: nos permite definir el comportamiento de las funciones a través de diversas combinaciones de tipos de argumentos.
2. Sistema de tipado dinámico: tipos para la documentación, la optimización y el despacho de funciones.
3. Desempeño comparable al de lenguajes estáticamente tipados como C.
4. Gestor de paquetes integrado.
5. Macros tipo Lisp y otras herramientas para la meta-programación.
6. Capacidad de llamar otros lenguajes (Python, R, Java/Scala).
7. Llamar funciones de C directamente: sin necesidad de usar envoltorios o APIs especiales.
8. Poderosas características de línea de comandos para gestionar otros procesos.
9. Diseñado para la computación paralela y distribuida.
10. Los tipos definidos por el usuario son tan rápidos y compactos como los tipos estándar integrados.
11. Generación automática de código eficiente y especializado para diferentes tipos de argumentos.
12. Elegantes y extensibles conversiones y promociones para tipos numéricos y de otros tipos.
13. Soporte eficiente de Unicode no limitado a UTF-8.
14. Licencia MIT: libre y de código abierto.

6 Resultados Experimentales

Para verificar la competitividad de FFA, se resolvió un problema de EDVPL de tres generadores ignorando las pérdidas. FFA fue programado en Julia 1.6.4 y ejecutado en un AMD Ryzen 5 5600G 3.9GHz con 16Gb de memoria en Ubuntu 20.04.

Los datos del sistema de tres generadores se encuentran en [5] y en Tabla 2. Este sistema tiene una demanda total de 850MW ya que así se indica en [5].

Tabla 2. Se muestran los valores de los coeficientes de las características de los generadores del sistema de prueba [5].

Unidad	$P_{i,min}$	$P_{i,max}$	a	b	c	d	e
1	100	600	0.001562	7.92	561	300	0.0315
2	50	200	0.004820	7.97	78	150	0.0630
3	100	400	0.001940	7.85	310	200	0.0420

El algoritmo fue ejecutado 50 veces con los siguientes parámetros:

$$\gamma = 6.89655172413793 \times 10^{-7}$$
$$\beta = 1.0$$
$$\alpha = 34.76108935769036$$
$$\theta = 0.99$$
$$niter = 1000$$

En la Tabla 3 se muestran: el costo mínimo, el costo máximo, la media muestral y la desviación estándar muestral de las 50 ejecuciones.

Tabla 3. Se muestran los datos muestrales de las siguientes ejecuciones del algoritmo de luciérnagas para el problema EDVPL del sistema de tres generadores de la tabla 2.

F_{min}	F_{max}	F_{mean}	F_{std}
8234.07188	8424.6852	8260.1579	46.1204

En la Tabla 4 se muestra la comparación del mejor resultado obtenido por FFA comparado con cada algoritmo presentado en [5]. Se puede ver que FFA obtiene resultados equiparables a las demás metaheurísticas empleadas previamente. Excepto por la solución encontrada por GA. El costo total y la potencia de salida del generador

tres son iguales para todas las soluciones encontradas, mientras que las potencias de salida de los generadores uno y dos varían un poco. La solución encontrada por GA tampoco satisface la demanda por lo que esta es una solución no factible. Todas las demás soluciones encontradas son factibles, ya que satisfacen la demanda de 850MW.

Tabla 4. Se muestra la comparación de los mejores resultados obtenidos por FFA, ABC, EP, EP-SQP, PSO, PSO-PSOSQP, GA y GWO-HC. FFA encontró resultados equiparables a los demás algoritmos. GA es el único que no satisface la demanda de 850 MW.

Método	P_1	P_2	P_3	P_4	Costo
FFA	300.267	149.733	400.000	850.0	8234.07
ABC	300.266	149.734	400.000	850.0	8234.07
EP	300.264	149.736	400.000	850.0	8234.07
EP-SQP	300.267	149.733	400.000	850.0	8234.07
PSO	300.268	149.732	400.000	850.0	8234.07
PSO-SQP	300.267	149.733	400.000	850.0	8234.07
GWO-HC ($R_\beta = 0$)	300.262	149.738	400.000	850.0	8234.07
GWO-HC ($R_\beta = 0.05$)	410.374	97.748	341.878	850.0	8234.07
GWO-HC ($R_\beta = 0.1$)	395.403	147.377	307.219	850.0	8234.07
GWO-HC ($R_\beta = 0.5$)	393.533	152.586	303.881	850.0	8234.07
GA	398.700	50.100	399.6	848.4	8222.07

En la figura 1 se muestra la aptitud de la mejor solución $-pfit(x)$ encontrada por FFA contra las iteraciones. Se puede ver como FFA se mueve rápidamente desde soluciones subóptimas hasta la mejor solución encontrada.

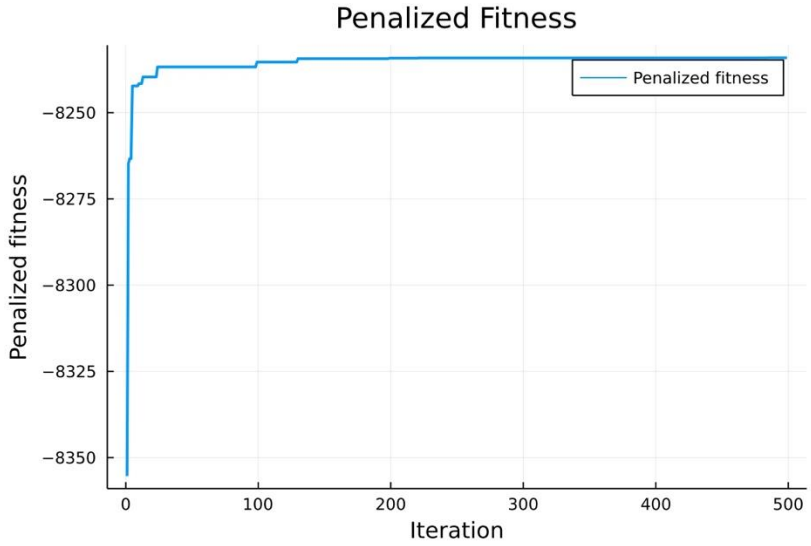


Fig. 1: Se muestra la aptitud de la mejor solución encontrada por FFA contra las iteraciones.

7 Conclusiones

En este trabajo utilizamos una versión básica de FFA para resolver el EDVPL ignorando las pérdidas de un sistema de tres generadores. FFA fue capaz de resolver el problema satisfactoriamente. Adicionalmente, los resultados obtenidos fueron equiparables a otras metaheurísticas, mostrando que es competente a pesar de su simplicidad. En trabajos futuros, sería deseable explorar modificaciones o hibridaciones para reducir el número de iteraciones necesarias para la convergencia, considerar la pérdida en la transmisión y otras restricciones operativas. Así como sistemas con más generadores.

Referencias

1. Wood, A.J. et al.: Power Generation, Operation and Control. 3rd edn. John Wiley & Sons, Hoboken, New Jersey (2014).
2. Luenberger, D.G., Ye, Y.: Linear and Nonlinear Programming. 5th edn. Springer, Switzerland (2021).
3. Abbas G. et al.: Solution of an Economic Dispatch Problem Through Particle Swarm Optimization: A Detailed Survey - Part I. IEEE Access Volume 5, (2017).
4. Yang, X.-S.: Nature-Inspired Optimization Algorithms. 1st ed. Elsevier,Jamestown Road, London (2014).

5. Labbi, Y. et al.: Artificial bee colony optimization for economic dispatch with valve point effect. Frontiers in Energy, 8(4): 449–458 (2014).
6. Victoire, T.A.A, Jeyakumar, A.E.: Hybrid PSO–SQP for economic dispatch with valve-point effect. Electric Power Systems Research 71 (2004) 51–59 (2013).
7. Walters, D.C., Sheble, G.B.: Genetic Algorithm Solution of Economic Dispatch With Valve Point Loading. IEEE Transactions on Power Systems, Vol. 8, No. 3 (1993).
8. Li, Y. et al.: An Improved Differential Evolution Algorithm for Economic Dispatch with Value Point Effect. In: Xing, S. et al. (eds.) Unifying Electrical Engineering and Electronics Engineering, Lecture Notes in Electrical Engineering 238. Springer Science+Business Media New York (2014).
9. Azmi, A.M. et al: A non-convex economic load dispatch problem with valve loading effect using a hybrid grey wolf optimizer. Neural Computing and Applications 32:12127–12154 (2020).
10. Secui, D.C., Rancov, N.: Hybrid Sine–Cosine Algorithm with Flower Pollination Algorithm for Economic Dispatch Problem with Valve-Point Effects and Wind Power Integration. Arabian Journal for Science and Engineering 47:3421–3445 (2022).
11. Azmi, A.M. et al: Economic load dispatch using memetic sine cosine algorithm. *J Ambient Intell Human Comput* (2022).
12. Azmi A.M.: Island-Based Harmony Search Algorithm for Non-convex Economic Load Dispatch Problems. *J. Electr. Eng. Technol.* **16**, 1985–2015 (2021).

Laboratorio de Cómputo Forense en Alma Linux

Yeiny Romero Hernández, Maria del Carmen Santiago Díaz,
Gustavo Trinidad Rubín Linares, Ricardo Martínez Pérez

Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla,
Av. San Claudio y 14 sur, C.P. 72000.Puebla, Pue., México
{yeiny.romero, maricarmen.santiago, gustavo.rubin}@correo.buap.mx,
ricardo.martinezp@alumno.buap.mx

Resumen El cómputo forense analiza y previene delitos informáticos, por lo cual es de suma importancia para la ciberseguridad, dado que nos ayuda a prevenir algunos ataques en la web, proteger información e incluso recuperar datos, con el estudio de esta rama de la ciberseguridad se busca dar a conocer herramientas que serán de utilidad en un laboratorio forense con la finalidad de practicar en diversos sistemas operativos para prevenir algunos ataques.

1 Introducción

Hoy en día existen miles de usuarios conectados al internet lo que lleva a la movilidad de grandes cantidades de datos y, por ende, no hay día que no se reporte que millones de páginas de internet, servidores y equipos móviles que son blancos de la ciberdelincuencia. En los últimos años estos ataques han incrementado dado que cada vez hay más usuarios conectados y no todos cuentan con la cultura de la protección. La ciberseguridad se encarga de la protección de la información y dentro de ella existen diferentes ramas; una de ellas es el cómputo forense que cuenta con muchas herramientas para análisis y prevención de ataques, así como recuperación de la información. Dentro de las herramientas importantes se encuentran NetworkMiner, Snort, FTK Imager, etc.

1.1 NetworkMiner

NetworkMiner es una Herramienta de Análisis Forense de Redes (NFAT) que trabaja en diversos sistemas operativos. Puede ser usado como una herramienta pasiva de captura de paquetes/sniffer para detectar sistemas operativos, sesiones, hostnames, puertos abiertos, etc. sin añadir más tráfico a la red. También puede analizar archivos PCAP para análisis sin conexión y para regenerar archi- vos transmitidos y certificados desde archivos PCAP [1].

NetworkMiner cuenta con una versión gratuita, algunas de sus características importantes son: Sniffing activo, Análisis de archivos PCAP, PcapNG y ETL, Soporte para IPv6, Extracción de archivos desde FTP, TFTP, HTTP, HTTP/2, SMB, SMB2, SMTP, POP3, IMAP y tráfico LPR, entre otras.

1.2 Snort

Es un sistema de prevención de intrusiones de código abierto, cuenta con análisis de tráfico en tiempo real y registro de paquetes. Utiliza una serie de reglas que ayudan a definir actividad de red maliciosa y utiliza estas reglas para encontrar paquetes que coincidan con dicha actividad y genera alertas para los usuarios.

Snort tiene tres usos primarios, como sniffer de paquetes, como registro de paquetes y finalmente, un sistema de prevención de intrusiones de red. Puede ser descargado y configurado para uso personal o para empresas.

Las reglas de Snort son distribuidas en dos sets, el primero se conoce como "Snort Subscriber Ruleset" o "Set de Reglas de Suscriptores Snort", estas son desarrolladas, testeadas y aprobadas por Cisco Talos, y estas reglas son recibidas en tiempo real tan pronto sean liberadas para los clientes Cisco. El segundo set es el "Community Rule- set" o "Set de Reglas de la Comunidad", las cuales son desarrolladas por la Comunidad Snort y QAed de Cisco Talos, estas son libres para todos los usuarios. [6]

1.3 FTK Imager

FTK Imager es una herramienta de análisis forense para la creación de copias perfectas, o imágenes forenses de datos de cómputo sin realizar cambios a la evidencia original. Esto permite almacenar los medios originales con seguridad, libres de daño mientras la investigación procede utilizando la imagen. También tiene la utilidad de generar reportes hash para usar como referencia para probar la integridad de la evidencia. Cuando se crea la imagen completa de un disco, un hash generado por FTK Imager puede ser usado para verificar que la imagen hash y el hash del disco coincidan después de que la imagen es creada y que la imagen ha permanecido intacta desde su adquisición. [15]

Una vez que conocemos las herramientas, el objetivo de este proyecto será ver la compatibilidad con el sistema operativo Alma Linux para ponerlas en práctica en cualquier laboratorio destinado al cómputo forense y comprobar que se puede practicar en dicho S.O.

2 Metodología

2.1 Instalación NetworkMiner

Para comenzar es necesario instalar la plataforma Mono, la cual ofrece soporte para CentOS/RHEL versión 8 [3]. Mono ofrece 2 tipos de instalación, mono-devel, que incluye la funcionalidad de compilar código, y mono-complete, que incluye todos los módulos, para evitar posibles errores se eligió el paquete mono-complete. El proceso es sencillo y no presentó problema alguno, la versión utilizada de Mono es la 6.8.0.123.

Para la instalación de NetworkMiner se utilizan comandos desde consola para descargar, descomprimir y modificar los permisos de las carpetas. Finalmente, desde la

carpeta que contiene NetworkMiner.exe, se puede ejecutar utilizando el comando *mono* . véase Fig. 1

Fig. 1. NetworkMiner en AlmaLinux utilizando comando Mono

Funcionamiento

Ejecutar NetworkMiner con Mono no arroja ninguna advertencia o error y la aplicación inicia sin ningún problema. (Véase Fig. 2) La interfaz de la aplicación contiene todos los elementos de la versión de Windows a excepción de la funcionalidad de elegir un adaptador de red para iniciar la captura (sniffing) de paquetes, sin embargo, esta funcionalidad tampoco está disponible en otras distros Linux. Cabe mencionar que Netresec recomienda utilizar herramientas como Wireshark o tcpdump para la captura de paquetes, incluso para usuarios de Windows.

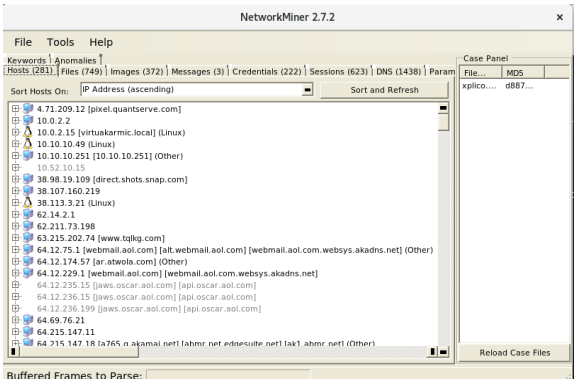


Fig. 2. NetworkMiner en AlmaLinux utilizando Mono

2.2 Instalación Snort

Antes de instalar Snort se deben instalar varios paquetes necesarios para su funcionamiento, no existe problema en instalarlos, sin embargo, algunas otras librerías fueron necesarias para Snort o para el plugin OpenAppID, estas son openssl-devel, libdnet, luajit, daq-devel, libtirpc-devel.

Para la instalación de Snort se utilizó el archivo RPM de CentOS que es compatible con Alma Linux, el instalador de Snort crea todos las carpetas y archivos necesarios, además de crear configuraciones de Snort para su funcionamiento, de haber instalado Snort desde el archivo fuente, estas configuraciones deberían hacerse manualmente. Finalmente, no se encontró algún problema de configuración.

Funcionamiento

Para comprobar el funcionamiento de Snort, se utilizarán los 3 modos básicos de uso, el sniffer de paquetes, el logger de paquetes, y el sistema de detección de intrusiones. También se prueba el complemento PulledPork, que será usado para desacargar el Community Ruleset.

El primer modo de Snort probado es el sniffer de paquetes, el cual solo se encarga de escuchar un puerto de red para recibir los paquetes y mostrarlos continuamente en la pantalla de la consola. (Véase Fig.3) Este modo es el más simple y funcionó sin ningún problema.

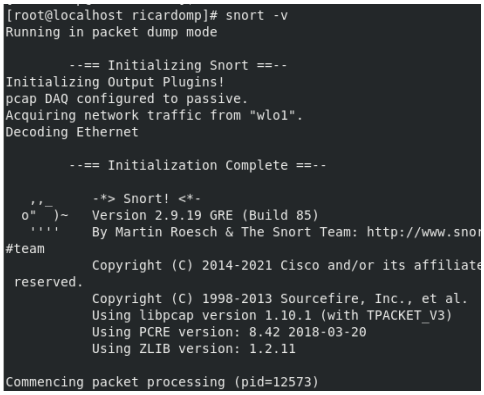


Fig. 3. Inicio de Snort en modo sniffer

2.3 Instalación FTK Imager

Esta versión de FTK viene en forma de un binario precompilado, el cual solo necesita ser descargado y ejecutado. Adicionalmente, se puede crear un enlace simbólico para abrir FTK Imager desde cualquier ubicación.

Funcionamiento

Utilizando la línea de comandos, se pueden utilizar las distintas opciones de FTK Imager. Una de ellas es –list-drives, el cual despliega la lista de los discos conectados en la PC, esta opción es importante para saber cómo referirnos al disco que queremos copiar. Véase Fig. 4

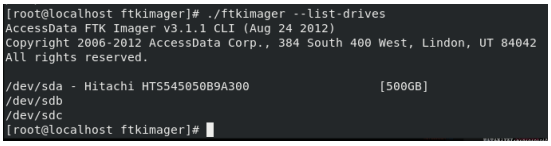


Fig. 4. Lista de discos

3 Resultados

3.1 NetworkMiner

Se utiliza un archivo PCAP de prueba para corroborar la funcionalidad de la aplicación, en la Fig. 5 se observa la pestaña de imágenes, la cual sirve para visualizar imágenes contenidas en los paquetes, así como información sobre estas.

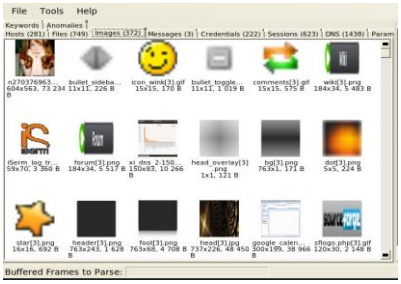


Fig. 5. Análisis de imágenes en el archivo PCAP

En la Fig. 6 se muestra la interfaz de un archivo PCAP y lo que puede mostrar.

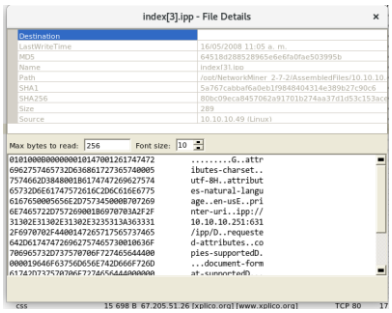


Fig. 6. Información de un archivo del PCAP

Finalmente, todas las herramientas de análisis y el despliegue de información de los archivos PCAP funciona sin ningún problema.

3.2 Snort

Recordemos que Snort tiene modos básicos de uso, el sniffer de paquetes, el logger de paquetes, y el sistema de detección de intrusiones. Mostraremos el modo Sniffer ver Fig.7

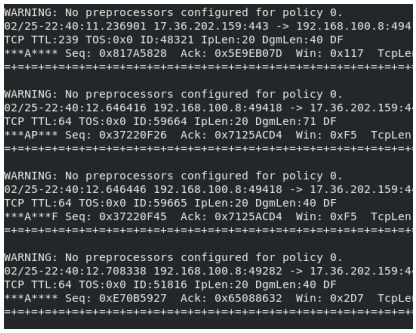


Fig. 7. Paquetes capturados

El segundo modo para probar es el logger de paquetes, el cual captura paquetes constantemente pero ahora guarda la información dentro de un archivo de log, en esta configuración es necesario crear una carpeta llamada log/ en donde Snort guardará la información en el disco. Ver Fig. 8

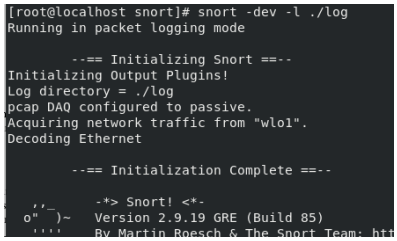


Fig. 8. Inicio de Snort en modo logging

El último modo para probar es el IDS o sistema de detección de intrusiones, el cual es el modo más extenso y complejo de Snort, para las pruebas vamos a probar la funcio- nalidad básica. Pero antes de esto, se realizó la prueba de PulledPork, ya que para probar el modo IDS es recomendable probar las reglas predefinidas, así como las reglas defi- nidas por el usuario. Ver Fig. 9 y Fig. 10

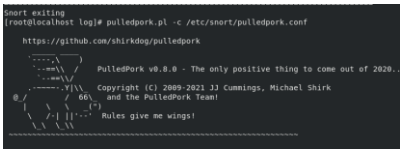


Fig. 9. Inicio de PulledPork

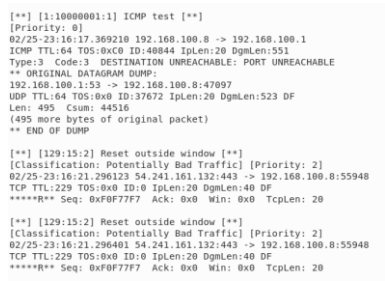


Fig. 10. Alertas generadas por Snort

3.3 FTK Imager

Para comenzar la creación de una imagen solamente se especifica el nombre del disco como aparece en –list-drives, y seguido la ubicación para guardar la imagen, opcionalmente se incluyen opciones de compresión, fragmentación y de metadata utilizadas para enumerar y clasificar la evidencia. Véase Fig. 11

Fig. 11. Creación de imagen de disco

Después de la creación del disco, los archivos generados pueden ser encontrados en la carpeta especificada, junto con estos también se genera un reporte hash el cual contiene información sobre los metadatos, información del disco, errores durante la copia y avisos y fecha de creación de la imagen. Véase Fig. 12

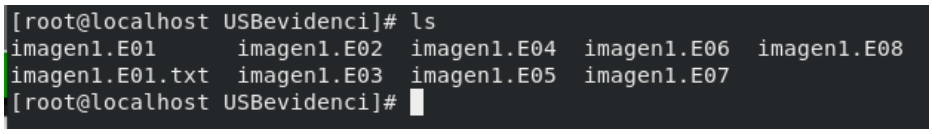


Fig. 12. Archivos generados

4. Conclusiones

NetworkMiner es una herramienta poderosa para el análisis de archivos PCAP, su funcionamiento no es tan complejo por lo que puede ser utilizada sin problemas usando la plataforma Mono, es una herramienta bastante útil para el cómputo forense, y complementándose de aplicaciones como Wireshark, se tiene un conjunto de herramientas poderosas para el análisis de redes.

Snort ha sido un proyecto muy importante para la seguridad en redes, ya que no solo funciona como un capturador de paquetes, sino también como un detector de intrusiones en tiempo real, con una gran cantidad de características y capacidad de personalización para cada sistema, se encuentra continuamente desarrollado para distintas plataformas. FTK Imager en AlmaLinux no presentó ningún problema, a pesar de ser una versión lanzada hace 10 años, el software puede realizar las imágenes de disco y la creación de reportes correctamente, por lo que, si se necesita de alguna herramienta gratuita de creación de imágenes, FTK Imager ofrece una solución sencilla pero eficaz.

Una vez probadas y analizadas estas herramientas en Alma Linux se puede decir que si se instalan en un laboratorio de cómputo forense serán complementarias para realizar practicas completas que nos lleven a mantener la seguridad de la red.

Referencias

1. NetworkMiner. (s/f). Netresec. Recuperado el 5 de febrero de 2022, de https://www.netresec.com/?page=NetworkMiner
2. HowTo install NetworkMiner in Ubuntu Fedora and Arch Linux. (2014, febrero 1). Netresec. https://www.netresec.com/?page=Blog&month=2014-02&post=HowTo-install-NetworkMiner-in-Ubuntu-Fedora-and-Arch-Linux
3. Download. (s/f). Mono-Project.Com. Recuperado el 5 de febrero de 2022, de https://www.mono-project.com/download/stable/
4. Chapter 11: Network analysis – digital forensics with Kali Linux – second edition – Dev tutorials. (s/f). Goois.Net. Recuperado el 5 de febrero de 2022, de https://goois.net/chapter-11-network-analysis-digital-forensics-with-kali-linux-second-
5. edition.html?fbclid=IwAR3k6UkT2_NtD17cFe8-0bqThZWApMZxZxk5eNQ7hzBYGG-FaFPePoOjpdpw
6. PacketCache. (s/f). Netresec. Recuperado el 5 de febrero de 2022, de https://www.netresec.com/?page=PacketCache
7. Snort - network intrusion detection & prevention system. (s/f). Snort.Org. Recuperado el 10 de febrero de 2022, de https://www.snort.org/
8. Snort supported OSes. (s/f). Snort.Org. Recuperado el 10 de febrero de 2022, de https://www.snort.org/documents/snort-supported-oses
9. Rezaei, M. (2020). Snort 2.9.16.1 on Centos 8. https://www.snort.org/documents
10. Snort rules and IDS software download. (s/f). Snort.Org. Recuperado el 10 de febrero de 2022, de https://www.snort.org/downloads
11. Shirk, M. (s/f). pulledpork: Pulled Pork for Snort and Suricata rule management (from Google code). Recuperado el 11 de febrero de 2022, de https://github.com/shirkdog/pu- lledpork
12. Snort Team. (2020). SNORT Users Manual 2.9.16. https://www.snort.org/documents
13. Volatility Foundation. (s/f). volatility: An advanced memory forensics framework. Recuperado el 8 de marzo de 2022, de https://github.com/volatilityfoundation/volatility
14. Volatility 2.6 release. (s/f). The Volatility Foundation. Recuperado el 8 de marzo de 2022, de https://www.volatilityfoundation.org/26
15. volatilityfoundation. (s/f). Installation. Volatility Wiki. Recuperado el 8 de marzo de 2022, de https://github.com/volatilityfoundation/volatility
16. FTK Imager. (2021, febrero 15). Exterro. https://www.exterro.com/ftk-imager
17. Fedora and red hat version x64 – 3.1.1. (s/f). AccessData. Recuperado el 4 de abril de 2022, de https://accessdata.com/product-download/fedora-and-red-hat-version-x64-3-1-1

Análisis de las Capacidades de un IDS: Suricata y Tripwire

Alexis Martinez, Ana C. Zenteno, Gustavo T. Rubín, Leslie Gómez

Facultad de Ciencias de la Computación, Benemérita Universidad
Autónoma de Puebla, C. 4 Sur 104, Centro Histórico de Puebla, 7200, Puebla, Pue, México.
{ana.zenteno, gustavo.rubin}@correo.buap.mx, {leslie.gomezm,
alexis.martinezga}@alumno.buap.mx

Resumen. Un Sistemas de Detección de Intrusos (IDS) es un software que, en ciberseguridad, permiten observar lo que sucede en la red en tiempo real. Recopilan información para reconocer modificaciones en documentos y en los paquetes que son enviados a través de una red. Son sistemas de seguridad, cuya función principal es alertar de posibles ataques a la infraestructura de red. Esto, sin bloquear o minimizar las consecuencias del ataque a la información o sistema objetivo. En el mercado existe una gran variedad de IDS, y en este trabajo se analizan las ventajas de IDS que detectan cambios a sistemas de archivos y sobre las conexiones en la red. Se analiza la relevancia de estos, además se comparan las funcionalidades, definiendo sus casos de uso proporcionado situaciones de usabilidad.
Palabras Clave: Ciberseguridad, IDS, Alertas.

1 Introducción

Los humanos y las computadoras han compartido el escenario durante siglos. En sus inicios allá en los 1800 esta relación era mayormente impráctica debido a su alto costo y poca accesibilidad. Las primeras computadoras eran gigantescas, tanto como el tamaño de una habitación entera, y eran miles de veces menos potentes que lo que hoy en día todos cargamos en el bolsillo, un celular. Como una idea general, un smartphone de gama alta del 2022 es aproximadamente 100000 veces más potente que la computadora del Apolo 11, misión espacial que llevó al hombre a la Luna en 1969.

Esta explosión de avances tecnológicos derivó en un desarrollo extremadamente acelerado, dando pie al desarrollo de una multitud de herramientas que en su momento parecían ‘de otro mundo’ y que hoy en día son indispensables. Tales herramientas, como lo es el Internet, permiten el intercambio de una cantidad incontable de información a lo

largo y ancho de todo el mundo. La Pandemia por coronavirus mostró la gran necesidad de hacer eficiente el uso de los recuros y datos en la red. El flujo de información fue tal, que el uso de la tecnología creció considerablemente y nos dejó claro la necesidad estos recursos para realizar las tareas que diariamente realizamos. Tanto en términos de educación, trabajo, entretenimiento, etc., La sociedad actual y los avances que se presentan dependen en gran medida de la capacidad para procesar y compartir información [1].

La cantidad de ataques y detección de vulnerabilidades creció también durante esta pandemia. De acuerdo con datos de PCW, los ciberataques son cada vez más frecuentes y sofisticados, desde phishing hasta ataques de malware en la cadena de suministro o en los servicios de nube. De acuerdo con los resultados de la Digital Trust Insights 2022, más de la mitad (58%) de empresas mexicanas aumentó su presupuesto para atender un creciente número de riesgos [2]. El uso de palabras por moda ha sido una de las estrategias para llamar la atención de posibles víctimas en ataques de ingeniería social aumentando los casos de phishing en correo electrónico en los últimos años.

Pero no solo en la pandemia las empresas o usuarios de internet han sido víctimas de ciberataques, dentro de los más comunes efectuados para el robo de información son:

1.1 Phishing

Ataca al “eslabón más débil” dentro de la ciberseguridad: el usuario final. Usa técnicas de ingeniería social para engañar al usuario, uno de los ejemplos más recientes fue al incluir correos con enlaces que incluyen información sobre la COVID-19 debido a la vigencia del tema en la sociedad.

1.2. Malware

La manera en que se almacena la información no es segura en la mayoría de los casos, es por esto que los ciber atacantes aprovechan esta vulnerabilidad para robar los datos de acceso de los usuarios por medio de software que se ejecuta sin conocimiento ni permiso del administrador.

1.3 Ataques de fuerza bruta y diccionario

El más común de los ataques consiste en que el atacante intente adivinar la contraseña del usuario usando las palabras más comunes que hacen uso, por ejemplo “1234” o “admin” de esta manera no se requiere un elaborado plan para el robo de información, simplemente una técnica de ciberseguridad mal implementada. Actualmente es posible encontrar bases de datos de contraseñas comunes para usar en ataques y acelerar el robo de cuentas.

1.4 Vulnerabilidades de sistema y de sitio web

Como se mencionó en el punto anterior, los ataques más sencillos se dan entre las vulnerabilidades del sistema, de esta manera, el atacante intenta explotar las vulnerabilidades y fallas de seguridad para leer, modificar y cargar archivos dentro de la infraestructura.

Queda claro entonces, la importancia de los sistemas de comunicación y del manejo de información en la actualidad, además de la necesidad de protegerlos. Es también importante, implementar estrategias de análisis con el fin de identificar las herramientas de detección en todas las áreas de ciberseguridad para reducir los casos de poder vulnerar sistemas e información [3].

En particular encontramos la existencia de dos tipos de sistemas para prevenir y detectar intrusiones. Por un lado, un IPS (sensor de prevención de intrusiones) es un IDS en la mayoría de los aspectos, excepto por el hecho de que puede actuar en línea sobre el tráfico actual. Debe estar en línea y por lo tanto solo puede ver el tráfico que entra y sale de un área. Las acciones de IPS incluyen descartar, restablecer, evitar o acciones con secuencias de comandos personalizadas y todo esto ocurre inmediatamente después de la coincidencia de la firma. En cambio, un IDS, es una herramienta que muy fácilmente detecta intrusiones. Para cada red, se encuentra analizando paquetes en busca tráfico inusual basado en políticas y bases de datos actualizadas con el malware vigente.

En la figura 1 se muestran los diagramas de funcionamiento de un IPS y de un IDS. Es importante resaltar la capacidad de emisión de alertas en el IDS en una comunicación que ha traspasado el firewall de una red corporativa, mientras que en el IPS se previene la intrusión bloqueando la comunicación. Ambos sistemas tienen la ventaja de aprender basados en el entorno que se implementen.

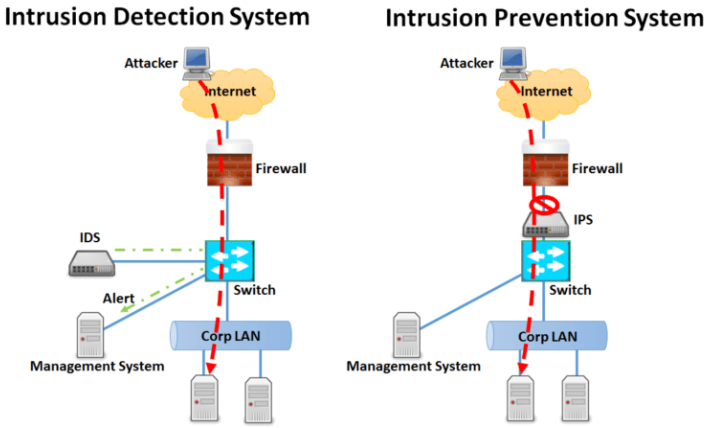


Fig.1. Diagramas de funcionamiento de IDS y de IPS en una red con acceso a internet [4].

2 Sistemas de Detección de Intrusos

En la actualidad es casi imposible no encontrar sistemas de comunicación, los cuales por sus cualidades de transmitir activos tan importantes como la información, es importante establecer protección, como menciona el sitio INCIBE “actúan monitorizando el tráfico que entra y sale de la red”. De manera general, se describen tres sistemas, encontrándose sistemas de prevención y detección [5].

2.1 Clasificación

Un IDS (Sistemas de Detección de Intrusos), tiene el objetivo principal de evitar conexiones indeseables. Hoy en día se pueden encontrar diversas opciones adaptables a nuestras necesidades, por ejemplo:

- Basados en firmas: es el que monitorea el tráfico de la red para localizar patrones inusuales o código malicioso por medio de la detección de firmas de ataque que se encuentran en los encabezados de los paquetes. Estas firmas se convierten en una base de datos conocidas para solo identificar y alertar al administrador.
- Basados en anomalías: detecta ataques más sofisticados, ya que actualmente el malware implementa inteligencia artificial y aprendizaje automático dando como resultado que se detecten ataques nuevos, sin embargo su función se limita a detectar y alertar al administrador [7].

2.2 Suricata

Suricata es un software de análisis de red y detección de amenazas de código abierto y alto rendimiento utilizado por la mayoría de las organizaciones públicas y privadas, e integrado por los principales proveedores para proteger sus activos [8]. Después de realizar un análisis general de las funciones de diversos IDS se eligió el IDS Suricata. Para estas pruebas se ejecutó Suricata en sobre la distribución Ubuntu Linux. Asignando las correspondientes características y añadiendo el repositorio al sistema para obtener la versión estable del IDS.

Se realiza la configuración para su uso en el archivo ubicado en la ruta */etc/suricata/* y de nombre *suricata.yaml*. Las configuraciones se realizan en los apartados *Community* e *Interface* para habilitar las funciones y poder cambiar la red por defecto que se maneja en el dispositivo.

Una vez iniciado el servicio de Suricata es posible implementar reglas de acuerdo a las necesidades del administrador y de la red que maneja. Es importante corroborar la habilitación de los archivos log (bitácoras del sistema), para analizar cada paquete ICMP que se genere o llegue al servidor. Es importante crear un archivo para el manejo de reglas en el directorio rules de suricata y que lleva el nombre por lo general de *my_rules.* Los paquetes y/o conexiones de *ping* que llegan al servidor son detectados por el IDS como alertas.

Suricata también permite el rechazo de conexiones por medio de una segunda regla. Se sigue el mismo proceso, en este caso para que rechace todas aquellas conexiones que

tenga que ver con Facebook por medio de la función *drop* para Facebook.

Una de las características de Suricata, consiste en que si se mantiene abierta la bitácora (log) se puede verificar su funcionamiento, con el comando *Curl -i Facebook.com* si se desea analizar las conexiones a este dominio.

Se encuentra, que efectivamente, la alerta era notificada dentro del archivo log.

Cabe destacar que las características más destacas de este IDS son:

- Analiza el tráfico de red por medio de reglas predeterminadas para detectar ataques y comportamientos inusuales.
- Soporte para la programación e inserción de scripts para la detección de amenazas más complejas.
- Detección automática de protocolos
- Análisis y registro de solicitudes de diversos protocolos, como por ejemplo HTTP, HTTPS, DNS, TSL/SSL entre otros [8].

2.2 Tripwire

Como segundo IDS se utilizó Tripwire en un Subsistema Linux. Este IDS nos permite registrar eventos de tipo cambio/adición a los archivos de nuestro sistema. Tripwire lo alerta sobre cambios no planificados y automatiza la remediación para fortalecer sus sistemas de manera proactiva y reducir su superficie de ataque. Detecte, priorice y neutralice las amenazas con la gestión de vulnerabilidades (VM) [9].

La instalación es interactiva, y como primer paso preguntará si se quiere añadir un e-mail para recibir notificaciones de alertas. Esto se logra seleccionando “internet site” para configurar el e-mail. Requiere la configuración de contraseñas y configuración de preguntas, para reforzar la seguridad. Es importante mencionar que requiere de una base de datos para generar una base de conocimiento que permita evolucionar en la protección de sistemas.

También es necesaria la configuración de un archivo de reglas que debe convertirse en un archivo cifrado para que el IDS pueda realizar su tarea. Como resultado entrega un reporte listando los cambios desde la última vez que se actualizó la base de datos del IDS Dentro de las principales ventajas y características de Tripwire se encuentran:

- Mapea la red: identifica cada activo y analiza el tráfico en la red.
- Solucionar vulnerabilidades: detecta ataques sin interrumpir las operaciones.
- Vectores de ataque de bloques: El modelado de amenazas muestra cómo proteger los activos más sensibles.
- Controles automáticos de seguridad: La gestión de cambios y el registro de eventos detectan desviaciones.

3 Conclusiones y Trabajos Futuros

En la realización de esta investigación, se pudieron identificar los sistemas de detección de intrusos y sus características, de esta manera, se analizaron las cualidades que permitían su uso, y al determinar el uso de Suricata, se identificó su funcionamiento y el

establecimiento de reglas con éxito en la configuración de reglas e identificación de intrusiones. La principal ventaja del análisis de este IDS es la visualización de ataques hacia las conexiones y/o comunicaciones en tiempo real.

Se complementa con Tripwire debido a las alertas que lanza sobre cambios en contenido de los archivos que se generan en el sistema operativo, proveyendo al administrador de una visualización completa tanto en conexiones como en la estabilidad del sistema.

La implementación de IDS aunado a sistemas de monitoreo permitirá lograr una visión general del sistema sin necesidad de dirigir los esfuerzos de análisis de forma particular. La implementación de estas herramientas deriva positivamente en la adquisición de información sobre el comportamiento del sistema, software, conexiones y comunicaciones. Se espera seguir implementando diversas reglas en Suricata o en otros IDS, así como implementar laboratorios de pruebas para probar otros IDS y generar una mejor visión sobre el avance en la detección de intrusos de manera eficiente.

Referencias

1. Jiménez, J. (2022, 7 noviembre). Qué es y cómo funciona un sistema de detección de intrusos. RedesZone. https://www.redeszone.net/tutoriales/seguridad/sistema-deteccion-intrusos/
2. Ciberseguridad y privacidad de datos. (s. f.). PWC. Recuperado 2 de agosto de 2022, de https://www.pwc.com/mx/es/ciberseguridad.html
3. Pérez, J. L. R. (2014a). Técnicas de aprendizaje automático para la detección de intrusos en redes de computadoras. redalyc.org. Recuperado 1 de junio de 2022, de https://www.redalyc.org/journal/3783/378368201003/html/
4. Consentini, D. (2022, 18 julio). Prevención de amenazas, IDS / IPS. Sysbeards Tu Blog De Tecnología. https://sysbeards.com/prevencion-de-amenazas-ids-ips/
5. ¿Qué son y para qué sirven los SIEM, IDS e IPS? (2020, 3 septiembre). INCIBE. https://www.incibe.es/protege-tu-empresa/blog/son-y-sirven-los-siem-ids-e-ips
6. Ramírez, H. (2022, 6 junio). El sistema de detección de intrusiones (IDS). Grupo Atico34. https://protecciondatos-lopd.com/empresas/sistema-deteccion-intrusiones-ids/
7. Sniffer de red. (2a. C.). www.avast.com. Recuperado 21 de mayo de 2022, de https://www.avast.com/es-es/c-sniffer
8. Suricata IDS. (s. f.). suricata. Recuperado 17 de mayo de 2022, de https://suricata.io/
9. IDS Suricata. (s. f.). www.mancomun.gal. Recuperado 25 de abril de 2022, de https://www.mancomun.gal/es/solucion-tic/suricata/#:~:text=Caracter%C3%ADsticas%20m%C3%A1s%20destacadas%20de%20Suricata,detecci%C3%B3n%20de%20amenazas%20m%C3%A1s%20complejas
10. Implementación de un honeypot. (s. f.). www.welivesecurity.com. Recuperado 8 de mayo de 2022, de https://www.welivesecurity.com/la-es/2020/07/31/que-es-honeypot-como-implementarlo-nuestra-red/
11. Tripware. (s. f.). Recuperado 27 de abril de 2022, de https://www.tripwire.com/

Desarrollo de un Simulador de Exámenes en Línea con UWE para el Acompañamiento en el Aprendizaje de la Ingeniería de Software

José Miguel López-Aguilera, Mario Rossainz-López, Bárbara Sánchez-Rinza

Benemérita Universidad Autónoma de Puebla,
Avenida San Claudio y 14 sur, San Manuel, Puebla, Puebla, 72000, México
miguel.lopezag@alumno.buap.mx, {mario.rossainz, barbara.sanchez}@correo.buap.mx

Resumen. Mostramos el desarrollo de un Simulador de exámenes en línea de las materias de Ingeniería de Software que se imparten en los programas educativos (PE) de la Facultad de Ciencias de la Computación (FCC) de la BUAP. El objetivo es, por un lado, que los profesores diseñen exámenes de simulación y, por otro lado, que los alumnos se ejerciten y refuercen el aprendizaje adquirido al realizar los exámenes simulados y que ello garantice una buena nota en la calificación de los exámenes reales. La motivación tiene su origen en la problemática del rendimiento académico de los alumnos y al bajo índice de aprovechamiento. Para el diseño del sistema se utiliza UWE. Para la creación de las preguntas se utiliza el diseño instruccional de reactivos del CENEVAL y se concluye con un comparativo del aprovechamiento de las materias de ingeniería de software de seis años atrás a la fecha con y sin el simulador.
Palabras Clave: Ingeniería de Software, UML, UWE, CENEVAL

1 Introducción

El presente trabajo tiene como objetivo principal mostrar el desarrollo y utilidad de una aplicación web destinada a la simulación de exámenes de las materias de Ingeniería de Software de los programas educativos de la Licenciatura e Ingeniería en Ciencias de la Computación, así como de la Ingeniería en Tecnologías de la Información de la Facultad de Ciencias de la Computación de la BUAP, con la finalidad de que dicha propuesta sirva como acompañamiento en el aprendizaje que el alumno adquiere al cursar este tipo de asignaturas y tenga más posibilidades de acreditar la materia. Durante los últimos años, desde el año 2016 al año 2021 (tomando en cuenta una sección por año de cualquier materia de Ingeniería de software de los 3 programas educativos de la FCC) [1], el rendimiento académico del estudiante en esta área no ha sido el que se esperaría pues el índice de aprobación de la materia está por debajo de la media, es decir, de 40 estudiantes que aproximadamente tiene un grupo normal de una materia de Ingeniería de software, en general solamente 14 de ellos acreditan la asignatura con una calificación entre 6 y 10, y el resto la reprueban. Esto conlleva a una serie de afectaciones en el historial y rutas críticas de éstas asignaturas que provocan un atraso en el plan de vida académico marcado por el alumno [2]. Para mayor detalle sobre el análisis de la problemática de los índices

de reprobación de estas materias, la metodología propia del simulador propuesto y resultados explícitos de los logros obtenidos por los estudiantes al utilizar el simulador respecto de sus calificaciones obtenidos se sugiere al lector referirse a [3] que es el trabajo de tesis de donde surge el presente escrito, ya que por motivos de espacio y restricciones propias del template nos es imposible detallar al respecto. Una alternativa para ayudar a revertir estos números es la propuesta que se presenta en este trabajo. El desarrollo del sistema web simulador de exámenes de las materias de ingeniería de software que se propone se basa formalmente en el uso de la metodología UWE (UML-Web Engineering) que es una metodología ad hoc para desarrollo de sistemas web orientados a objetos [4]. Esta metodología se basa en el Proceso Unificado de Desarrollo de Software o PUDS y en el modelado con UML bajo una semántica distinta, de manera que los diagramas utilizados en UML representen parte del diseño que se suele hacer cuando se desarrollan aplicaciones web: mapas de sitio, story-boards, wireframes, interfaces gráficas de usuario, etc., representados con diagramas de actividades, diagramas de clases, diagramas de transición de estados, etc.

Por otro lado, el diseño de las preguntas que formarán parte de un examen en el sistema se basa en el diseño instruccional que utiliza el CENEVAL para la producción de reactivos en los exámenes de certificación que ofrece tanto a particulares como a instituciones educativas. Ese diseño lo genera la aplicación y es el profesor titular de la materia el que se encarga de generar las distintas preguntas que son de opción múltiple y que constituirán el banco de preguntas de los exámenes que a manera de simulación un estudiante puede realizar para reforzar sus conocimientos de forma que con ello garantice una mayor probabilidad de acreditar un examen real de su materia. Finalmente, el sistema propuesto es utilizado como prueba piloto en dos asignaturas particulares del área de ingeniería de software en el periodo de primavera 2022: en la materia de Ingeniería de Software y en la materia de Ingeniería de Software Avanzada, ambas pertenecientes al programa educativo de la Ingeniería en Ciencias de la Computación con una población de 40 alumnos en cada asignatura. Los resultados que se encuentran en el apartado correspondiente son interesantes y muestran la utilidad que puede tener una herramienta como la que se presenta en este trabajo para ayudar por un lado al profesor a tener el control y seguimiento académico de sus alumnos y por otro lado a los alumnos a reforzar los conocimientos teórico-prácticos que va adquiriendo para así tener una mayor probabilidad de acreditar la materia.

2 Marco Teórico y Antecedentes

Los Programas Educativos (PEs) de nivel licenciatura de la FCC de la BUAP tienen dentro de su currícula de materias aquellas relacionadas con la Ingeniería de Software en donde se estudian métodos y técnicas del desarrollo de software en sus distintas vertientes [5], [6], [7]: desarrollo clásico de sistemas, desarrollo de sistemas orientados a objetos, orientados a componentes, desarrollo de sistemas ágiles, etc. Estas materias se encuentran en los niveles formativos de los 3 PE en donde se imparten y forman parte de rutas críticas

importantes en donde la no acreditación de una de estas materias significa un retraso significativo en materias siguientes que tienen como prerrequisito para ser cursadas la aprobación de todas estas materias. Por otro lado, los perfiles de egreso tienen una estrecha relación con las materias de Ingeniería de Software ya que los conocimientos y habilidades adquiridos en ellas por parte del alumno ayudan a formar profesionistas capaces de liderar equipos de trabajo y trabajar de forma colaborativa con ellos, diseñar soluciones, administrar proyectos, y aplicar metodologías y técnicas en el desarrollo avanzado del software, entre otros [5], [6], [7]. Sin embargo, por distintas circunstancias, este tipo de materias suelen ser muy complicadas para el alumno en cuanto a su aprovechamiento y acreditación. Regularmente materias como Ingeniería de Software, Ingeniería de Software Avanzada, Ingeniería de Software I, Ingeniería de Software II, se incluyen dentro del conjunto de materias que suelen ser difíciles acreditarlas para el alumno [8]. La gráfica de la Fig.1, muestra la tendencia del aprovechamiento que los alumnos tienen regularmente en materias de ingeniería de software dentro de los PE que estudian. En ella se observa que año con año el índice de aprobación está por debajo del índice de no-aprobación. En promedio durante estos seis años el índice de aprobación de los cursos de ingeniería de software es de un 34% mientras que el índice de no-aprobación es de un 66%.

Ésta ha sido la motivación principal que ha dado origen a la propuesta que aquí se presenta, desarrollando una aplicación web que se utilice como una herramienta de aprendizaje para que a través de la realización simulada de exámenes tomando como base de diseño el que se utiliza en el CENEVAL para la elaboración de reactivos que conforman los exámenes de conocimientos generales para la titulación como Ingenieros o Licenciados en Ciencias de la Computación y afines [9]; los alumnos se ejerciten en los conocimientos tanto teóricos como prácticos que estudian en materias relacionadas con la Ingeniería del Software y con ello incrementen sus probabilidades de aprobar esas materias y se pueda empezar a revertir la tendencia mostrada en la gráfica de la Fig.1. Adicionalmente, el profesor titular de la materia puede utilizar el sistema simulador como un complemento en sus clases, diseñando preguntas que formarán parte de los exámenes simulados por la aplicación y con ello promover un uso más frecuente de esta aplicación para complementar el aprendizaje del alumno y poder también con la aplicación identificar a aquellos alumnos que les cuesta trabajo llevar la materia para enfocar la atención en ellos y poder ofrecer una asesoría más personalizada.

Cabe aclarar que si bien las herramientas de gestión de conocimiento como Google-Classroom, Moodle o Blackboard proporcionan al profesor y alumno la generación y realización de exámenes, éstos no son considerados exámenes de entrenamiento como lo es nuestra propuesta. Lo presentado en este escrito no pretende ofrecer la generación de exámenes para evaluación sino una herramienta de entrenamiento para el alumno particularmente en los conocimientos teórico-prácticos que se requieren para el reforzamiento y posible acreditación de las asignaturas de ingeniería de software que se imparten en la Facultad de Ciencias de la Computación. La herramienta facilita al profesor en la elaboración de un examen de simulación en donde el docente diseña las preguntas que considere pertinentes para reforzar cierto conocimiento adquirido en el aula y el alumno al resolver el examen va adquiriendo un entrenamiento de dichos

conocimientos ya que al final de éste ejercicio el alumno obtiene retroalimentación del sistema mostrándole no sólo su resultado, sino lo que debería de haber respondido como una respuesta correcta a una pregunta que erró. El banco de preguntas del sistema propuesto se va generando conforme el profesor vaya diseñando sus reactivos. Su tamaño depende de la cantidad de preguntas que el profesor quiera diseñar para la generación de los exámenes de reforzamiento.

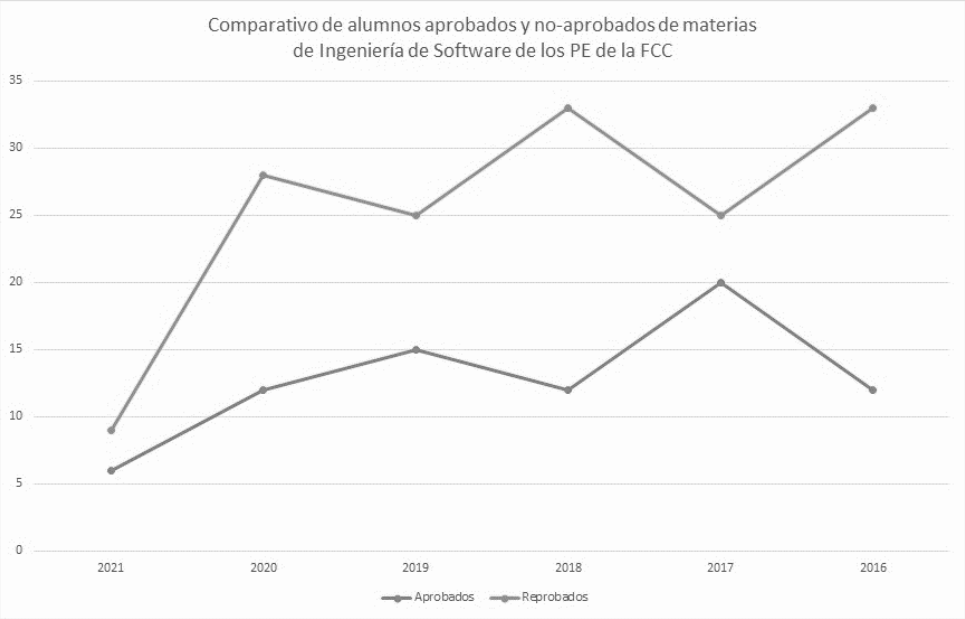


Fig. 1. Tendencia del índice del aprovechamiento de alumnos que cursan materias de ingeniería de software de los PE de la FCC de la BUAP en una sección por periodo del 2016 al 2021.

3 Diseño Instruccional de Reactivos del CENEVAL

Para el diseño de las preguntas (las cuales son de opción múltiple con 4 posibles respuestas de las cuales sólo una es la correcta) que el profesor titular de una materia de Ingeniería de Software puede elaborar con el simulador web, se decidió utilizar el diseño instruccional del CENEVAL (asociación civil que tiene como actividad principal el diseño y aplicación de instrumentos de evaluación de conocimientos, habilidades y competencias; así como el análisis y la difusión de sus resultados [10]) para la elaboración de reactivos. Los exámenes del CENEVAL tienen validez oficial en todo el país y con ellos universidades públicas y privadas adoptan este proceso de evaluación como parte del abanico de formas de titulación que ofrecen a sus egresados, entre ellas la BUAP y la FCC ofrece esta alternativa de titulación a sus egresados y pasantes [11]. El diseño

instruccional del CENEVAL para la elaboración de un reactivo se adaptó a nuestra propuesta como se muestra a continuación a manera de ejemplo:

<table>
<tr><td>ASIGNATURA</td><td>Ingeniería de Software I</td></tr>
<tr><td colspan="2">UNIDAD/TEMA</td></tr>
<tr><td colspan="2">Análisis y Estimación de Costos de los Sistemas Software</td></tr>
<tr><td colspan="2">PREGUNTA</td></tr>
<tr><td colspan="2">Se requiere desarrollar un sistema software de aplicación de Diseño Asistido por Computadora (CAD) de componentes mecánicos. La descomposición de éste sistema software por funciones y sus respectivas líneas de código estimadas (LDC) son las siguientes:
<table>
<tr><td>FUNCIÓN</td><td>LDC Estimadas</td></tr>
<tr><td>IUFC: interfaz de Usuario y Facilidades de Control</td><td>2300</td></tr>
<tr><td>AG2D: Análisis Geométrico de 2D</td><td>5300</td></tr>
<tr><td>AG3D: Análisis Geométrico de 3D</td><td>6800</td></tr>
<tr><td>GBD: Gestión de Base de Datos</td><td>3350</td></tr>
<tr><td>FPGC: Facilidades de Presentación Gráfica por Computadora</td><td>4950</td></tr>
<tr><td>CP: Control de Periféricos</td><td>2100</td></tr>
<tr><td>MAD: Módulos de Análisis del Diseño</td><td>8400</td></tr>
<tr><td>Líneas de Código Estimadas</td><td>33200 LDC</td></tr>
<tr><td>Puntos de Función Estimados</td><td>372 PF</td></tr>
</table>
Calcule el costo de estimación de éste sistema software usando LDC suponiendo una productividad media para sistemas de este tipo según históricos de 620 LDC/pm (donde pm=persona-mes) equivalentes a 6.5 PF/pm con una tarifa laboral de $8000.00 US/m</td></tr>
<tr><td>OPCIÓN DE RESPUESTA</td><td>TIPO DE RESPUESTA</td><td>ARGUMENTACIÓN</td></tr>
<tr><td>A) 457560.00 US</td><td>Incorrecta</td><td>Es el cálculo del costo estimado usando puntos de función ($8000.00/6.5)*372=457560.00</td></tr>
<tr><td>B) 431600.00 US</td><td>Correcta</td><td>Se calcula el costo por línea de código: $8000.00/620= $13.00 US y éste resultado se multiplica por el número total de líneas de código estimadas: 33200*13= 431600.00 US.</td></tr>
<tr><td>C) 432000.00 US</td><td>Incorrecta</td><td>(33200 LDC)/(620 LDC/pm)= 54 pm y luego 54*8000.00=432000.00 lo cual es erróneo pues la primera operación calcula el esfuerzo estimado usando líneas de código.</td></tr>
<tr><td>D) 456000.00</td><td>Incorrecta</td><td>(372 PF)/(6.5 PF/pm)= 57pm y luego 57*8000.00 lo cual es erróneo pues la primera operación calcula el esfuerzo estimado usando Puntos de Función.</td></tr>
</table>

4 UML-Based Web Engineering (UWE)

Para el desarrollo del Simulador de Exámenes en Línea se utilizó UWE que son las siglas en ingles de “Ingeniería Web basada en UML” y es una metodología de ingeniería de software para desarrollar aplicaciones web que está basada en UML [12], [13]. UWE usa notación y diagramas UML puro, siempre que sea posible para el análisis y diseño de

aplicaciones web sin extensiones de ningún tipo, sin embargo, para modelar características específicas de la Web como por ejemplo los enlaces de la estructura de un hipertexto, UWE incluye estereotipos, etiquetas y restricciones definidas en su semántica para los elementos de modelado que se utilicen de UML [13]. UWE cubre aspectos de modelado y diseño de diagramas de navegación, presentación, procesos y adaptación de aplicaciones web (ver Fig.2) [14].

5 Desarrollo del Simulador de Exámenes en Línea

El desarrollo del Simulador de Exámenes en Línea se llevó a cabo utilizando UWE y constó de las siguientes etapas: Un análisis de requerimientos -diagrama de casos de uso-, una realización de cada caso de uso -diagramas de actividades-, el diseño del sitio web (mapa de sitio e history-boards o wireframes) – Diagramas de navegación y Diagramas de Presentación-, diseño de procesos -diagramas de contenido- (ver Fig. 3) y finalmente el diseño y creación de la base de datos -diagrama E-R, modelo relaciona y diseño de consultas en SQL (ver Fig.4).

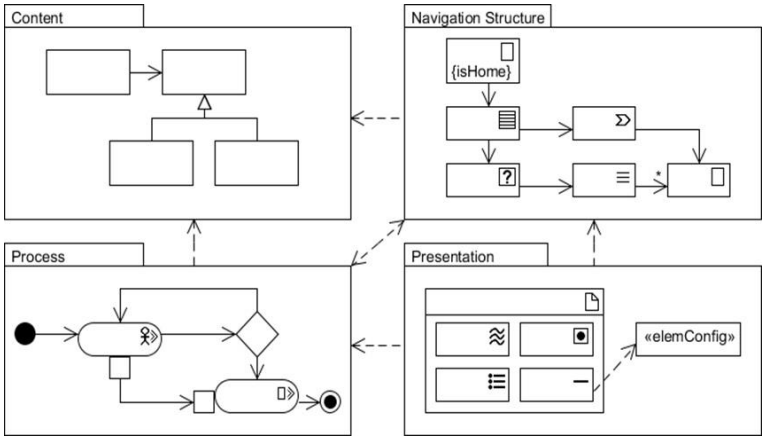


Fig. 2. Diagramas modelo para el diseño de contenido, navegación, procesos y presentación en el desarrollo de una aplicación web con UWE.

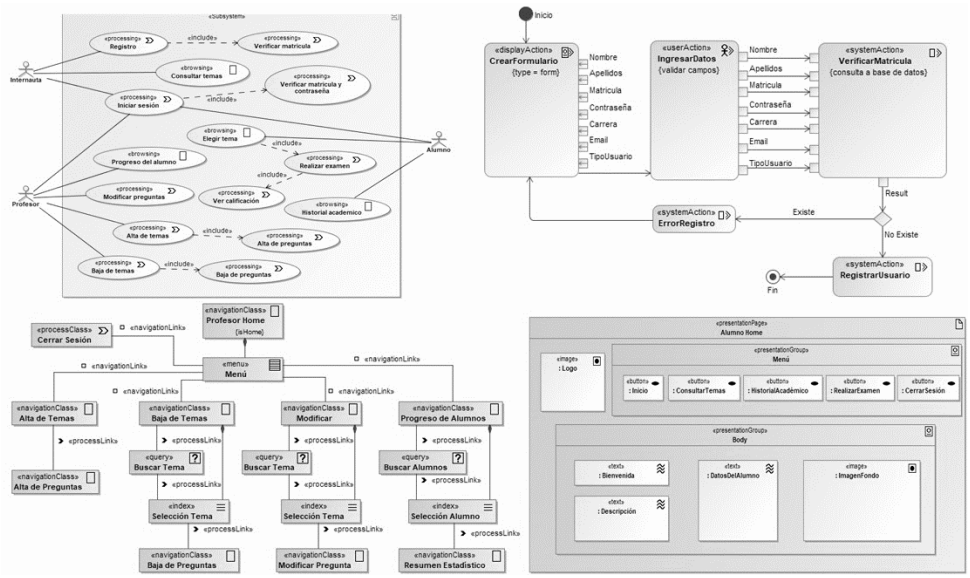


Fig.3. Análisis y Diseño del Simulador Web de exámenes utilizando UWE

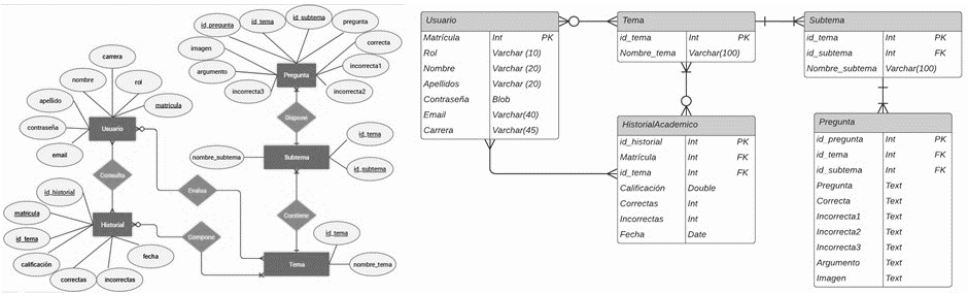


Fig. 4. Modelado y Diseño de la Base de Datos relacional del Simulador Web de exámenes

Finalmente, en la Fig.5 se muestra un collage del front-end de la aplicación web desarrollada tanto para la parte del profesor como para la parte del alumno.

Fig. 5. Front-end inicial del simulador de exámenes que muestra parte de lo que es su implementación

6 Resultados y Conclusiones

Se aplicó la metodología UWE para desarrollar un simulador web de exámenes para las materias de Ingeniería de Software que se imparten en los distintos programas educativos que oferta la Facultad de Ciencias de la Computación de la BUAP con el objetivo de servir como acompañamiento en el aprendizaje que el alumno adquiere en este tipo de asignaturas debido a la problemática que experimenta el alumno en cuanto a su rendimiento académico y al bajo índice de aprovechamiento que se presenta año con año al tratar de aprobar la materia (ver Fig.1). La creación de las preguntas que forman parte de un examen de simulación y que son creadas por el profesor a través del sistema web propuesto, se generaron tomando como base el diseño instruccional de reactivos que utiliza el CENEVAL para la elaboración de los exámenes de certificación que ofrece. Las características finales que ofrece el simulador de exámenes a sus usuarios finales son las siguientes: El sistema genera exámenes por temas y cada examen se construye de una serie de preguntas asociadas a los temas a evaluar que son elegidas al azar de un banco almacenado en una base de datos. Todas las preguntas son de respuestas de opción múltiple y cada pregunta tiene 4 posibles respuestas donde una de ellas es la correcta. Para el caso del alumno, éste se registra, autentifica y selecciona el tema en el que se quiere evaluar. El sistema le muestra al alumno un resumen estadístico de los exámenes que ha realizado con el fin de que el alumno tenga una retroalimentación del número de respuestas correctas e incorrectas que tuvo en el examen que realizó, si reprobó o no el

examen, la fecha de realización del examen y el número de veces que ha realizado dicho examen. Para el caso del profesor, éste funge como el administrador del sistema y es quien actualiza los temas, preguntas y respuestas (agregar, eliminar o modificar) de la materia de la cual es titular, y tiene el control de sus alumnos registrados. El profesor también se registra y autentifica para usar la aplicación. Toda la información con la que trabaja el sistema se encuentra almacenada en una base de datos relacional. Finalmente, para demostrar la utilidad de esta propuesta, se realizó una prueba piloto del uso del simulador de exámenes en una de las secciones de las materias de Ingeniería de Software e Ingeniería de Software Avanzada del PE de la Ingeniería en Ciencias de la Computación de la FCC de la BUAP del periodo lectivo de primavera 2022 con 45 alumnos en cada sección. Los resultados en porcentajes fueron: para el caso de la materia de Ingeniería de Software el 80% de los alumnos aprobaron la materia, mientras que para Ingeniería de Software Avanzada se tuvo un porcentaje de aprobación del 71%. Lo anterior contrasta con los resultados obtenidos de la gráfica de la Fig.1, cuya actualización se muestra a continuación en la gráfica de la Fig.6.

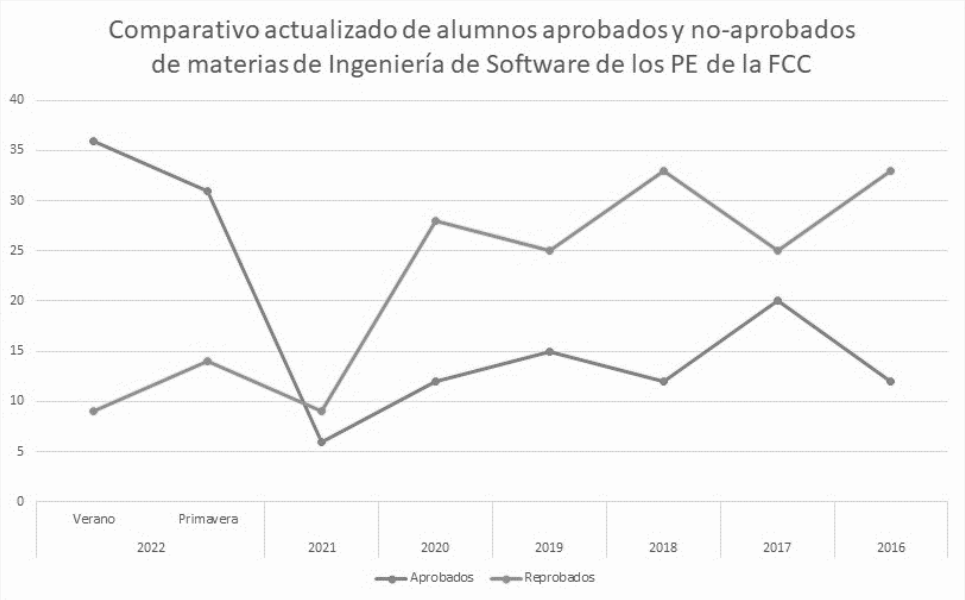


Fig. 6. Tendencia actualizada del índice absoluto del aprovechamiento de los alumnos que cursan materias de ingeniería de software de los PE de la FCC de la BUAP en por lo menos una sección por periodo del 2016 al 2022 con el uso del simulador de exámenes.

Referencias

1. KPI-IngSw. (2020). Análisis de Indicadores Claves de Rendimiento de las materias del área de Ingeniería de Software y Base de Datos de la Fac. de Cs. de la Computación de la BUAP. México.
2. Secretaría Académica (2017). Mapa Curricular de Rutas Críticas del Programa Educativo de la Ingeniería en Ciencias de la Computación de la FCC de la BUAP. México. Recuperado de: https://secreacademica.cs.buap.mx/MumMapas/Semestres/Mapa_ICC%202017_071217.pdf
3. López-Aguilera J.M., (2022). Simulador Web de Exámenes de la Materia de Ingeniería de Software para el Acompañamiento en el Aprendizaje del Alumno. Tesis de Licenciatura en Ciencias de la Computación. Fac. Cs. De la Computación. BUAP. Puebla, México
4. Nieves_Guerrero C., Ucán-Petch J., Menendez-Rodriguez V., (2014). UWE en Sistema de Recomendación de Objetos de Aprendizaje. Aplicando Ingeniería Web: Un método en caso de estudio. Revista Latinoamericana de Ingeniería de Software. Volumen 2, Número 3. México.
5. FCC (2017). Plan de Estudios de la Licenciatura en Ingeniería en Ciencias de la Computación. BUAP. Puebla, México. Recuperado de: https://secreacademica.cs.buap.mx/MumMapas/Semestres/planEstud/Programa%20Educativo%20Sintesis%20_ICC%202016.pdf
6. FCC (2017). Plan de Estudios de la Licenciatura en Ciencias de la Computación. BUAP. Puebla, México. Recuperado de: https://secreacademica.cs.buap.mx/MumMapas/Semestres/planEstud/Programa%20Educativo%20S%C3%ADntesis%20LCC%202016.pdf
7. FCC (2017). Plan de Estudios de la Licenciatura en Ingeniería en Tecnologías de la Información. BUAP. Puebla, México. Recuperado de: https://secreacademica.cs.buap.mx/MumMapas/Semestres/planEstud/Programa%20Educativo%20S%C3%ADntesis%20ITI%2027102017.pdf
8. FCC. (2016). Contenido de Materias de los PE de la FCC de la BUAP. Modelo Minerva. Puebla, México. Recuperado de: https://secreacademica.cs.buap.mx/matICCS.html, https://secreacademica.cs.buap.mx/matCCOS2021.html, https://secreacademica.cs.buap.mx/matITIS.html
9. Herrera-Ortiz M., Marín-Martínez A., Rodríguez-Pérez A., (2020). Taller de Elaboración de Reactivos de Opción Múltiple. CENEVAL. México. Recuperado de: https://ceneval.edu.mx/taller-de-elaboracion-de-reactivos/
10. CENEVAL. (2020). Sobre el CENEVAL. México. Recuperado de: https://ceneval.edu.mx/sobre_el_ceneval-perfil_institucional/
11. FCC (2022). Requisitos para obtener el oficio que indica la modalidad de titulación. Opción por EGEL-CENEVAL. México. Recuperado de: https://secreacademica.cs.buap.mx/alumnos/Titulacion2022/L%20Titulaci%C3%B3n-Ceneval.pdf
12. Kroi C., Koch N. (2008). The UWE Metamodel and Profile – User Guide and Reference. Ludwig-Maximilians-Universität München (LMU), Germany.
13. Sommerville I. (2011). Ingeniería de Software. Novena Edición. Addison-Wesley. México.
14. Et-Al (2010). Estudio de UWE (UML-Based Web Engineering). Universidad Carlos III de Madrid. España.

Metodología para Cifrar Información en Aplicación de Mensajería

Ana C. Zenteno, Gustavo T. Rubín, Judith Pérez, Emmanuel Márquez

Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla,
Av. San Claudio y 14 Sur, Ciudad. Universitaria, 72592 Puebla, Pue.
{ana.zenteno, gustavo.rubin, judith.perez}@correo.buap.mx,
emmanuel.marquezc@alumno.buap.mx

Resumen. El envío de información por canales inseguros en internet o dentro de una red local compromete organizaciones y personas ya que puede ser interceptada y exhibida. Las comunicaciones por mensajería han desplazado a medios tradicionales como el correo electrónico debido a la inmediatez con que se comunican los usuarios. El intercambio seguro de mensajes entre emisor y el receptor es una característica básica de seguridad en las aplicaciones de mensajería, por lo que la investigación e implementación de algoritmos de cifrado derivan en la propuesta de una metodología que provea de seguridad a los usuarios. El intercambio de claves públicas y privadas en la aplicación que se ejecuta en servidor y cliente es indispensable en la actualidad para el cifrado de los mensajes.
Palabras Clave: Criptografía, Seguridad, Mensajería.

1 Introducción

Proteger la información es una tarea imperativa en las comunicaciones en internet, debido a la importancia de la información a nivel personal y profesional de los usuarios. La criptografía proviene de palabras griegas que significan "escritura oculta", por lo tanto, es el arte o la ciencia que utiliza métodos (algoritmos) para transformar un mensaje legible a uno que es ininteligible o cifrado y, por medio de un proceso inverso convertir el mensaje a su forma original. Los usuarios que cuenten con una clave secreta pueden descifrar el mensaje y dejarlo en texto plano. La criptografía se ha usado en comunicaciones telefónicas, fax, correo electrónico, transacciones monetarias, entre otros ejemplos en la internet. Actualmente se utiliza en la emisión e identificación de facturas digitales, incluyendo firmas electrónicas y certificados digitales para demostrar quién envió un mensaje [1].

La criptografía se define como un conjunto de métodos y/o técnicas matemáticas que son aplicadas a datos para ocultarlos de terceros que lleguen a interceptarlos. Lo que asegura que los datos sean privados, íntegros y auténticos. Existen principios para el manejo de información, entre ellos se encuentran: la *confidencialidad* que asegura por medio de autorización solo a personas con los permisos adecuados el acceso a la información. La *integridad* que asegura la no modificación de los datos durante la

transmisión. La *autenticidad* que asegura la comprobación de quien envía la información en quien dice ser por medio de firmas digitales [2]. El criptoanálisis tiene como objetivo buscar debilidades en dichos métodos matemáticos, con el fin de evadir la seguridad y conocer los datos ocultos, dado que la información pasa por etapas desde su creación hasta la recepción por parte del destinatario.

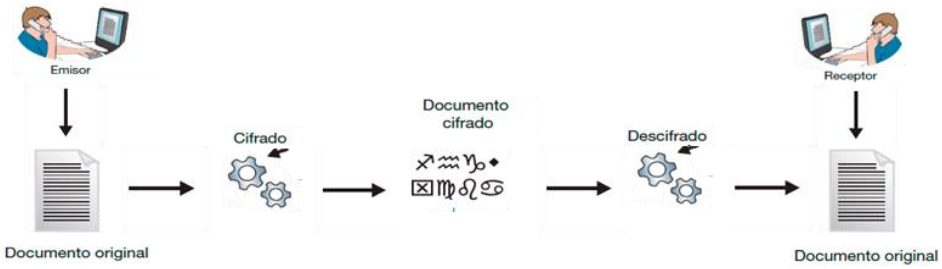


Fig. 1. Proceso de cifrado de un documento entre el emisor y receptor.

La criptografía es indispensable en el proceso de aseguramiento de la información. Ayuda también asegurar el proceso de transporte ocultando el contenido del mensaje aplicando diferentes técnicas de cifrado. Una de las técnicas más usadas es el cifrado asimétrico que utiliza distintas al menos dos claves, una para cifrar y otra descifrar el contenido de un mensaje y/o archivo. En este trabajo se implementa el algoritmo de cifrado RSA, que lleva su nombre debido a los autores que lo desarrollaron Rivest, Shamir y Adleman.

1.1 Clasificación según su algoritmo

Existen diversos algoritmos de cifrado, entre los más populares tenemos:

1.1.1 Cifrado simétrico

Este tipo de cifrado utiliza una misma clave para el método de cifrado y descifrado del mensaje, es decir, todos los usuarios comparten una clave la cual es privada.

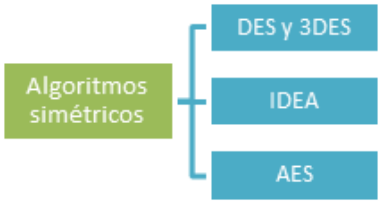


Fig. 2. Ejemplos de algoritmos simétricos.

1.1.2 Cifrado asimétrico

En este tipo existen dos claves una de ellas es publica con la que se debe encriptar los datos mientras que la clave privada que es secreta nos ayudara a descifrar los datos, ya que estas dos claves son complementarias.

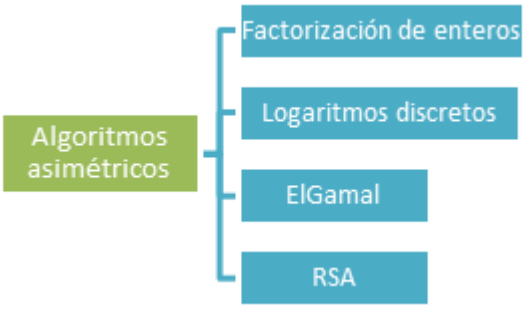


Fig. 3. Ejemplos de algoritmos asimétricos.

1.2 AES

Es un algoritmo de cifrado basado en sustituciones, permutaciones y transformaciones lineales, ejecutadas en bloque de datos de 16 bytes, que se repiten varias veces.

1.3 RSA

El algoritmo RSA es asimétrico el cual fue creado por Ron Rivest, Adi Shamir y Leonard Adleman. Implementa factorización de número para la seguridad que ofrece. Los mensajes se representan mediante números y mediante el producto de dos números primos grandes y elegidos al azar se realiza el cifrado.

1.4 Propuesta

Desarrollar una aplicación de mensajería instantánea en lenguaje Python donde se implementen algoritmos de cifrado para la comunicación entre usuarios. De tal forma que, ante la intercepción de los paquetes de mensajes de la aplicación, estos no puedan ser expuestos de manera inmediata.

2 Metodología

Se implementan dos algoritmos de criptografía asimétrica RSA y AES para el cifrado y descifrado de archivos respectivamente, en ambos algoritmos se usan números primos de distinta longitud que mediante operaciones a los datos permite el envío cifrado y de esta forma se ocultan los mensajes a los usuarios que no son autenticados por la aplicación para aplicar el proceso de descifrado de mensajes.

2.1 Implementación

Se desarrolla un chat colectivo en donde todos aquellos que compartan la misma contraseña podrán interactuar en la plática, de no tenerla solo verán mensajes no legibles.

La aplicación se forma utilizando un cliente y un servidor, los cuales están soportados por máquinas virtuales con distribuciones Linux para crear una red virtual de comunicación.

Servidor: La aplicación se mantiene en un ciclo de escucha para detectar nuevos usuarios lanzando un hilo cada vez que un usuario nuevo se conecta, ese hilo se mantendrá escuchando e imprimiendo en consola la conversación por ello los mensajes se encriptaran y descifraran en la aplicación del Usuario.

Usuario: Para esta aplicación ocuparemos demonios (hilos que se ejecutan en segundo plano) en primer plano tendemos la aplicación operando y en segundó plano las operaciones de enviar y recibir mensajes para ello al iniciar la aplicación se pedirán dos datos la contraseña con la que se encriptan y descifran los mensajes. Solo los usuarios que tengan la contraseña podrán ver los mensajes.

Los nombres de los usuarios en el servidor serán transparentes para así poder hacer alguna gestión de ser necesaria en el servidor mientras que la conversación podrá verla, pero cifrada.

3 Resultados

Se desarrolló el entorno de comunicación de chat multiusuario que permite acceso por contraseñas para interactuar en la conversación. De ser válida la contraseña se pueden leer apropiadamente los mensajes de la sala de chat. El servidor detecta las conexiones de los clientes y los valida para que en las ventanas de chat se lean los mensajes.

En la figura 4 se observa la comunicación entre cliente y servidor, el servidor observa caracteres que conforman el mensaje pero no los puede interpretar debido a que están cifrados; mientras que los clientes, una vez autenticados pueden comunicarse de manera normal ya que los mensajes se descifran.

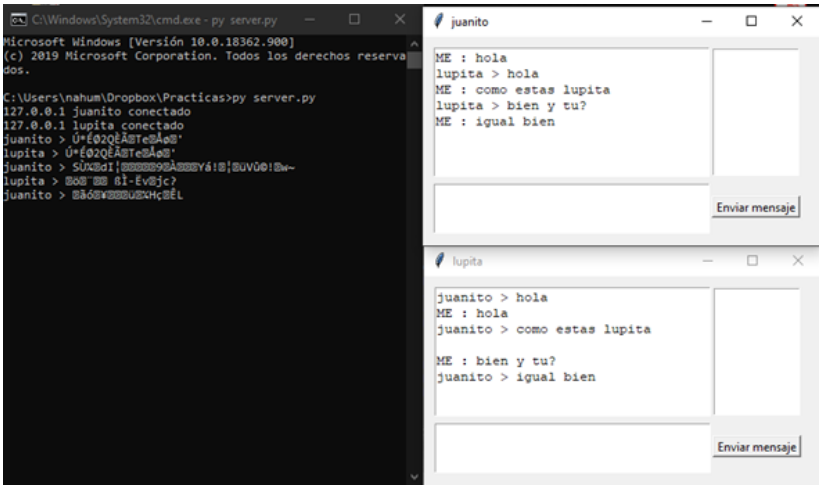


Fig.4. Resultado de la instalación del servidor web e interacción de comunicación en dos clientes de mensajería.

Herramientas como *wireshark* nos permiten la captura de paquetes en el tráfico de la red, es importante mencionar que si el texto fluye sin ser procesado es posible la intercepción, modificación o alteración de la información antes de la entrega al receptor. Este proceso representa una vulnerabilidad que debe ser contenida y revisada en las aplicaciones y en los mecanismos que realizan el tratado de la información. Se tiene como resultado en el desarrollo de este trabajo, el correcto cifrado de los datos y que ante la intercepción de paquetes con esta herramienta la información no puede ser interpretada de primera mano.

4 Conclusiones y Trabajos Futuros

El desarrollo de aplicaciones incluye actualmente la implementación de módulos de seguridad de la información en el desarrollo de software. El diseño e implementación de módulos de seguridad son atributos que pueden dar el éxito en el mercado de software en especial con el tratamiento de la información en el trayecto de emisor-receptor. La investigación de los diferentes algoritmos y el desarrollo de una estrategia resultan en un correcto cifrado de mensajes aunado al intercambio de las claves que permiten descifrar solo para los usuarios permitidos en una aplicación.

En el presente documento se destaca la pertinencia de la implementación de módulos de cifrado para asegurar la información ofreciendo una experiencia de privacidad en el uso de mensajería instantánea. En conjunto se evidencia que los algoritmos de cifrado con llaves públicas y privadas son efectivos en el desarrollo de software de mensajería que salvaguarda la privacidad de los usuarios. Corresponde a trabajos futuros realizar las implementaciones y pruebas para asegurar el proceso de modificación de la información antes de la entrega al receptor. Aunque es importante mencionar que con la gran cantidad

de software de mensajería que existen actualmente, también existe una gran cantidad de técnicas implementadas que abordan cada uno de los aspectos del tratado de información y que se recomienda actualizar los métodos para evitar futuras vulnerabilidades.

Referencias

1.Tomás M, Escudero José, (noviembre 2017). Análisis Comparativo de Cifrado Asimétrico algoritmos RSA y ElGamal. Recuperado el 14 de mayo de http://www.ecorfan.org/spain/researchjournals/Sistemas_Computacionales_y_TICs/vol3num10/Revista_de_Sistemas_Computacionales_y_TICS_V3_N10_5.pdf, ISSN-2444-5002

2. Mustafa M. (2017) Design Review on improvement of advanced encryption standard (AES) algorithm based on time execution, differential cryptanalysis and level of security. Recuperado el 18 de mayo de https://www.researchgate.net/publication/323081584_Review_on_improvement_of_advanced_encryption_standard_AES_algorithm_based_on_time_execution_differential_cryptanalysis_and_level_of_security

3. El servidor HTTP número uno en Internet; https://httpd.apache.org/; Accedido el 1 de mayo de 2022.

4. ¿Qué es PHP?; https://php.net/manual/es/intro-whatis.php; Accedido el 1 de mayo de 2022.

5. Mariadb sitio oficial https://mariadb.org/learn/; Accedido el 1 de mayo de 2022.

6. Acerca de Moodle: https://docs.moodle.org/all/es/Acerca_de_Moodle; Accedido el 4 de mayo de 2022.

7. Moodle soporte https://docs.moodle.org/32/en/MySQL_full_unicode_support; Accedido el 4 de mayo de 2022.

8. https://docs.moodle.org/dev/Moodle_and_PHP7; Accedido el 71 de mayo de 2022.

9. MySQL soporte completo de Unicode; https://docs.moodle.org/32/en/MySQL_full_unicode_support; Accedido el 17 de mayo de 2022.

10. Joel Barrios Dueñas; Permisos del sistema de archivos en GNU/Linux; http://www.alcancelibre.org/staticpages/index.php/permisos-sistema-de-archivos; Accedido el 18 de mayo de 2022.

11. Actualización a Moodle; https://docs.moodle.org/all/es/Actualizaci%C3%B3n_de_moodle; Accedido el 18 de mayo de 2022.

Metodología para Controlar Robots con Multiagentes Colaborativos

Yael Atletl Bueno Rojas, María del Carmen Santiago Díaz, Judith Pérez Marcial, Ana Claudia Zenteno Vázquez

Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla
Av. San Claudio y 14 Sur s/n, Cd Universitaria, C.P. 72592 Puebla, Pue.
yael.buenor@alumno.buap.mx, { marycarmen.santiago, judith.perez,
ana.zenteno}@correo.buap.mx

Resumen. El control de robots físicos reales requiere de etapas de diseño y simulación en base al análisis de sus características físicas a fin de brindar al sistema real toda la información necesaria para un control eficiente. En éste trabajo proponemos una herramienta para entrenar agentes de inteligencia artificial con base en aprendizaje reforzado, para que estos puedan controlar un robot dentro de una simulación física, con el propósito de trasladar este control al mundo real.
Palabras claves: Inteligencia artificial, Robótica, Aprendizaje de Máquina, Aprendizaje Reforzado**,** Simulación por Computadora.

1 Introducción

Durante mucho tiempo, la robótica ha sido responsable de realizar actividades repetitivas o peligrosas, y ha mejorado constantemente con el tiempo. Sin embargo, a medida que se han acelerado los requisitos para estas mejoras, los métodos utilizados para controlar los movimientos y comportamientos de estos robots también han tenido que acelerar.

Para enfrentar este desafío, la robótica recurrió a la inteligencia artificial, que se ha ido expandiendo y refinando cada vez más; una de las ramas de la inteligencia artificial, el aprendizaje profundo, tiene técnicas prometedoras que podrían permitir que los robots reaccionen a su entorno. En este trabajo, nos centramos especialmente en el Aprendizaje Reforzado Profundo, que toma principios del condicionamiento clásico para influenciar el comportamiento de un programa, al que llamaremos .ªgente".

Los robots interactúan con el mundo real, en condiciones fuera de su control; tienen que utilizar información incompleta e incierta, esto es, la información que pueden obtener de sus sensores; para conseguir sus objetivos. Estos límites presentan retos importantes en lo que se categoriza como razonamiento, interacción con el mundo físico y aprendizaje. Poder superarlos, es esencial para la seguridad no sólo del robot, sino también de su entorno, tomando en cuenta las potenciales consecuencias de fallar, especialmente en sistemas críticos [25].

El aprendizaje reforzado profundo es una técnica que consigue agentes capaces de obtener comportamientos complejos, a través de recompensas y penalizaciones; este ha

ganado amplia adopción en la creación de agentes capaces de interactuar con una simulación, siendo aplicados en ambientes de alta complejidad, como lo podría ser un videojuego [2, 4]. En publicaciones recientes, esta técnica ha conseguido agentes capaces de llevar a cabo movimientos físicos complejos, incluso en ambientes irregulares [14,18,20].

La mayoría de los algoritmos existentes para esta técnica requieren una amplia muestra de datos y una gran cantidad de memoria, sin embargo, existen algoritmos, como Proximal Policy Optimization (PPO) que logran resultados aceptables con menor requerimiento en tiempo y memoria [24].

Avances recientes han llevado a utilizar algoritmos de aprendizaje reforzado para controlar robots, sin embargo, llevar a cabo el aprendizaje utilizando directamente los robots puede ocasionar desperfectos en ellos, como fallas en los sensores, desgaste de motores, ruptura de piezas entre otros; además de requerir una cantidad de tiempo significativa para preparar el ambiente de entrenamiento y establecer las condiciones iniciales [11]. Esto se ha solucionado utilizando simulaciones para el entrenamiento, lo que permite explorar técnicas y soluciones de forma más eficiente. Sin embargo, requieren la aplicación de técnicas que lleven a una generalización transferible a la realidad, como la aleatorización de dinámicos [19] o la adaptación de dominio [21]; técnicas que permiten que los agentes puedan adaptarse de la invariabilidad de una simulación a la impredecibilidad de la realidad.

La propuesta de este documento es establecer un marco de trabajo para facilitar el entrenamiento de agentes que puedan ser trasladados al mundo real, al proveer herramientas para la aleatorización en la generación de escenarios [19], esquemas de colaboración para múltiples agentes y componentes de software compatibles con múltiples plataformas.

1.1. Trabajos relacionados

En el control a través de inteligencia artificial, se pueden describir varias vertientes, está por ejemplo el control relacionado a la navegación, que debe resolver la búsqueda de caminos hacia el objetivo [23,26]; otro, en el que nos enfocamos en este documento, es el control de más bajo nivel, relacionado con los motores y sensores.

Transferir un modelo entrenado en una simulación a un robot real conlleva muchos desafíos, por los cuales es necesario desarrollar e implementar técnicas que le permitan a estos agentes llevar a cabo sus tareas adaptándose a las irregularidades e impredecibilidades del mundo real. Por mencionar algunos ejemplos, los robots cuadrúpedos entrenados a partir de imitación de movimientos en un ambiente virtual, usando técnicas de adaptación de dominio [21]; o un brazo entrenado con la ayuda de la aleatorización de dinámicos [19]. Los cuales proponen y demuestran técnicas funcionales. Esto puede ser apoyado con técnicas de aprendizaje de habilidades motoras complejas en simulación, como Adversarial Motion Priors [18], que ha demostrado su utilidad en el campo de la robótica [6]; Adversarial Skill Embeddings (ASE) [22], o Motion Parametrization [14] que sustituyen las complejas funciones de recompensa que se

utilizan para evitar que los agentes desarrollen comportamientos motrices no deseados. Estas técnicas permiten entrenar agentes que simulan entidades físicas complejas del mundo real.

Por otro lado, es posible ver ejemplos de agentes que manejan interacciones complejas y múltiples objetivos a través de sus implementaciones en videojuegos, que han sido empleados principalmente para probar las capacidades de los algoritmos y políticas [3,7,10], pero cuya utilidad puede variar desde la automatización de pruebas [4,8], área que ha explorado la división SEED de Electronic Arts; hasta desafiar a los jugadores y personajes no jugadores programados con técnicas tradicionales [5,13].

2 Métodos y Materiales

Para detallar el acercamiento teórico y práctico que se propone en este documento, empezaremos describiendo la arquitectura del agente

Se propone una estructura multinivel, en la que interactúen varios agentes responsables del comportamiento y toma de decisiones, y un agente encargado de la ejecución de tareas motoras. Este procedimiento en sentido abstracto se trata de composición de agentes, en la que un agente toma como entrada los valores producidos por otro y las señales que reciba de ciertos sensores (véase la Fig. 1).

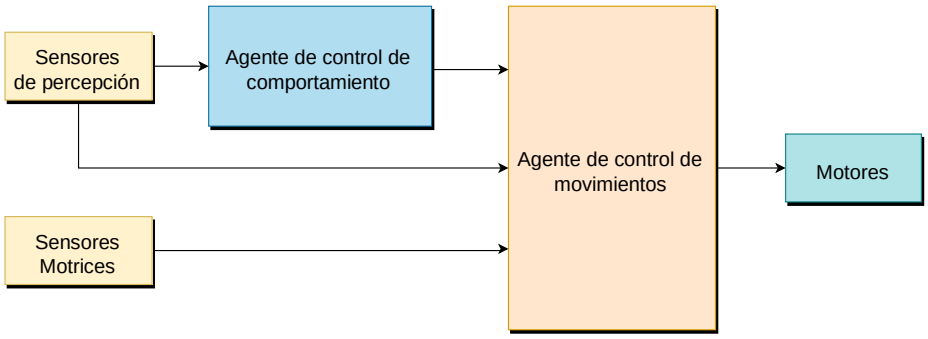


Fig. 1. Interacción entre los diferentes agentes, los sensores y los motores, relacionados por entradas y salidas de información.

A través de un acercamiento modular, se busca combinar los datos que generen múltiples agentes entrenados por separado para conseguir un comportamiento mas complejo que el que presentan individualmente, utilizando un planificador orientado a objetivos como arquitectura de decisión, el cual cederá el control a uno o varios agentes al filtrar las salidas de aquellos desactivados, como si se tratase de estados en una máquina de estados finitos como se puede observar en la figura 2.

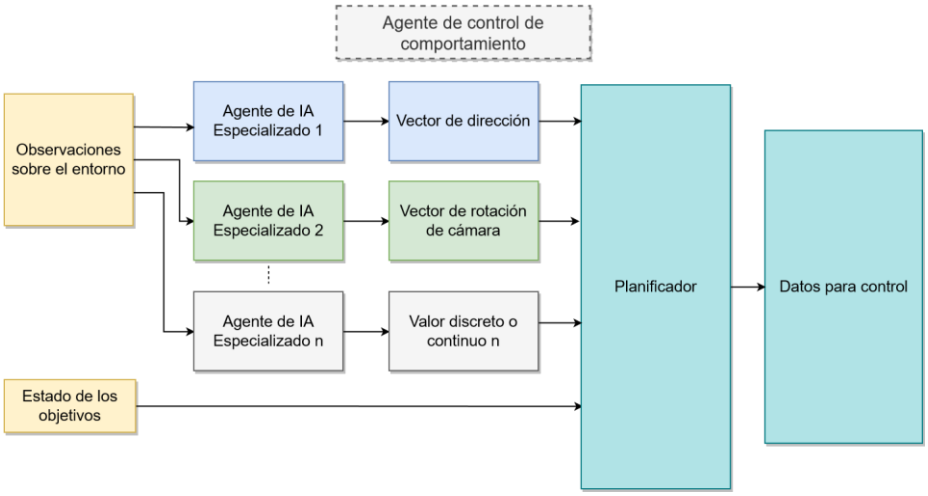


Fig. 2. Esquema que ilustra un ejemplo de la arquitectura propuesta para el agente de control de comportamiento, utilizando tres agentes especializados distintos.

Utilizamos un planificador derivado de STRIPS, Goal Oriented Action Planning, que provee una toma de decisiones dinámica sin requerir el poder de cómputo que los agentes requieren, evaluando únicamente precondiciones y efectos asignados a cada agente [17].

Los agentes que conforman el control de comportamiento utilizan técnicas de aprendizaje reforzado, específicamente el algoritmo Proximal Policy Optimization (PPO), el cual destaca por su facilidad de implementación y ajuste, que además permite el uso de espacios de acción continuos y discretos [2,24].

Cada uno es entrenado en el ambiente de simulación elegido. Se crean múltiples instancias, las cuales contribuyen al entrenamiento del modelo [2]. Cabe notar que las instancias funcionan en paralelo, contribuyendo al entrenamiento del agente al mismo tiempo.

2.1. Acercamiento práctico

Para llevar a cabo el desarrollo del software descrito en este documento, se comienza por elegir el software y librerías. Debido a que el propósito de este proyecto es acercar el uso de agentes de Inteligencia Artificial al publico general, se favorecen programas de código abierto con licencias no restrictivas que únicamente requieren atribución, como lo son MIT, las BSD, zlib o Apache; en vez de licencias que restrinjan la utilización comercial, como la utilizada por NVIDIA al liberar sus proyectos, o como lo son GPL, LGPL y AGPL, que requiere que se libere el código del software derivado, además de que este utilice la misma licencia [16]. Es por esto que se eligió Godot Engine para llevar a cabo

la implementación de este conjunto de herramientas, principalmente por su licencia MIT y su integración del motor de físicas Bullet, el cual ha sido utilizado exitosamente para la simulación robótica y la investigación en inteligencia artificial [9,18,21]; además, emplear un motor de videojuegos tiene otras ventajas, como su manejo de memoria; en este caso, su naturaleza auto-contenida, sin necesidad de dependencias; y la eficacia de utilizar lenguajes de alto nivel [1].

Debido a que no existían integraciones completas de estos algoritmos a este motor, originalmente se conseguía la comunicación entre el motor y otras aplicaciones para llevar acabo el entrenamiento del agente usando un puerto local hacia Python, donde los módulos RayRLlib y godot-rl-agents se encargan de sincronizar los datos de entrenamiento que provienen del motor, para que puedan ser aprendidos por un mismo agente [2]. Un desglose más amplio de esta arquitectura puede verse en la Fig. 3.

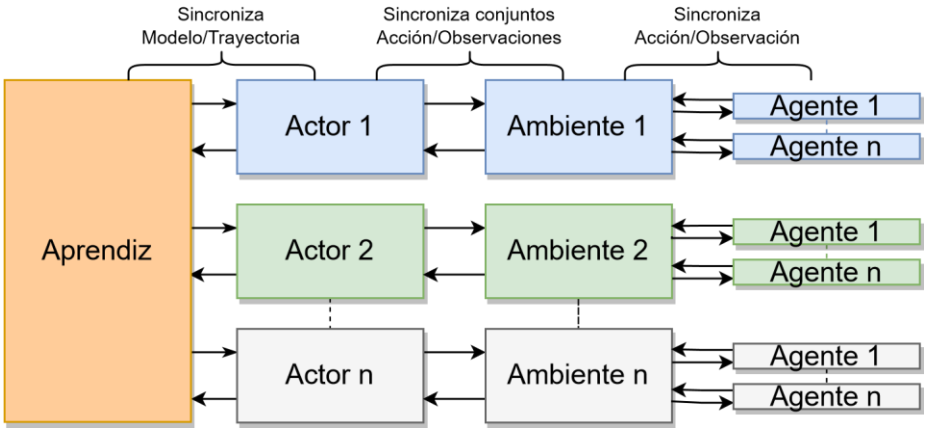


Fig. 3. Adaptación del esquema creado por Edward Beeching [2]: ”La arquitectura de entrenamiento utilizada para la interfaz de Agentes de Godot RL. Los procesos de actores paralelos recopilan trayectorias de varios ejecutables del entorno. Cada entorno contiene múltiples agentes paralelos, cada uno con su propia instanciación del entorno.”

Para superar las limitaciones en este sentido, se modificó el código fuente de esta herramienta para exportar los modelos entrenados como archivos ONNX, un estándar abierto para inteligencia artificial, que son capaces de replicar las salidas que de otra forma se obtendrían en Python. Estos archivos pueden ser integrados al motor elegido utilizando C#, de forma que se elimina las dependencias externas al tiempo de distribución, la desventaja es la incapacidad de entrenar los modelos directamente.

2.2. Agentes

Navegador La primera fase del entrenamiento, consiste en darle al agente un objetivo y la distancia del mismo, además se le proporcionan sensores de distancia en un arreglo de cono, simulado un sensor de profundidad, para verificar que pueda resolver problemas complejos de navegación como lo es el problema del callejón sin salida y los cambios de elevación utilizando rampas. Se disen˜an estos escenarios teniendo en consideración los desafíos más comunes en la navegación de robots móviles, como lo son la búsqueda de caminos, [12,15]. En la sección de trabajo a futuro se discuten más posibles escenarios.
Como conjunto inicial de escenarios de prueba, se eligieron 2 variaciones de cada problema, al correr el entrenamiento, se selecciona al azar uno de los 4 escenarios.

4. Conclusiones

Hasta este momento se han realizado las etapas previas de diseño y se estan cuantificando los rangos de precision a fin de retroalimentar el sistema y generar una versión mejorada y mas certera que nos permita extrapolar esta metodología a otros sistemas con relativa facilidad. En fases posteriores, se incluirán más variaciones para el entrenamiento y se creará un nuevo conjunto de escenarios para realizar evaluaciones para evitar el overfitting. Una propuesta a futuro es aleatorizar la posición de inicio del agente, la ubicación del objetivo, el peso del agente y la velocidad del mismo [19].

Aunque es posible reemplazar PPO dentro del módulo de aprendizaje, aún es necesario exponer en alto nivel este parámetro.

Referencias

1. Bartneck, C., Soucy, M., Fleuret, K., Sandoval, E.B.: The robot engine - makingthe unity 3d game engine work for hri. In: IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN2015). pp. 431 – 437. IEEE (2015). https://doi.org/10.1109/ROMAN.2015.7333561
2. Beeching, E., Dibangoye, J., Simonin, O., Wolf, C.: Godot reinforcement learningagents. arXiv preprint arXiv:2112.03636. (2021)
3. Bellemare, M.G., Naddaf, Y., Veness, J., Bowling, M.: The arcade learning environment: An evaluation platform for general agents. Journal of Artificial Intelligence Research **47**, 253–279 (jun 2013). https://doi.org/10.1613/jair.3912, https://doi.org/10.1613/jair.3912
4. Bergdahl, J., Gordillo, C., Tollmar, K., Gisslén, L.: Augmenting automated game testing with deep reinforcement learning (2021). https://doi.org/10.48550/ARXIV.2103.15819, https://arxiv.org/abs/2103.15819
5. Bogdanovic, M., Markovikj, D., Denil, M., de Freitas, N.: Deep apprenticeshiplearning for playing video games. In: AAAI Workshop: Learning for General Competency in Video Games (2015)

6. Escontrela, A., Peng, X.B., Yu, W., Zhang, T., Iscen, A., Goldberg, K.,Abbeel, P.: Adversarial motion priors make good substitutes for complex reward functions (2022). https://doi.org/10.48550/ARXIV.2203.15103, https://arxiv.org/abs/2203.15103
7. Gold, A.: Academic ai and video games: A case study of incorporating innovativeacademic research into a video game prototype. In: Proceedings of the IEEE 2005 Symposium on Computational Intelligence and Games (CIG'05). Piscataway, NJ: IEEE (2005), http://nn.cs.utexas.edu/?gold:cig05
8. Gordillo, C., Bergdahl, J., Tollmar, K., Gisslén, L.: Improving playtesting coveragevia curiosity driven reinforcement learning agents. CoRR **abs/2103.13798** (2021), https://arxiv.org/abs/2103.13798
9. Heiden, E., Millard, D., Coumans, E., Sheng, Y., Sukhatme, G.S.: NeuralSim: Augmenting differentiable simulators with neural networks. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2021), https://github.com/google-research/tiny-differentiable-simulator
10. Ibarra, I.A., Ramos, B., Roemheld, L.: Angrier birds: Bayesian reinforcement learning. CoRR **abs/1601.01297** (2016), http://arxiv.org/abs/1601.01297
11. Ibarz, J., Tan, J., Finn, C., Kalakrishnan, M., Pastor, P., Levine, S.: How to train your robot with deep reinforcement learning: lessons we have learned. The International Journal of Robotics Research **40**(4-5), 698–721 (jan 2021). https://doi.org/10.1177/0278364920987859, https://doi.org/10.1177/0278364920987859
12. Kurzer, K.: Path Planning in Unstructured Environments: A Real-time Hybrid A* Implementation for Fast and Deterministic Path Generation for the KTH Research Concept Vehicle. Ph.D. thesis (12 2016). https://doi.org/10.13140/RG.2.2.10091.49444
13. Lample, G., Chaplot, D.S.: Playing fps games with deep reinforcement learning.In: AAAI Conference on Artificial Intelligence (2017)
14. Lee, S., Lee, S., Lee, Y., Lee, J.: Learning a family of motor skills from a singlemotion clip. ACM Trans. Graph. **40**(4) (2021)
15. Lu, D.: Eight degrees of difficulty for autonomous navigation (Dec 2020),https://picknik.ai/ros/navigation/2020/12/04/navigation.html
16. Morin, A., Urban, J., Sliz, P.: A quick guide to software licensing for the scientist-programmer. PLOS Computational Biology **8**(7), 1–7 (07 2012). https://doi.org/10.1371/journal.pcbi.1002598, https://doi.org/10.1371/journal.pcbi.1002598
17. Orkin, J.: Symbolic representation of game world state: Toward real-time planningin games (2004)
18. Peng, X.B., Abbeel, P., Levine, S., van de Panne, M.: Deepmimic: Exampleguided deep reinforcement learning of physics-based character skills. ACM Trans. Graph. **37**(4), 143:1–143:14 (Jul 2018). https://doi.org/10.1145/3197517.3201311, http://doi.acm.org/10.1145/3197517.3201311
19. Peng, X.B., Andrychowicz, M., Zaremba, W., Abbeel, P.: Sim-toreal transfer of robotic control with dynamics randomization. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE (may 2018). https://doi.org/10.1109/icra.2018.8460528, https://doi.org/10.1109/icra.2018.8460528

20. Peng, X.B., Berseth, G., van de Panne, M.: Terrain-adaptive locomotion skills using deep reinforcement learning. ACM Trans. Graph. **35**(4) (jul 2016). https://doi.org/10.1145/2897824.2925881, https://doi.org/10.1145/2897824.2925881
21. Peng, X.B., Coumans, E., Zhang, T., Lee, T.W.E., Tan, J., Levine, S.: Learning agile robotic locomotion skills by imitating animals (07 2020). https://doi.org/10.15607/RSS.2020.XVI.064
22. Peng, X.B., Guo, Y., Halper, L., Levine, S., Fidler, S.: Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters. ACM Trans. Graph. **41**(4) (Jul 2022)
23. Pfeiffer, M., Shukla, S., Turchetta, M., Cadena, C., Krause, A., Siegwart, R., Nieto,J.I.: Reinforced imitation: Sample efficient deep reinforcement learning for map-less navigation by leveraging prior demonstrations. CoRR **abs/1805.07095** (2018), http://arxiv.org/abs/1805.07095
24. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms (2017). https://doi.org/10.48550/ARXIV.1707.06347, https://arxiv.org/abs/1707.06347
25. Su¨nderhauf, N., Brock, O., Scheirer, W., Hadsell, R., Fox, D., Leitner, J., Upcroft,B., Abbeel, P., Burgard, W., Milford, M., Corke, P.: The limits and potentials of deep learning for robotics (2018). https://doi.org/10.48550/ARXIV.1804.06557, https://arxiv.org/abs/1804.06557
26. Zhu, K., Zhang, T.: Deep reinforcement learning based mobile robot navigation: A review. Tsinghua Science and Technology **26**(5), 674–691 (2021). https://doi.org/10.26599/TST.2021.9010012

Dinámica de Sistemas y Medio Ambiente

Gladys Linares-Fleites, María de Lourdes Sandoval-Solis,
Rossana Schiaffini-Ponte, Luis Ignacio Juárez-Ruanova

Posgrado en Ciencias Ambientales, Benemérita Universidad Autónoma de Puebla
{gladys.linares, maria.sandoval}@correo.buap.mx,
rossana.schiaffiniaponte@viep.com.mx, bioluis1@hotmail.com

Resumen. Los estudios ambientales requieren cada vez más del uso de softwares para esclarecer las dinámicas que se encuentran relacionadas en el sistema ambiente-sociedad. Desde la perspectiva del pensamiento sistémico, estas investigaciones están íntimamente relacionadas al lenguaje de la Dinámica de Sistemas. Las problemáticas ambientales requieren de la utilización de nuevos marcos teóricos y de softwares específicos para comprender las relaciones existentes entre las políticas públicas y las carencias de servicios de agua potable, drenaje, recolección de basura y salud. El objetivo de este trabajo es aplicar este nuevo marco conceptual al estudio de las carencias de servicios y de las políticas públicas que ha enfrentado el municipio de Atoyatempan, en el estado de Puebla, México.
Palabras Clave: Análisis de Coincidencias, Grafos, Estudios Ambientales.

1 Introducción

El estudio del medio ambiente, desde la perspectiva del pensamiento sistémico, está vinculado estrechamente al lenguaje de la Dinámica de Sistemas. Esto se debe al reconocimiento de la interrelación entre la naturaleza y el hombre.
El estado del arte de la disciplina Dinámica de Sistema puede apreciarse a través de las publicaciones [1], [2], [3] y [4], que muestran el desarrollo de la misma.
La disciplina Dinámica de Sistemas, se plantea como "una forma o un paradigma de pensamiento que se expresa a través de cierto sistema de convenciones, es decir, a través de un lenguaje particular" [1].
En la actualidad existen herramientas computacionales que permiten apoyar el proceso de modelado y simulación con Dinámica de Sistemas [2]. Estas herramientas han permitido el uso y la difusión de la Dinámica de Sistemas en diversos sectores, entre los que se destacan los estudios ambientales.
En las ciencias ambientales, donde se interrelacionan las ciencias naturales y las ciencias sociales, debe asumirse un papel importante en el mundo emergente del *big data*. Hay que comprender el enorme potencial de los grandes conjuntos de datos construidos mediante las interacciones sociales y las interacciones de la naturaleza, que aportan técnicas para mejorar el análisis de los datos y sus interpretaciones ilustrativas. [3].

El objetivo de este trabajo es aplicar este nuevo marco conceptual en el estudio de las estructuras de los principales problemas de carencias de servicios y de políticas públicas que ha enfrentado el municipio de Atoyatempan, Puebla, México.

2 Métodos

Los sistemas computacionales han desarrollado entornos de softwares para el modelado con la Dinámica de Sistemas. En particular, nos referiremos a la propuesta denominada Análisis Reticular de Coincidencias [4].

El Análisis Reticular de Coincidencias es un conjunto de técnicas que persiguen detectar y representar qué sucesos tienden a aparecer al mismo tiempo en ciertos espacios delimitados. Estos N espacios delimitados (i) se denominan *escenarios* y pueden considerarse unidades de análisis (*registros*). En cada uno de estos *escenarios* o *campos* un conjunto de J *sucesos* (x_{ij}) pueden estar presentes (denotándose por 1) o ausentes (denotándose por 0).

Un conjunto de escenarios forman una matriz binaria de incidencias (X) con dimensiones (N × J). Estos escenarios pueden agruparse en H subconjuntos para poderlos comparar.

A partir de las matrices de incidencias y ocurrencias pueden obtenerse las matrices de coincidencias. A continuación se establece la definición de *coincidencia*.

Definición. Dos sucesos (j y k) reciben la calificación de coincidentes si ocurren conjuntamente en el mismo escenario i.

Las coincidencias pueden ser medidas, empleando para ello las medidas de proximidad binaria. Estas medidas poseen un valor máximo de 1 cuando hay total coincidencia entre dos sucesos dicotómicos y 0 cuando hay total independencia entre ellos.

Se han ideado diferentes gráficos para estudiar coincidencias, pero de todos ellos el que más información puede ofrecer es el *grafo de coincidencias*, que consiste en dibujar todos los sucesos que interesan u ocurren en los escenarios como nodos o vértices vinculados entre sí por aristas, siempre y cuando sean coincidentes. Estos grafos pueden experimentar importantes variaciones con el fin de mejorar la representación de los fenómenos estudiados

Se han diseñado una gran variedad de programas de computación para representar las coincidencias [5]. En lenguaje R se han desarrollado tres programas gratuitos con los que cualquier usuario podría aplicar este tipo de análisis. Ellos son: **coin, netcoin y webcoin.** Para este estudio se ha elegido el software **netcoin** [6].

Netcoin es una librería escrita en R que permite al usuario generar matrices de coincidencias con sus correspondientes grafos y crear páginas web interactivas a partir de ellas. En dichas páginas interactivas pueden cambiarse un gran conjunto de elementos de los grafos, así como, generar tablas y gráficos descargables. Entre los elementos modificables, entre otros, destacan los siguientes:

a) La etiqueta, el tamaño, el color y la forma de los sucesos o nodos, en función de sus propiedades.
b) La etiqueta, el grosor y el color de las aristas que representan las coincidencias entre los sucesos, en función de las propiedades de los vínculos (frecuencias, grado de coincidencia, significación).

Con los nodos seleccionados se forman dos tipos de tablas de atributos: la de los sucesos y la de coincidencias.

3 Resultados y Discusión

El municipio de Atoyatempan, cuya localización se muestra en la figura 1, ha cambiado en su estructura paisajista en las tres últimas décadas.

El rumbo de su historia agrícola está marcado principalmente por sus actividades agropecuarias y comerciales [7].

En la actualidad, los principales problemas que enfrenta este municipio son: la falta de planificación para el crecimiento urbano y el crecimiento en la demanda del agua para uso doméstico y la agricultura. Por otra parte, un manejo inadecuado de los suelos podría afectar su desarrollo.

También, el crecimiento poblacional, las políticas públicas y las actividades de producción y de comercio, están influyendo en los procesos de transformación de las coberturas vegetales.

Para ayudar a prevenir la disminución de coberturas de vegetación nativas, deben revisarse los planes de desarrollo municipal y estatal, junto con las políticas de gestión ambiental.

Con el propósito de alcanzar el objetivo planteado, se realizaron entrevistas a pobladores del municipio Atoyatempan para valorar su percepción sobre la carencia de servicios públicos y esclarecer su relación con las políticas públicas.

Se visitaron 62 viviendas del total de las 1 947 viviendas existentes. Se aplicó una encuesta por vivienda, seleccionándose personas mayores de edad por cada vivienda. De las preguntas realizadas a los encuestados, sólo se analizan las preguntas 8 **(P8)** y 9 **(P9)** a través de un análisis de coincidencias.

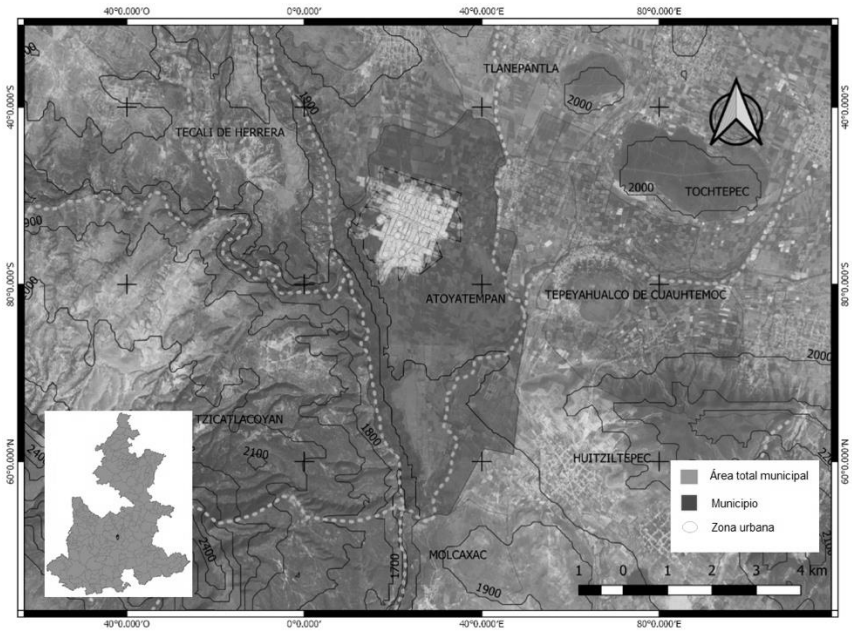


Fig. 1. Municipio de Atoyatempan, Puebla, México.
Localización según información del INEGI 2018. Elaboración propia.

A continuación se relacionan dichas preguntas y las respuestas dadas por los entrevistados; cada pregunta es clasificadas en cuatro clases:
P8. ¿Tiene carencia de servicios públicos como: agua potable, drenaje, recolección de basura y salud?
Respuestas: A) Los cuatro B) Tres C) Dos o uno D) Ninguno
P9. ¿Cómo considera las políticas públicas para el desarrollo del municipio?
Respuestas: A) Idóneas B) Suficientes C) Escasas D) Insuficientes
Los resultados obtenidos, utilizando el paquete MINITAB 17 [8], se resumen en la tabla 1.

Tabla 1. Frecuencias de las respuestas a las preguntas 8 y 9.
(Filas: P8, Columnas: P9)

	A	**B**	**C**	**D**	Total
A	0	0	0	1	1
B	0	0	0	4	4
C	0	1	13	1	15
D	14	18	7	3	42
Total	14	19	20	9	**62**

Puede apreciarse que en el 67.74 % de las viviendas, se plantea ninguna carencia de servicios; en el 24.19% de las viviendas se plantea uno o dos carencias de servicios; en el 6.45% de las viviendas se plantea carencia de tres servicios y en el 1.61% se plantea carecer de los cuatro servicios.

También se aprecia que con respecto a las políticas públicas del municipio, 14 viviendas consideran que son idóneas (22.58%), 19 consideran que son suficientes (30.64%), 20 consideran que son escasas (32.25%) y 9 consideran que son insuficientes (14.51%).

Para esclarecer la existencia de relaciones entre la percepción sobre las carencia de servicios públicos (**P8**) y cómo se consideran las políticas implementadas (**P9**), se utilizó el paquete **netcoin** [6].

Las tabla 2a, 2b y 3 muestran algunos de los resultados que pueden obtenerse con el commando *barCoin* de dicho paquete.

Tabla 2a. Coincidencias en red.

Nodes(8)

Nombre	**incidencias**
P8.D (No tiene ninguna carencia en los servicios)	42
P9.C (Las policas son escasas)	20
P9.B (Las politicas son suficientes)	19
P8.C (Carece de tres servicios)	15
P9.A (Las políticas son idóneas)	14
P9.D (Las políticas son insuficienes)	9

Tabla 2b. Coincidencias en red.

Links(9)

Origen	**Destino**	**coincidencias**	**valor esperado**
P8.D	P9.C	7	13.548387
P8.D	P9.B	18	12.870968
P8.D	P9.A	14	9.483871
P8.D	P9.D	3	6.096774
P9.C	P8.C	13	4.838710
P9.B	P8.C	1	4.596774

Puede apreciarse la coincidencia entre la no existencia de carencias de servicios público y la percepción de politicas sufucuenbtes.

Obsérvese que en la tabla 3 se brindan intervalos de confianza que permiten hacer inferencias a la totalidad de viviendas del municiío, con un nivel de confianza del 95%.

Estos resultados concuerdan con los obtenidos en Juárez *et al*., 2021, a través de un procedimiento de prueba de hipótesis

Tabla 3. Coincidencias en red e intervalos de confianza.

Nodes(8):

Nombre	incidencias
P8.D	67.74194
P9.C	32.25806
P9.B	30.64516
P8.C	24.19355
P9.A	22.58065
P9.D	14.51613

Links(9):

Origen	Destino	coincidencias	Valor esperado	Lim.Inf	Lim Sup.
P8.D	P9.C	11.290323	21.852237	5.7408934	16.839752
P8.D	P9.B	29.032258	20.759625	23.5593305	34.505186
P8.D	P9.A	22.580645	15.296566	17.6170848	27.544206
P8.D	P9.D	4.838710	9.833507	0.6568654	9.020554
P9.C	P8.C	20.967742	7.804370	15.8837690	26.051715
P9.B	P8.C	1.612903	7.414152	0.0000000	6.626791

En la figura 2 se puede observar la representación visual de estas relaciones.

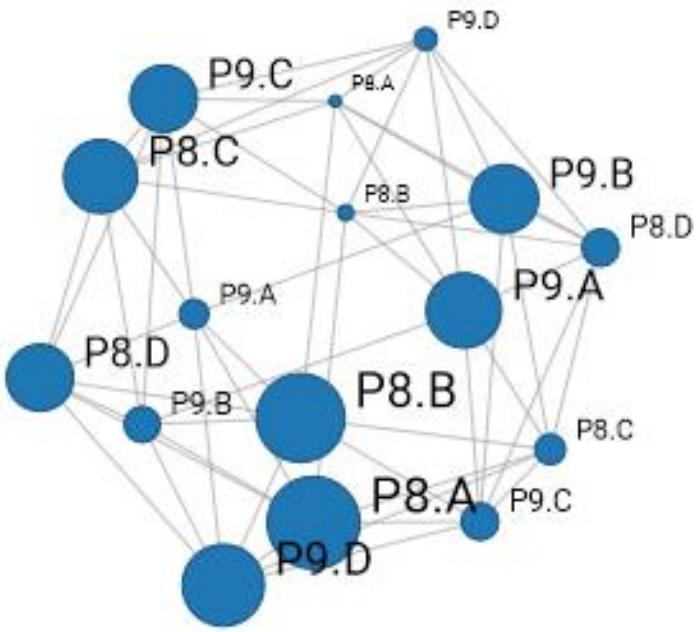


Fig. 2. Gráfico que expresa las coincidencias entre la percepción de las carencias en servicios públicos y de las políticas públicas.

3. Conclusiones

Como se expresó anteriormente, el *análisis reticular de coincidencias* tiene como finalidad descubrir las pautas de concurrencia de una serie de sucesos en varios escenarios, dado que se persigue esclarecer cómo se distribuyen conjuntamente esas características en las distintas unidades, en las que pueden o no estar presente. En el gráfico pueden apreciarse relaciones entre carencias de servicios y políticas públicas, en base a criterios expresados por los pobladores y sus necesidades y también los mecanismos de progreso municipal, sosteniendo también una visión integral del fenómeno bajo estudio.

La toma de decisiones para el desarrollo del municipio de Atoyatempan se debe dirigir, fundamentalmente, a: fortalecer a la población vulnerable, fomentar el sentido de preservación natural y transformación social a través de políticas públicas adecuadas, e identificar los territorios más propicios para los asentamientos humanos y la capacidad ecológica del área. Debe destacarse que lograr incentivar instancias de participación del

público en la definición de actividades y objetivos, contribuirá al desarrollo del municipio.

La difusión del paradigma dinámico-sistémico y el desarrollo de propuestas útiles a este fin es un reto para los especialistas en Ingeniería de Software, constituyéndose en una tarea estratégica para la comunidad comprometida con este reto.

Referencias

1. Andrade, H., Dyner, I., Espinosa, A., Lopez, H.,& Sotaquira, R.: Pensamiento Sistémico: Diversidad en búsqueda de Unidad. División de Publicaciones UIS. 423 p (2001)
2. Andrade Sosa H.H.; Lince Mercado E.; Hernández Cuadrado A.E. & Monsalve Quintero A.J.:Evolución: herramienta software para modelado y simulación con dinámica de sistemas Revista de Dinámica de Sistemas Vol. 4 Núm. 1 (2010)
3. Escobar M . "Studying Coincidences with Network Analysis and Other Multivariate Tools." The Stata Journal, 15(4), 1118–1156. (2015)
4. Escobar M. & Tejero C: El análisis reticular de coincidencias EMPIRIA. Revista de Metodología de Ciencias Sociales. No 39 , pp. 103- 128. (2018)
5. Escobar M. & Martinez-Uribe L.: Network Coincidence Analysis: The netCoin R Package Luis *Journal of Statistical Software*, Volume 93, Issue 11 (2020)
6. Escobar M.; Barrios D.; Prieto C.; Martinez-Uribe L. & Cabrera-Álvarez R. Interactive Analytic Networks Paquete R Version 2.0.20 (2021)
7. Juárez Ruanova L.I.; Linares Fleites G.; Sandoval Solís M.L &. Cigarroa Alonso K.M.: Cambio de uso de suelo y vegetación asociado a la carencia de servicios públicos y políticas públicas en Atoyatempan, Puebla. Nexo Revista Científica Vol. 34 Núm. 06, pp 1611-1622/ (2021)
8. Minitab Statistical Software Version 17 gratuita. (2020)
9. R Core Team: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.(2020)

Análisis de Variables Implícitas para Determinar el Autismo

Jorge Martínez Vargas, María del Carmen Santiago Díaz, Gustavo Trinidad Rubín Linares, Yeiny Romero Hernández

Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla
Av. San Claudio y 14 Sur s/n, Cd Universitaria, C.P. 72592 Puebla, Pue.
jorge.martinezv@alumno.buap.mx, {marycarmen.santiago, gustavo.rubin, yeiny.romero}@correo.buap.mx

Resumen. El presente trabajo de investigación tiene como objetivo apoyar en el diagnóstico del Trastorno del Espectro Autista (TEA) a partir de ciertas características que identifican a este sector de la población, la investigación tiene cierto grado de dificultad ya que estas características varían para cada individuo por este motivo este campo se mantiene en continuo estudio. Este trabajo implementa un programa para determinar a partir de ciertas variables si una persona padece de autismo.
Palabras Clave: Trastorno del Espectro Autista (TEA), Machine learning, Ciencia de datos, Q-CHAT.

1 Introducción

1.1 Antecedentes históricos

Las primeras descripciones, consolidadas como relevantes, sobre lo que actualmente denominamos trastornos del espectro autista (TEA) corresponden a las publicaciones de Leo Kanner (1943) y Hans Asperger (1944).

La incorporación del término autismo al significado actual se debe a Leo Kanner, tras la aparición en 1943 del que se puede distinguir como el artículo fundacional del autismo actual: *"Autistic disturbances of affective contact"*. En los años siguientes a la publicación del citado artículo, Kanner siguió profundizando en la delimitación del trastorno, al cual le asignó la denominación de *"autismo infantil precoz"*, tras haber acumulado experiencia mediante la identificación personal de más de 100 niños y haber estudiado muchos otros procedentes de colegas psiquiatras y pediatras.

Hans Asperger, en 1944, publicó observaciones muy similares a las de Kanner. La

publicación de Asperger recogía la historia de cuatro muchachos, y al igual que Kanner, utilizaba el término autismo (psicopatía autista), coincidencia asombrosa si se tiene en cuenta, como parece ser, que Asperger desconocía el trabajo y la publicación de Kanner, y viceversa. Los pacientes identificados por Asperger mostraban un patrón de conducta caracterizado por: falta de empatía, ingenuidad, poca habilidad para hacer amigos, lenguaje pedante o repetitivo, pobre comunicación no verbal, interés desmesurado por ciertos temas y torpeza motora y mala coordinación.

No fue hasta 1980, con la publicación del DSM-III, cuando se incorporó el autismo como categoría diagnóstica específica. Se contemplaba como una entidad única, denominada "autismo infantil".

El DSM III-R, aparecido en 1987, supuso una modificación radical, no solo de los criterios sino también de la denominación. Se sustituyó autismo infantil por trastorno autista. Con ello el autismo quedaba incorporado a la condición de "trastorno" *(disorder)*, término que se usa en los manuales para definir genéricamente los problemas mentales, marcando una distancia conceptual con la terminología propia de los problemas médicos de etiología y fisiopatología conocida total o parcialmente.

En los años 1994 y 2000 aparecieron respectivamente el DSM-IV y el DSM IV-TR, que, aunque no planteaban modificaciones sustanciales entre ellos, representaron un nuevo cambio radical. Por una parte, se definieron 5 categorías de autismo: trastorno autista, trastorno de Asperger, trastorno de Rett, trastorno desintegrativo infantil y trastorno generalizado del desarrollo no especificado. Además, se incorporó el término trastornos generalizados del desarrollo *(pervasive developmental disorders)*, como denominación genérica para englobar los subtipos de autismo.

El DSM 5 va a consolidar conceptualmente el autismo, sustituyendo la denominación actual de trastornos generalizados del desarrollo por la de Trastorno del Espectro Autista (TEA).

Los trastornos del espectro autista (TEA) son un grupo de afecciones diversas. Se caracterizan por algún grado de dificultad en la interacción social y la comunicación. Otras características que presentan son patrones atípicos de actividad y comportamiento; por ejemplo, dificultad para pasar de una actividad a otra, gran atención a los detalles y reacciones poco habituales a las sensaciones.

En este proyecto se busca determinar un modelo de aprendizaje computacional que sea capaz de usar respuestas de un instrumento de evaluación para el autismo de pacientes para poder identificar si este padece o no un Trastorno del Espectro Autista. Para lograr esto, habrá que obtener las respuestas del cuestionario al igual que ciertos datos del paciente. Posteriormente, debemos elegir el modelo de aprendizaje computacional que se utilizara, y entrenarlo para poder realizar la determinación del autismo.

Además, se aplicará un proceso de validación y evaluación del modelo utilizando los conjuntos de datos de entrenamiento, validación y test. Una vez que tengamos el modelo capaz de determinar si un paciente tiene espectro autista o no, se procederá a integrar el modelo obtenido en una herramienta que permita el uso de este modelo por parte del usuario, que podrá usarlo con datos nuevos para obtener una determinación de este trastorno.

1.2 Estado del arte

El trastorno del espectro del autismo es una categoría de trastornos que se caracteriza por patrones de conducta restrictiva, repetitiva y estereotípica, con alteraciones de la interacción y comunicación social. El trastorno tiene una amplia variedad de expresiones clínicas, gravedad y nivel de función. El aumento de la prevalencia en las últimas décadas puede deberse a una mejor detección y más temprana. La prevalencia actual es de cerca de uno en cada 88 niños. Se desconoce la causa, pero no hay una relación con las inmunizaciones de la infancia. En general, la mayoría de los niños con trastorno del espectro autista no tienen trastornos médicos relacionados y sus necesidades médicas son similares a las de los niños de su misma edad con desarrollo normal.

La detección del autismo en México es muy importante ya que la falta de reconocimiento de este trastorno tiene costos muy elevados para las familias y los prestadores de servicios de salud y educación. Muy a menudo el diagnóstico de autismo se realiza cuatro o cinco años después de que los padres observan los primeros síntomas. Las razones para este reconocimiento tardío son diversas; pero una de las principales es la falta de identificación de síntomas clave que obliguen a una evaluación diagnóstica en forma, además en nuestro país son poco conocidos los instrumentos de tamizaje y diagnóstico por parte de los profesionistas primarios como maestros y médicos familiares, quienes son los primeros en escuchar las quejas y preocupaciones de los padres. Aun en contextos más especializados, estas herramientas son poco conocidas pues su adquisición y aplicación es un proceso complejo y costoso que a menudo debe realizar el profesionista por su cuenta. A pesar de estos inconvenientes, en años recientes se han realizado grandes esfuerzos para el reconocimiento del autismo puesto que hay evidencias de que las intervenciones tempranas mejoran el pronóstico en estos niños. En la última década se han realizado avances muy importantes en el diseño de instrumentos de diagnóstico y tamizaje, a los que se han utilizado con propósitos de investigación clínica y epidemiológica. En algunos países su uso se ha vuelto una rutina en las escuelas y se ha logrado una mayor detección de autismo por lo que se han elevado las tasas de prevalencia. Los instrumentos son muy diversos, pueden ser listas de autoinforme dirigidos a los padres para que registren los síntomas de los niños, o cédulas de observación para ser completadas por el clínico o el personal entrenado para tal propósito.

Obtener un diagnóstico de autismo puede llevar mucho tiempo, porque el autismo es diferente entre las personas, pero también porque depende de la forma en que se diagnostica. Esto es especialmente importante en países más pobres o en el caso de personas pobres que viven en países más ricos que tienen grupos significativos de comunidades desfavorecidas. Adaptamos una versión de 10 ítems del cuestionario Q-CHAT-25 para su uso en el programa de controles de salud de rutina en Chile y reclutamos a 287 participantes menores de tres años divididos en tres grupos: Controles (125), Retraso en el desarrollo (149) y Condición del espectro autista (13). Nuestros resultados muestran que un breve cuestionario para la detección del autismo se puede aplicar con éxito en un programa de control de salud en entornos de escasos recursos. Nuestros resultados muestran que nuestro cuestionario tuvo un buen desempeño general,

no diferente a su versión más larga, el Q-CHAT-25. Nuestro cuestionario fue específico para el autismo, con buena sensibilidad y confiabilidad, y es adecuado para usarse en un entorno de detección. Este estudio proporciona evidencia de que la implementación de programas de detección de condiciones del espectro autista utilizando el Q-CHAT-10 proporciona valor por dinero y mejora el diagnóstico de la condición del espectro autista en aquellos que participan en programas de control de salud de rutina en países en desarrollo o áreas pobres de países ricos.

En las últimas dos décadas, se desarrollaron varios instrumentos de detección para detectar niños pequeños que pueden ser autistas tanto en muestras clínicas como no seleccionadas. Entre otros, la Lista de verificación cuantitativa para el autismo en niños pequeños (Q-CHAT) es una medida cuantitativa y distribuida normalmente de los rasgos autistas que demuestra buenas propiedades psicométricas en diferentes entornos y culturas. Recientemente, el aprendizaje automático (ML) se ha aplicado a la ciencia del comportamiento para mejorar el rendimiento de clasificación de las herramientas de detección y diagnóstico del autismo, pero principalmente en niños, adolescentes y adultos. En este estudio, usamos ML para investigar la precisión y confiabilidad del Q-CHAT en la discriminación de niños pequeños autistas de aquellos que no lo tienen. Se aplicaron cinco algoritmos ML diferentes (bosque aleatorio (RF), bayesiano ingenuo (NB), máquina de vectores de soporte (SVM), regresión logística (LR) y vecinos más cercanos K (KNN)) para investigar el conjunto completo de Q-CHAT elementos. Nuestros resultados mostraron que ML logró una precisión general del 90%, y SVM fue el más efectivo, pudiendo clasificar el autismo con un 95% de precisión. Además, utilizando el enfoque de eliminación de características recursivas (RFE) de SVM, seleccionamos un subconjunto de 14 elementos que garantizan una precisión del 91 %, mientras que se obtuvo una precisión del 83 % de los 3 elementos con mejor discriminación en común con el nuestro y el Q-CHAT informado anteriormente. Esta evidencia confirma el alto rendimiento y la validez transcultural del Q-CHAT, y respalda la aplicación de ML para crear versiones más cortas y rápidas del instrumento, manteniendo una alta precisión de clasificación, para ser utilizado como una herramienta rápida, fácil y de alto nivel.

1.3 Base de datos

Nombre del conjunto de datos: Datos de detección del trastorno del espectro autista para niños pequeños
Fecha: 22 de julio de 2018.
Autor: Dr. Fadi Thabtah
Fuente: Fayez Thabtah
Departamento de Tecnología Digital
Instituto de Tecnología de Manukau,
Auckland, Nueva Zelanda
fadi.fayez @manukau.ac.nz
El conjunto de datos fue desarrollado por el Dr. Fadi Fayez Thabtah (fadifayez.com)

utilizando una aplicación móvil llamada ASDTests (ASDtests.com) para evaluar el autismo en niños pequeños. Este conjunto de datos se puede usar para análisis descriptivos y predictivos, como clasificación, agrupación, regresión, etc. Puede usarlos para estimar el poder predictivo de las técnicas de aprendizaje automático para detectar rasgos autistas.

El rápido crecimiento en el número de casos de TEA en todo el mundo requiere conjuntos de datos relacionados con los rasgos de comportamiento. En la actualidad, se encuentran disponibles conjuntos de datos de autismo muy limitados asociados con pruebas clínicas o de detección y la mayoría de ellos son de naturaleza genética. Por lo tanto, proponemos un nuevo conjunto de datos relacionado con la detección del autismo en niños pequeños que contenía características influyentes que se utilizarán para un análisis posterior, especialmente para determinar los rasgos autistas y mejorar la clasificación de los casos de TEA. En este conjunto de datos, registramos diez características de comportamiento (Q-Chat-10) más otras características individuales que han demostrado ser efectivas para detectar los casos de TEA de los controles en la ciencia del comportamiento.

Tipo de dato: Predictivo y Descriptivo: Nominal/categórico, binario y continuo
Tarea: Clasificación
Tipo de atributo: categórico, continuo y binario
Área: Ciencias médicas, de la salud y sociales
Tipo de formato: Sin matriz
¿Su conjunto de datos contiene valores faltantes? No
Número de instancias (registros en su conjunto de datos): 1054
Número de Atributos (campos dentro de cada registro): 18 incluyendo la variable de clase

Tabla 1. En la siguiente tabla se muestra la descripción de cada atributo utilizado en el conjunto de datos.

Rasgos	Tipo	Descripción
A1: Pregunta 1 Respuesta	Binary (0, 1)	Código de respuesta
A2: Pregunta 2 Respuesta	Binary (0, 1)	Código de respuesta
A3: Pregunta 3 Respuesta	Binary (0, 1)	Código de respuesta
A4: Pregunta 4 Respuesta	Binary (0, 1)	Código de respuesta
A5: Pregunta 5 Respuesta	Binary (0, 1)	Código de respuesta
A6: A6: Pregunta 6 Respuesta	Binary (0, 1)	Código de respuesta
A7: Pregunta 7 Respuesta	Binary (0, 1)	Código de respuesta
A8: Pregunta 8 Respuesta	Binary (0, 1)	Código de respuesta
A9: Pregunta 9 Respuesta	Binary (0, 1)	Código de respuesta
A:10 Pregunta 10 Respuesta	Binary (0, 1)	Código de respuesta
Edad	Number	Niños pequeños (meses)

Puntos de Q-chat-10	Number	1-10 (menor o igual a 3 sin rasgos de TEA; > 3 rasgos de TEA)
Sexo	Character	Masculino o femenino
Ethnicity	String	Lista de etnias comunes en formato de texto
Born with jaundice	Boolean (yes or no)	Si el caso nació con ictericia
Familiar con antecedentes de TEA	Boolean (yes or no)	Si algún miembro de la familia inmediata tiene un PDD
Quién está completando la prueba	String	Padre, yo, cuidador, personal médico, médico, etc.
¿Por qué te hacen la prueba?	String	Usar cuadro de texto de entrada
variable de clase	String	Rasgos ASD o Rasgos No ASD (asignados automáticamente por la aplicación ASDTests). (Sí No)

2 Metodología

La base de datos utilizada fue la del Dr. Fadi Fayez Thabtah obtenida de la página kaggle, esta es una subsidiaria de Google LLC.

Esta base de datos consta de 18 atributos incluyendo el valor de clase, una parte está conformada por datos personales del niño tales como (edad, sexo, entre otros) y otra parte se compone por preguntas tomadas del cuestionario Q-CHAT-10.

Se probaron dos algoritmos de aprendizaje, K-means y Random Forest (Bosques Aleatorios), y se realizaron pruebas con ambos algoritmos. El algoritmo de aprendizaje seleccionado fue “Random Forest” ya que tienen una capacidad de generalización muy alta para muchos problemas.

Se utilizaron las librerías de Python no estándar que son parte de la ciencia de datos tales como Numpy, Pandas y Sklearn, además de Tkinter y Pyttsx3, para mejorar la interfaz con el usuario.

Durante el entrenamiento se obtienen los valores del conjunto de datos utilizado y creamos un dataframe con esos valores. Posteriormente, realizamos un pequeño diccionario para realizar una predicción y asignamos la separación de los valores.

El modelo está siendo implementado en un programa donde el usuario ingresa la información de preguntas del cuestionario Q-chat-10, nombre, sexo, edad, entre otros y,

al terminar este formulario se hará la evaluación del posible autismo, como se muestra en la captura de pantalla de la figura 1.

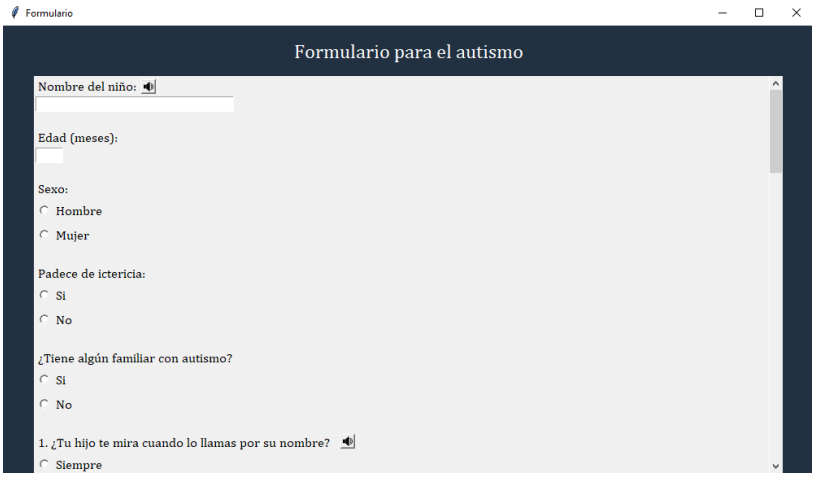


Fig. 1. Interfaz gráfica del usuario.

3 Conclusiones

En este trabajo presentamos las bases para llevar a cabo la generación y posterior análisis de la información relacionada con el TEA, de tal forma que aunque se seleccionó una base de datos de dominio público, es importante conocer los antecedentes históricos de su generación ya que el trastorno en sí tiene una naturaleza compleja que requiere de una profunda revisión antes de llevar a cabo cualqquier implementación, análisis y diagnóstico que la información procesada genere.

En una siguiente etapa se validarán los diagnósticos con un especialista a fin de permitir una retroalimentación del sistema.

Referencias

1. O'Mahony L (2018). El niño con necesidades especiales de atención de la salud. Tintinalli J.E., & Stapczynski J, & Ma O, & Yealy D.M., & Meckler G.D., & Cline D.M.(Eds.), *Tintinalli. Medicina de urgencias, 8e*. McGraw Hill. https://accessmedicina.bibliotecabuap.elogim.com/content.aspx?bookid=2329&sectionid=202927810
2. Albores-Gallo, L., Hernández-Guzmán, L., Díaz-Pichardo, J. A., & Cortes-Hernández, B. (2008). Dificultades en la evaluación y diagnóstico del autismo. Una discusión. Salud Mental, 31(1), 37–44.

https://ebsco.bibliotecabuap.elogim.com/login.aspx?direct=true&db=lth&AN=31808249&lang=es&site=ehost-live.

3. Roman-Urrestarazu, A., Yáñez, C., López-Garí, C., Elgueta, C., Allison, C., Brayne, C., Troncoso, M., & Baron-Cohen, S. (2021). Autism screening and conditional cash transfers in Chile: Using the Quantitative Checklist (Q-CHAT) for early autism detection in a low resource setting. Autism : The International Journal of Research and Practice, 25(4), 932–945. https://doi.org/10.1177/1362361320972277
4. Tartarisco, G., Cicceri, G., Di Pietro, D., Leonardi, E., Aiello, S., Marino, F., Chiarotti, F., Gagliano, A., Arduino, G. M., Apicella, F., Muratori, F., Bruneo, D., Allison, C., Cohen, S. B., Vagni, D., Pioggia, G., & Ruta, L. (2021). Use of Machine Learning to Investigate the Quantitative Checklist for Autism in Toddlers (Q-CHAT) towards Early Autism Screening. Diagnostics (Basel, Switzerland), 11(3). https://doi.org/10.3390/diagnostics11030574

Aplicaciones de las Ciencias Computacionales en Sistemas Inteligentes y Ciberseguridad
se terminó de editar en Diciembre de 2022 en la Facultad de Ciencias de la Computación
Av. San Claudio y 14 Sur
Jardines de San Manuel
Ciudad Universitaria
C.P. 72570

Aplicaciones de las Ciencias Computacionales en Sistemas Inteligentes y Ciberseguridad

Coordinado por

María del Carmen Santiago Díaz