

REVECOM

REVISTA VENEZOLANA DE COMPUTACIÓN

ML • Java • Python • C# • C++ • Scala
Perl • Haskell • Lisp • JavaScript



Sociedad Venezolana de Computación

Vol. 9, No. 1
Junio 2025



ISSN: 2244-7040



REVECOM

Revista Venezolana de Computación

Sociedad Venezolana
de Computación

Editores:

Egluibet Vásquez, Wilmer Pereira, Jesús Pérez

ISSN: 2244-7040



Vol. 9, No. 1
Junio 2025

Editorial

La Empresa Digital y su Transformación

Uno de los sectores de la sociedad digital y su economía que mayores transformaciones ha experimentado en las últimas décadas, es el empresarial; sector en el cual el desarrollo tecnológico avanza con tal impulso que obliga a las empresas a transformarse continuamente. En este editorial, analizamos las propiedades que caracterizan a una empresa digital y veremos su relación con el conocido proceso de transformación digital.

Una empresa digital es un tipo de organización que emplea las tecnologías digitales de un modo holístico y estratégico, transformado así la forma en que capta, sirve y retiene a sus consumidores, opera sus procesos de negocio internos y lleva a cabo sus servicios de negocio. Podemos definir a la empresa digital como aquella que utiliza este tipo de tecnología para: 1) mejorar la experiencia del cliente o consumidor; 2) transformar su modelo de negocios, esto es, la manera en que ella opera y sirve sus clientes; 3) optimizar sus procesos internos tanto medulares como de gestión y soporte; 4) crear mayor valor para sus socios y clientes; 5) ampliar su conectividad y presencia en Internet y 6) explorar nuevas oportunidades de negocio en un mercado global y altamente digitalizado.

Las empresas digitales se caracterizan por lo siguiente:

Están centradas en el cliente o consumidor: La empresa digital busca que la experiencia del cliente sea cada vez más personalizada y satisfactoria, desde el comienzo de la relación hasta su fin. Para ello, se requiere conocer al cliente, anticipar sus necesidades y determinar sus requisitos particulares.

Son ágiles y flexibles: Tienen la capacidad de adaptarse rápidamente a los cambios del mercado y a las nuevas demandas de los consumidores. Su estructura organizacional es plana, flexible y se modifica con facilidad para asegurar la competitividad, agilidad y capacidad de respuesta a las demandas del entorno.

Promueven una cultura del dato: La toma de decisiones se apoya en el uso intensivo de datos y la aplicación de herramientas de analítica de datos e inteligencia artificial generativa que facilitan, a los cuadros gerenciales de la empresa, el diseño de estrategias y la elaboración de los planes respectivos. Los datos y las tecnologías encargadas de su procesamiento son considerados activos organizacionales que se administran de manera análoga a otros activos como el capital financiero o el capital humano. Los datos y las tecnologías digitales se utilizan de modo generalizado en todos los procesos de la organización.

Están interconectadas: Utilizan plataformas computacionales, redes de datos, y herramientas de software que facilitan tanto la comunicación interna entre sus departamentos, como la externa con sus clientes, consumidores y proveedores.

Automatizan sus procesos de negocio: Emplean tecnologías que automatizan tareas repetitivas y mejoran la eficiencia operativa. Los agentes inteligentes y la automatización de procesos robóticos (*RPA - Robotic Process Automation*), por ejemplo, son dos tipos de tecnologías que coadyuvan a la automatización de procesos. Aquellos procesos de carácter repetitivo, monótono o de alto riesgo se automatizan para liberar a los empleados de la rutina o de daños colaterales y reemplazar, así, aquellas actividades que comprometen la salud e integridad física o emocional de los empleados.

Emplean múltiples canales de comunicación: La comunicación con sus clientes se realiza, primordialmente, por medio de canales en línea que incluyen: correo electrónico, portales de autoservicio, redes sociales, comercio electrónico, centro de atención al cliente y asistentes virtuales (*chatbots*), entre otros.

Editorial

Innovan continuamente: La empresa digital promueve una cultura de experimentación, innovación y búsqueda constante de nuevos productos, servicios y procesos. La innovación es una actividad premiada e impulsada por la alta gerencia no solo para crear nuevos productos o servicios, sino para mejorar los procesos internos, el modelo de negocio y la experiencia del cliente.

Estas características varían de un tipo de empresa a otro. Las empresas de producción, por ejemplo, enfatizan la digitalización de los procesos operativos de la planta de producción y su integración con los procesos gerenciales; mientras que, las empresas de servicios se concentran en los procesos de prestación de servicios y su integración con los procesos de atención al cliente. De acuerdo a su origen, las empresas digitales pueden ser de tres tipos:

Nativas digitales: Son empresas que nacen digitalmente y cuyo modelo de negocio se basa sustancialmente en la tecnología. Empresas como Amazon, Netflix y Spotify son representativas de esta categoría.

Empresas tradicionales digitalizadas: Son empresas que han transformado sus procesos y modelos de negocio mediante la adopción de tecnologías digitales. La mayoría de los bancos tradicionales, las empresas de manufactura y de servicios que ofrecen sus servicios en línea pertenecen a esta categoría.

Empresas de plataforma: Son empresas que crean y gestionan plataformas digitales que conectan o prestan servicios de plataforma e infraestructura computacional y de comunicaciones en la nube. Google, Amazon Web Services, Salesforce, Microsoft y OpenAI son ejemplos de esta categoría. Ellas ofrecen servicios a otras empresas para facilitar sus procesos de transformación digital.

Aquellas empresas que han alcanzado un alto grado de madurez digital se distinguen por considerar a la tecnología digital como un activo estratégico. Este activo es administrado directamente por la alta gerencia o por los cuadros ejecutivos encargados de definir los objetivos que la empresa desea alcanzar y las estrategias que debe implementar para que su empresa se transforme digitalmente. La transformación digital (TD) es, por consiguiente, un proceso de gestión estratégica y cambio organizacional para el diseño y creación de empresas digitales desde cero o para la transformación de una empresa tradicional en una digital. Como puede verse en esta definición, la TD, más que un proceso meramente tecnológico, es un proceso de carácter gerencial que involucra a todos niveles ejecutivos de una empresa, especialmente a la alta gerencia.

La TD de una empresa persigue varios objetivos: 1) optimizar la eficiencia operativa de la empresa por medio de la mejora, integración y automatización de sus procesos, tanto gerenciales como operativos; 2) mejorar la experiencia del cliente mediante la personalización de la relación y la disposición de nuevos canales digitales de interacción; 3) fomentar la innovación de nuevos productos, servicios y procesos internos fundamentados en modelos de negocio digitales; 4) aumentar la adaptabilidad y agilidad organizacional para responder rápidamente a los cambios que impone el mercado y 5) aprovechar nuevas oportunidades de negocio.

La TD es un proceso complejo de carácter interdisciplinario, en el cual participan expertos de diferentes disciplinas. ¿Qué papel podemos jugar los computistas e ingenieros de computación, informática y sistemas en este proceso? Nuestro papel en la TD es fundamental ya que podemos aportar nuestro conocimiento, experiencia y dominio de tecnologías digitales complejas como: la computación en la nube, la inteligencia artificial, la cadena de bloques, el Internet de las cosas y muchas otras más. Sin nuestra participación, es imposible llevar adelante iniciativas de TD en cualquier empresa.

Dr. Jonás A. Montilva C.
Profesor de la Universidad de Los Andes

Revista Venezolana de Computación

ReVeCom (Revista Venezolana de Computación) es la primera revista venezolana arbitrada, periódica, digital, orienta a la publicación de resultados de investigación en el campo de la computación. ReVeCom fue creada por la SVC (Sociedad Venezolana de Computación) y tiene entre sus objetivos hacer conocer los trabajos de alta calidad investigativa que se realizan a nivel nacional, latinoamericano e internacional. La revista permite la divulgación de artículos con aporte original en castellano o inglés.

ReVeCom es una revista abierta para una mayor difusión de los resultados de investigación. Cuenta con una página web (<https://svc.net.ve/revecom-informacion>), donde se encuentran los trabajos publicados e información sobre la revista. La revista promueve la pluralidad de intereses, dando cabida a la divulgación de trabajos de todos los campos del conocimiento inherentes a la computación. En marzo de 2024, ReVeCom entró en el directorio de Latindex.

Este volumen de ReVeCom (Vol. 9, No. 1) corresponde a un artículo de investigación que fue seleccionado a través de un riguroso arbitraje por expertos del área nombrados en la siguiente tabla. Además, se encuentran los dos mejores trabajos presentados en CoNCISa 2024 (Conferencia Nacional de Computación, Informática y Sistemas).

Nombre	Afiliación
Junior Altamiranda	Universidad de Los Andes – Venezuela
Jorge Cordero	Universidad Técnica Particular de Loja – Ecuador
Eric Gamess	Jacksonville State University – USA
Alexandra González	Universidad Técnica Particular de Loja – Ecuador
Yosly Hernández	Universidad Central de Venezuela – Venezuela
Marxjhony Jerez	MIU City University Miami – USA
Haydemar Nuñez	Universidad Central de Venezuela – Venezuela
Dinarle Ortega	Universidad de Católica Andrés Bello – Venezuela
Wilmer Pereira	Instituto Tecnológico Autónomo de México – México
Jesús Pérez	Universidad de Los Andes – Venezuela
Eduard Puerto	Universidad Francisco de Paula Santander – Colombia
Wladimir Rodríguez	Universidad de Los Andes – Venezuela
Manuel Sánchez	Universidad Nacional Experimental del Táchira – Venezuela

Directorio de la Sociedad Venezolana de Computación

Presidente Interino:

Dr. Eric Gamess (Jacksonville State University – USA)

Secretario:

MSc. Alejandro Amaro (Colegio Universitario de Caracas – Venezuela)

Tesorero:

Dr. Leonid Tineo (Universidad Simón Bolívar – Venezuela)

Coordinadora de Educación e Investigación:

Dra. Dinarle Ortega (Universidad Católica Andrés Bello – Venezuela)

Coordinadora de Publicaciones:

Dr. Jesús Pérez (Universidad de Los Andes – Venezuela)

Coordinadora de Eventos:

Dra. Yosly Hernández (Universidad Central de Venezuela – Venezuela)

Edición

Comité Editorial

Directora:

Ing. Egluibet Vásquez (Instituto Universitario de Tecnología Industrial “Rodolfo Loero Arismendi” – Venezuela)
Campos de investigación: Alfabetización tecnológica, software educativo, telecomunicaciones.

Coordinador del Comité Editorial:

Dr. Wilmer Pereira (Instituto Tecnológico Autónomo de México – México)
Campos de investigación: Inteligencia artificial, robótica autónoma, aprendizaje automatizado.

Jefe de Redacción:

Dr. Jesús Pérez (Universidad de Los Andes – Venezuela)
Campos de investigación: Pensamiento computacional, enseñanza de la programación, interacción humano-robot, aplicaciones de la inteligencia artificial.

Tabla de Contenido

Editorial	ii
Revista Venezolana de Computación	iv
Directorio de la Sociedad Venezolana de Computación	v
Comité Editorial	vi
Tabla de Contenido	vii
1. Comparación de Metaheurísticas para Determinar la Estructura Óptima de una Red Neuronal para Simular el Consumo Eléctrico en Salta-Argentina Franco Zaneck	1-9
2. Análisis de Sentimientos Migratorios en Publicaciones de Venezolanos en Medios Sociales Livia Borjas, Mauricio Morales, Miguel Zamora	10-19
3. Customizing Software Development Methods: A Process Model Approach Judith Barrios, Jonás Montilva	20-27
Índice de Autores	28

Comparación de Metaheurísticas para Determinar la Estructura Óptima de una Red Neuronal para Simular el Consumo Eléctrico en Salta-Argentina

Franco Zaneck

zaneckfranco@gmail.com

Departamento de Informática, Universidad Nacional de Salta, Salta, Argentina

Resumen: El suministro eléctrico representa un pilar fundamental en la sociedad contemporánea, donde la demanda varía constantemente, requiriendo una oferta estable. Esto implica una coordinación meticulosa en la producción, distribución, y desarrollo de infraestructuras que soporten esta situación. La capacidad de prever con precisión la demanda es crucial para guiar estas decisiones. Aunque las redes neuronales han demostrado eficacia en este ámbito, encontrar la estructura óptima de capas ocultas y neuronas para abordar este desafío sigue siendo una tarea compleja. Por lo tanto, este estudio se centra en investigar dos heurísticas para optimizar la configuración de una red neuronal capaz de prever el consumo eléctrico; en particular, este estudio se centra en el consumo eléctrico de la Ciudad de Salta, Argentina. Este enfoque no solo busca mejorar la precisión de las estimaciones, sino también aumentar la eficiencia del sistema eléctrico en su conjunto. El objetivo último es contribuir al desarrollo sostenible y a la estabilidad energética, garantizando un suministro confiable y equitativo para todos los ciudadanos. Al desarrollar y comparar los resultados brindados por la heurística de Algoritmo Genético y Colonia de Abejas, se pretende encontrar un equilibrio entre la precisión de las predicciones y la eficiencia computacional, lo que permitirá una mejor planificación y gestión del suministro eléctrico. Este enfoque no solo beneficiará a los consumidores finales al garantizar un suministro estable y justo, sino que también proporcionará a los responsables de la toma de decisiones información valiosa para optimizar la infraestructura y los recursos disponibles. En última instancia, este estudio busca no solo mejorar la capacidad predictiva de las redes neuronales en el contexto específico de la Ciudad de Salta, Argentina, sino también sentar las bases para futuras investigaciones y aplicaciones en el campo de la gestión energética y la planificación del suministro eléctrico a nivel global.

Palabras Clave: Redes Neuronales; Consumo Eléctrico; Algoritmo Genético; Colonia de Abejas; Optimización.

Abstract: Electric supply represents a fundamental pillar in contemporary society, where demand constantly varies, requiring a stable supply. This entails meticulous coordination in the production, distribution, and development of infrastructure to support this situation. The ability to accurately forecast demand is crucial for guiding these decisions. Although neural networks have shown effectiveness in this area, finding the optimal structure of hidden layers and neurons to address this challenge remains a complex task. Therefore, this study focuses on investigating two heuristics to optimize the configuration of a neural network capable of predicting electrical consumption; specifically, this study focuses on electrical consumption in the City of Salta, Argentina. This approach not only aims to improve the accuracy of estimates but also to increase the efficiency of the electrical system as a whole. The ultimate goal is to contribute to sustainable development and energy stability, ensuring a reliable and equitable supply for all citizens. By developing and comparing the results provided by the Genetic Algorithm and Bee Colony heuristic, the aim is to strike a balance between prediction accuracy and computational efficiency, enabling better planning and management of electrical supply. This approach will not only benefit end consumers by ensuring a stable and fair supply but will also provide decision-makers with valuable information to optimize infrastructure and available resources. Ultimately, this study aims not only to enhance the predictive capacity of neural networks in the specific context of the City of Salta, Argentina, but also to lay the groundwork for future research and applications in the field of energy management and electrical supply planning globally.

Keywords: Neural Networks; Electrical Consumption; Genetic Algorithm; Bee Colony; Optimization.

I. INTRODUCCIÓN

El suministro eléctrico es un pilar fundamental en el funcionamiento de la sociedad moderna. Desde la iluminación

hasta la alimentación de dispositivos electrónicos esenciales, la electricidad se ha convertido en una necesidad básica en nuestra vida diaria. La dependencia de la electricidad es tan significativa que cualquier interrupción en el suministro puede

tener repercusiones graves en múltiples aspectos de la sociedad, desde la economía hasta la salud pública.

En el contexto actual, caracterizado por un aumento constante en el uso de la tecnología y la digitalización de numerosos procesos, la demanda de energía eléctrica está en constante crecimiento. Esta creciente demanda plantea desafíos significativos para los proveedores de energía, quienes deben garantizar un suministro estable y confiable en todo momento, a pesar de las fluctuaciones en el consumo.

Además, la naturaleza misma de la electricidad, que no puede almacenarse fácilmente en grandes cantidades, agrega una capa adicional de complejidad al sistema eléctrico. Esto significa que la oferta de electricidad debe estar en equilibrio con la demanda en todo momento, lo que requiere una coordinación cuidadosa entre la producción, la distribución y el consumo de energía.

En este contexto, la capacidad de predecir con precisión la demanda eléctrica se vuelve crucial. Una estimación precisa de la demanda no solo permite a los proveedores de energía planificar y gestionar mejor sus recursos, sino que también contribuye a garantizar un suministro continuo y eficiente de electricidad para los consumidores.

Es en este punto donde entran en juego las redes neuronales artificiales (RNN). Estas herramientas de inteligencia artificial han demostrado ser altamente efectivas en una variedad de aplicaciones, incluida la predicción de la demanda eléctrica. Al utilizar datos históricos de consumo, patrones climáticos y otras variables relevantes, las RNN pueden generar modelos predictivos precisos que ayudan a anticipar las necesidades futuras de energía y a tomar decisiones informadas sobre la infraestructura y la inversión en el sector eléctrico.

Por lo tanto, para abordar este desafío, diversos autores [1-6] han desarrollado una variedad de modelos de predicción de la demanda eléctrica a corto plazo en diferentes ubicaciones alrededor del mundo. Estos modelos incorporan una amplia gama de variables de entrada, incluidos datos climáticos específicos de las regiones en cuestión, así como registros históricos detallados del consumo eléctrico. Esta inclusión de múltiples factores permite una aproximación más precisa a la predicción de la demanda.

Al analizar estos trabajos, se puede observar que, en promedio, el valor del error cuadrático medio se sitúa alrededor del 4%. Sin embargo, es importante destacar que este valor puede variar según la complejidad del modelo, la calidad de los datos utilizados y la precisión de los métodos de predicción empleados. Además, se ha observado que la precisión de las predicciones puede verse influenciada por factores externos impredecibles, como cambios repentinos en el clima o eventos inesperados que afecten el comportamiento del consumo eléctrico.

Por otro lado, un grupo de investigadores (referenciados como [7-9]) ha centrado su atención en el desarrollo de modelos basados en redes neuronales para predecir la demanda eléctrica en un horizonte de tiempo más amplio, específicamente, en el mediano plazo. Estos modelos, que han sido aplicados en diversas ubicaciones geográficas alrededor del mundo, comparten similitudes con los modelos mencionados anteriormente en términos de las variables de entrada

consideradas. No obstante, se observa una diferencia significativa en cuanto a los resultados obtenidos.

En contraste con los modelos de predicción a corto plazo, estos modelos neuronales para el mediano plazo muestran un valor del Error Cuadrático Medio que oscila en torno al 7%. Esta discrepancia en la precisión de las predicciones sugiere la existencia de desafíos adicionales al proyectar la demanda eléctrica en un horizonte temporal más extenso.

La complejidad inherente al comportamiento de la demanda eléctrica a mediano plazo puede atribuirse a una serie de factores, como cambios estacionales, tendencias de consumo a largo plazo y la influencia de variables socioeconómicas y políticas. Estos elementos pueden introducir una mayor incertidumbre en los modelos predictivos basados en redes neuronales, lo que se refleja en un aumento del Error Cuadrático Medio.

Además, otro grupo de investigadores (referenciados como [10-16]) se ha dedicado al desarrollo de modelos destinados a prever la demanda eléctrica en un horizonte temporal a largo plazo. Estos modelos representan un esfuerzo por anticipar las necesidades futuras de energía eléctrica en un contexto más amplio y extendido en el tiempo. Al contrario de los modelos de corto y mediano plazo, la predicción a largo plazo implica una consideración más profunda de factores de influencia de largo alcance, que abarcan desde tendencias demográficas hasta indicadores económicos clave.

En este sentido, además de las variables climáticas y los datos históricos de consumo eléctrico que se han utilizado en los modelos previamente mencionados, se incorporan otras variables fundamentales como el tamaño de la población y el producto bruto interno (PBI). Estos factores socioeconómicos ofrecen una perspectiva más completa y detallada sobre las condiciones que influyen en la demanda eléctrica a largo plazo.

Los modelos de predicción a largo plazo se enfrentan a un conjunto único de desafíos y complejidades. Por un lado, deben considerar las tendencias de crecimiento de la población, así como los cambios en los patrones de consumo asociados con el desarrollo económico y tecnológico. Por otro lado, deben tener en cuenta las fluctuaciones a largo plazo en las condiciones climáticas y las políticas energéticas que pueden afectar la demanda eléctrica en el futuro. A pesar de estas dificultades, los modelos desarrollados por estos investigadores han logrado alcanzar un nivel de precisión notable, de acuerdo a lo mencionado en cada uno de los artículos y las conclusiones de los mismos.

Sin embargo, al trabajar con Redes Neuronales Recurrentes (RNN), surge un detalle crucial: el número de capas ocultas y el número de neuronas por capa varían según el problema específico en estudio, sin disponer de una fórmula matemática universal para su cálculo. No obstante, abordar el problema de determinar la estructura óptima de una red neuronal puede conceptualizarse como un problema de optimización combinatoria. En este enfoque, la función objetivo busca minimizar el error entre las predicciones de la red neuronal y los valores reales para cada uno de los patrones de entrada, mientras que las restricciones se centran en garantizar que el número de capas ocultas sea mayor que cero, al igual que el número de neuronas en cada una de estas capas.

El desafío de determinar la estructura óptima de una red neuronal ha sido un tema de interés constante, dado que no existe una fórmula matemática que permita definir de manera exacta cada uno de los elementos involucrados en su configuración. Por esta razón, diversos autores han propuesto diferentes enfoques para encontrar modelos que se adapten a los problemas específicos de cada caso. Un ejemplo de esto se encuentra en los artículos [17-19] donde se emplea el algoritmo genético, una metaheurística, para determinar los parámetros de la red neuronal. En otro enfoque, los autores [20] proponen la optimización de los parámetros de la red neuronal utilizando Metaheurísticas basadas en la inteligencia de enjambre mejorada, que consiste en un método automático para optimizar los hiperparámetros y diseñar estructuras a través de la implementación de algoritmos metaheurísticos mejorados. El objetivo de este artículo es doble. Primero, presentan versiones mejoradas de los algoritmos de crecimiento de árboles y luciérnagas que optimizan las implementaciones originales. En segundo lugar, adaptan estos algoritmos mejorados para la optimización de los hiperparámetros. Inicialmente, las metaheurísticas modificadas se evalúan en funciones de referencia estándar sin restricciones y se comparan con las versiones originales. Luego, los algoritmos mejorados se utilizan para el diseño de la red neuronal.

En este estudio, se desarrollan y comparan dos metaheurísticas, el Algoritmo Genético y la Colonia de Abejas Artificial, con el propósito de optimizar la estructura de una red neuronal orientada a la estimación del consumo eléctrico en la ciudad de Salta, Argentina. La optimización de redes neuronales representa un desafío significativo debido a la alta dimensionalidad y no linealidad del espacio de búsqueda asociado a su configuración. Problemas como la determinación del número óptimo de capas ocultas, la cantidad adecuada de neuronas por capa y las funciones de activación requeridas exigen la exploración de combinaciones de hiperparámetros que maximicen el desempeño del modelo.

En este contexto, se eligieron el algoritmo Genéticos y el algoritmo basado en Colonia de Abejas Artificial por sus ventajas frente a métodos tradicionales y técnicas más recientes basadas en gradiente. Estas metaheurísticas destacan por su capacidad para evitar mínimos locales, una limitación común en métodos como Adam o RMSProp, que dependen de derivadas y funciones continuas y diferenciables. La independencia de estas restricciones hace que el AG y el ABC sean particularmente efectivos en espacios de búsqueda no convexos, una característica frecuente en configuraciones complejas de redes neuronales.

Los métodos de búsqueda exhaustiva, si bien garantizan encontrar la mejor solución, resultan computacionalmente inviables en problemas con muchos hiperparámetros debido al crecimiento exponencial del espacio de búsqueda. Por su parte, técnicas basadas en gradientes, como Grid Search o Random Search, son más rápidas pero menos eficaces en espacios no convexos o con parámetros no diferenciables, como el número de capas ocultas o las funciones de activación. Aunque la optimización bayesiana es moderna y eficiente, su implementación puede ser más compleja y requiere suposiciones específicas sobre la función objetivo, como continuidad o correlaciones entre hiperparámetros, que

no siempre son aplicables en configuraciones de redes neuronales.

Entre las técnicas inspiradas en la naturaleza, como el Enjambre de Partículas (PSO), la Optimización por Búsqueda de Cuckoo (CS) y los Algoritmos de Evolución Diferencial (DE), el AG y el ABC se seleccionaron por su comprobado balance entre exploración y explotación. Además, ambas cuentan con amplia documentación y demostrada eficacia en problemas similares de optimización de redes neuronales.

El uso de AG y ABC no solo permite identificar configuraciones óptimas para redes neuronales aplicadas a la predicción del consumo eléctrico en Salta, sino que también establece un marco adaptable a otros dominios. Este enfoque abre la puerta al desarrollo de técnicas híbridas que combinen lo mejor de ambas estrategias o integren métodos más recientes, como la optimización bayesiana o el PSO, en contextos de metaheurísticas colaborativas.

Además, este enfoque sobresale en el manejo de la complejidad inherente al diseño de redes neuronales, facilitando una exploración eficiente del espacio de soluciones, reduciendo los errores de predicción y maximizando la aplicabilidad práctica de las configuraciones obtenidas. De este modo, sienta un precedente para futuras investigaciones en la optimización de modelos predictivos.

El presente trabajo detalla la metodología adoptada en su desarrollo, organizada de la siguiente manera: en la Sección II, se describe exhaustivamente la metodología empleada, abordando tanto los fundamentos teóricos como los procedimientos prácticos aplicados; en la Sección III, se presentan y analizan los resultados obtenidos con las heurísticas propuestas, destacando las implicaciones de los hallazgos y las comparaciones entre los enfoques; y, finalmente, en la Sección IV, se extraen las conclusiones más significativas del estudio y se identifican posibles líneas de investigación futura, aportando nuevas perspectivas para el desarrollo de técnicas optimizadas en la predicción del consumo eléctrico.

II. METODOLOGÍA DE TRABAJO

En esta sección se describen detalladamente los componentes utilizados en cada heurística, así como los datos y variables empleados en RNN. Este enfoque facilita una comprensión profunda de los procesos implicados en la determinación de la estructura óptima de la red neuronal, permitiendo además una evaluación crítica de la eficacia y eficiencia de las distintas estrategias heurísticas implementadas. Con esta información, se busca optimizar la capacidad predictiva del modelo y respaldar la toma de decisiones en la gestión y planificación del suministro eléctrico.

Es importante señalar que todas las estructuras de red analizadas con cada algoritmo mantendrán el mismo número de neuronas en las capas de entrada y salida. Por lo tanto, estas capas no se consideran al definir la estructura óptima de cada modelo.

A. Algoritmo Genético

El algoritmo genético es una metaheurística inspirada en los procesos de evolución natural. Su funcionamiento se basa en la generación y evolución de una población de soluciones

candidatas, llamadas individuos, representadas generalmente como cadenas (cromosomas) que codifican posibles soluciones a un problema. En cada iteración o generación, se evalúa la calidad de cada individuo mediante una función objetivo (fitness). Los mejores individuos se seleccionan para formar una nueva población mediante operadores genéticos como la **reproducción**, que combina características de dos padres (crossover), y la **mutación**, que introduce pequeñas modificaciones aleatorias para explorar nuevas soluciones. Este proceso iterativo favorece la supervivencia de las soluciones más prometedoras y permite a la población converger hacia una solución óptima o cercana a óptima del problema planteado.

Codificación de Individuos

En este proceso, se ha definido que cada individuo representa una red neuronal, y el objetivo es encontrar la configuración óptima de capas ocultas y neuronas. Para ello, se ha establecido la siguiente codificación: cada individuo consta de tantos genes como capas ocultas tiene la red neuronal en cuestión. Cada gen está compuesto por una secuencia de ceros y unos, que corresponde a un número expresado en el sistema binario. Este número, al ser convertido a su equivalente en base 10, indica la cantidad de neuronas presentes en la capa correspondiente. Se utiliza la codificación binaria porque facilita los procesos de reproducción y mutación, ya que las operaciones de cruce y modificación en los algoritmos evolutivos se manejan de manera eficiente con secuencias binarias

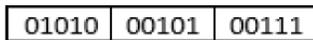


Figura 1: Ejemplo de Codificación de un Individuo

Por ejemplo, considerando la Figura 1, tenemos un individuo con tres genes, esto significa que representa una red neuronal con tres capas ocultas. Al examinar las secuencias binarias de cada gen y convertirlas a base 10, obtenemos valores como 10, 5 y 7, lo que indica que las capas ocultas tienen 10, 5 y 7 neuronas, respectivamente.

Población Inicial

El proceso de generación de la población inicial en el algoritmo genético se divide en dos etapas clave. En la primera, fase de generación inicial, la población de tamaño n se crea aleatoriamente. Luego, en la fase de verificación y reparación, se asegura que los individuos generados sean factibles. Esta fase es crucial, ya que algunos individuos pueden no cumplir con los requisitos necesarios para ser considerados viables. Para garantizar su validez, se implementan dos procesos de reparación: primero, se verifica que no haya capas con cero neuronas intercaladas entre capas que sí tengan neuronas; segundo, se asegura que cada individuo tenga al menos una capa oculta. Estos pasos permiten que la población inicial esté compuesta por individuos que cumplen con los criterios establecidos, listos para el proceso evolutivo del algoritmo.

Proceso de Reproducción

En el proceso de reproducción, se emplea un método de cruzamiento de un punto. En este método, los genes del primer padre se copian hasta un punto de corte aleatorio, y el resto de

los genes se completan con los del segundo padre. Para seleccionar los individuos que serán reproducidos, se utiliza el método de torneo binario. En este proceso, se eligen aleatoriamente dos individuos de la población, se comparan sus valores de aptitud y el individuo con la mejor aptitud es seleccionado como ganador del torneo. Este método se seleccionó por su simplicidad, su capacidad para incluir individuos menos aptos, facilitar la transferencia genética y su eficiencia computacional, ya que solo requiere comparar la aptitud de dos individuos por torneo. Estos enfoques de trabajo están respaldados por la literatura relacionada con el problema, que ha demostrado su eficacia en contextos similares [21-23].

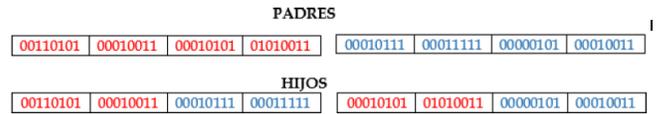


Figura 2: Reproducción de Dos Individuos

La Figura 2 proporciona un ejemplo que detalla el proceso de cruzamiento utilizado en este estudio, ilustrando cómo se combinan los genes de los padres para generar descendencia.

Tras la etapa de reproducción, no se realiza inmediatamente el proceso de verificación y reparación de factibilidad, ya que se espera que la mutación posterior pueda restaurar la factibilidad de las soluciones generadas. Si la mutación no logra restaurar la factibilidad, entonces se aplica el proceso de verificación y reparación, garantizando que todas las soluciones sean viables y cumplan con los requisitos del problema.

Proceso de Mutación

En el proceso de mutación, cada individuo tiene asignada una probabilidad de experimentar una modificación en sus genes. Si se decide que un individuo debe mutar, se selecciona aleatoriamente la cantidad de genes que se modificarán. Para cada gen seleccionado, se altera una cantidad específica de ceros por unos y viceversa, lo que genera variabilidad genética en la población. La Figura 3 ilustra este proceso de mutación.

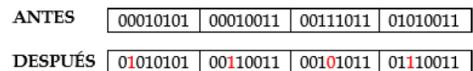


Figura 3: Ejemplo de Aplicación del Operador de Mutación

B. Colonia de Abejas

La metaheurística basada en colonia de abejas (ABC, por sus siglas en inglés) se inspira en el comportamiento colectivo de las abejas en busca de alimento. En este enfoque, se simula una población de abejas que exploran y explotan recursos (soluciones) en un espacio de búsqueda. Estas abejas se dividen en tres grupos principales:

Abejas Empleadas: representan soluciones actuales y están asociadas a fuentes de alimento específicas (posibles soluciones). Cada abeja empleada explora su vecindad para mejorar su fuente actual, evaluando la calidad de la solución mediante una función objetivo.

Abejas Observadoras: estas abejas permanecen en la colmena y eligen fuentes de alimento basándose en la calidad comunicada por las abejas empleadas a través de una “danza”. Las fuentes con mejor calidad tienen mayor probabilidad de ser seleccionadas.

Abejas Exploradoras: si una fuente de alimento no mejora después de un número determinado de iteraciones, las abejas empleadas asociadas a esa fuente la abandonan y se convierten en exploradoras. Estas abejas buscan nuevas fuentes de alimento en regiones aleatorias del espacio de búsqueda, promoviendo la diversidad y evitando el estancamiento en óptimos locales.

El proceso iterativo continúa hasta alcanzar un criterio de parada, como un número máximo de iteraciones o una solución aceptable. Este modelo logra un equilibrio entre exploración (búsqueda de nuevas soluciones) y explotación (mejora de soluciones actuales), permitiendo encontrar soluciones cercanas al óptimo para problemas complejos.

De acuerdo con el procedimiento descrito anteriormente, el primer paso para desarrollar el algoritmo destinado a resolver el problema es definir el tamaño de la población. En este contexto, esto implica determinar el número de fuentes de alimentos, así como la estructura de cada una de estas fuentes.

El siguiente paso es definir las fuentes de alimentos. En este caso, cada fuente de alimento se define como un vector n -dimensional. Cada elemento de este vector representa una posible solución para el problema en cuestión. Similar al caso anterior, cada uno de estos elementos se representa como un vector m -dimensional, donde cada dimensión representa una capa de una posible solución expresada en formato binario.

Este enfoque nos permite generar una variedad de soluciones potenciales dentro de la población inicial, explorando diferentes configuraciones de capas y neuronas en el contexto del problema abordado. La representación en forma de vector facilita la manipulación y la aplicación de operadores genéticos durante el proceso de evolución del algoritmo, lo que contribuye a la eficacia y eficiencia de la búsqueda de soluciones óptimas.

Las abejas empleadas actualizan continuamente las soluciones en función de su experiencia individual y del valor de aptitud de las soluciones recién encontradas. Si encuentran una solución con un costo menor, esta reemplaza a la solución actual. La actualización de la posición para la dimensión j del i -ésimo candidato se calcula utilizando la siguiente ecuación:

$$V_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (1)$$

De esta última ecuación, x_{ij} representa la posición actual en la j -ésima dimensión del i -ésimo candidato, φ_{ij} es un valor aleatorio entre 0 y 1, y x_{kj} es la posición del j -ésimo candidato seleccionado aleatoriamente de la población, y por último $\varphi_{ij}(x_{ij} - x_{kj})$ Representa el tamaño del paso entre dos soluciones determinadas. Esta ecuación refleja el proceso de búsqueda local que realizan las abejas empleadas para mejorar las soluciones existentes.

Además, es crucial considerar el papel de las abejas espectadoras, cuyo objetivo principal es calcular la probabilidad de selección de cada fuente de alimento generada

por las abejas empleadas. Durante esta fase, el número de fuentes de alimento para las abejas espectadoras es el mismo que el empleado por las abejas trabajadoras. Durante esta fase, todas las abejas empleadas comparten información sobre la aptitud de las nuevas fuentes de alimentos con las abejas espectadoras. Luego, cada espectador selecciona la mejor fuente de alimento más apta utilizando la siguiente ecuación:

$$Probabilidad\ Selección = \frac{f_{aptitud}}{\sum f_{aptitud-i}} \quad (2)$$

Donde $f_{aptitud}$ representa la aptitud de la fuente de alimento y $f_{aptitud-i}$ es la aptitud de la fuente de alimento i -ésima. Esta ecuación proporciona una medida de la calidad relativa de cada fuente de alimento, lo que permite a las abejas espectadoras tomar decisiones informadas sobre cuál explorar.

Por último, si la ubicación de una fuente de alimento no se actualiza durante un número predefinido de ciclos, se asume que la fuente de alimento se ha descuidado y se activa la fase de abejas exploradoras. Durante esta fase, la abeja asociada con la fuente de alimento olvidada se convierte en abeja exploradora y la fuente de alimento se reemplaza por otra generada aleatoriamente dentro del espacio de búsqueda. Se define un límite de rechazo como parámetro de control, y cuando este límite se alcanza, las abejas exploradoras generan aleatoriamente una nueva fuente de alimento.

Este proceso se repite hasta que se cumple un número fijo de iteraciones, que sirve como criterio de parada para el algoritmo. Este enfoque garantiza que se explore exhaustivamente el espacio de búsqueda y se encuentre una solución óptima o cercana a la óptima dentro de un número finito de iteraciones.

C. Función Objetivo

La función objetivo desempeña un papel fundamental en ambas heurísticas, ya que se utiliza para seleccionar a los mejores individuos de la población generada. Estos individuos seleccionados se convertirán en los padres de la siguiente generación o se utilizarán para actualizar las fuentes de alimento, según el contexto específico de cada heurística.

El problema que se busca resolver radica en determinar la estructura óptima de la red neuronal. Con este fin, se propone una función objetivo que busca minimizar la discrepancia entre los valores predichos por la red neuronal para un conjunto de datos de entrada y los valores reales correspondientes. En términos matemáticos, esta función objetivo puede definirse como:

$$\sum_{i=1}^N (y_{predicho-i} - y_{real-i})^2 \quad (3)$$

Donde N representa el número total de muestras en el conjunto de datos, $y_{predicho-i}$ denota el valor predicho por la red neuronal para la muestra i , y y_{real-i} representa el valor real correspondiente. La función objetivo busca minimizar la suma de los cuadrados de las diferencias entre los valores predichos y los valores reales para todas las muestras en el conjunto de datos.

Esta función objetivo proporciona una medida cuantitativa de la calidad de la solución generada por el algoritmo,

permitiendo así evaluar y comparar diferentes configuraciones de la red neuronal. El objetivo final es encontrar la estructura de red neuronal que minimice esta función objetivo, lo que se traduce en una capacidad predictiva óptima para el modelo.

D. Variables Involucradas

En esta sección se destacan las variables clave utilizadas para modelar el consumo de electricidad en los sectores residencial, comercial e industrial, subrayando por qué se eligieron ciertas variables como las más relevantes para la red neuronal en estudio. Las variables seleccionadas se agrupan según su contexto y sector, y su importancia se fundamenta en su capacidad para capturar los factores que impactan directamente en el comportamiento energético.

Para el sector residencial contamos con las siguientes variables:

Variables Relacionadas con la Vivienda: Estas incluyen la antigüedad, características y equipamiento de la vivienda, ingresos del hogar, número de ocupantes y la descripción eléctrica de la vivienda (electrodomésticos presentes). Se eligieron porque permiten modelar la vivienda en términos físicos y económicos, lo que a su vez tiene una relación directa con la demanda eléctrica. Estas variables son fundamentales para entender cómo la infraestructura y las condiciones económicas de la vivienda influyen en el consumo de electricidad.

Variables Relacionadas con el Individuo: En esta categoría se incluyen características como edad, género, nivel educativo y situación laboral de los habitantes. La elección de estas variables se basa en la idea de que el comportamiento de los individuos dentro de la vivienda influye considerablemente en su consumo eléctrico, ya que factores como la rutina diaria o el uso de electrodomésticos pueden variar según estos aspectos demográficos.

Variables Socioeconómicas: Consideran el entorno social y económico de la vivienda, lo que es crucial para entender el contexto en el que se toman las decisiones relacionadas con el uso de la electricidad. Factores como el nivel de ingresos o la situación económica del hogar afectan el tipo de tecnología utilizada y el patrón de consumo.

Por otro lado para el sector comercial e industrial, contamos con las variables:

Variables Relacionadas con el Comportamiento del Individuo: En el sector comercial e industrial, se destaca el perfil eléctrico del edificio, que incluye el número de ocupantes, el tipo de equipamiento utilizado, la duración de la jornada laboral y el tipo de actividad económica. Estas variables son esenciales para comprender cómo las actividades de los individuos dentro de los edificios afectan el consumo energético, ya que las necesidades energéticas varían según el tipo de actividad y la intensidad de la jornada.

Variables Relacionadas con el Edificio: Incluyen el tipo de construcción, antigüedad, dimensiones y nivel de aislamiento del edificio. Estas variables fueron seleccionadas porque las características físicas del edificio tienen un impacto directo en la eficiencia energética y el consumo, ya que un edificio bien aislado o de reciente construcción puede requerir menos energía para calefacción o refrigeración.

Variables Relacionadas con el Entorno: Consideran el entorno físico y socioeconómico en el que se encuentra el edificio. Estas variables fueron elegidas para capturar factores externos que pueden influir en el consumo de energía, como el clima local y las condiciones socioeconómicas del área.

Tras un análisis detallado y un estudio de las correlaciones entre estas variables, se determinó que las variables de entrada más representativas para la red neuronal son:

- Día del año (expresado en términos julianos): Esta variable es crucial porque captura los patrones de estacionalidad y variabilidad anual, lo que influye en el consumo energético.
- Hora del día: Es importante porque permite modelar la variabilidad del consumo a lo largo de las distintas horas del día, que está fuertemente influenciada por los hábitos de uso de la electricidad.
- Temperatura horaria: La temperatura tiene un efecto directo sobre el consumo de energía, especialmente en lo que respecta al uso de calefacción o refrigeración.
- Humedad relativa horaria: La humedad también influye en el consumo energético, especialmente en climas cálidos y en el uso de sistemas de control de la humedad.

Estas variables fueron seleccionadas porque proporcionan un conjunto de datos representativo que permite realizar predicciones precisas y útiles para la gestión de la demanda eléctrica, considerando tanto los aspectos temporales como los factores climáticos y de comportamiento. Además, estos datos fueron recopilados a lo largo de un período de seis años, lo que asegura una visión detallada y robusta para modelar el consumo energético.

E. Datos Utilizados

El conjunto de datos utilizado en este trabajo fue recopilado de diversas fuentes confiables: Compañía Administradora del Mercado Mayorista Eléctrico S.A. Argentino (CAMMESA) y Empresa Distribuidora de Electricidad de Salta (EDESA), que aportaron información sobre los consumos históricos de energía eléctrica, e Instituto Nacional de Tecnología Agropecuaria (INTA), que proporcionó datos climáticos. Estos datos, correspondientes a un período de 6 años con resolución horaria, fueron integrados y organizados para construir el conjunto de datos necesario para el estudio.

El objetivo principal fue determinar la estructura óptima del modelo utilizando algoritmo genético y la metaheurística basada en colonia de Abejas artificiales. Una vez definida esta estructura, se procedió al entrenamiento y validación del modelo sin emplear técnicas de cross-validation, ya que el enfoque adoptado buscaba evaluar directamente la capacidad de generalización de la red neuronal. Identificadas las variables de entrada y salida, se realizó una división estratégica del conjunto de datos para asignar los subconjuntos a las distintas etapas del proceso, garantizando así una evaluación objetiva del modelo y facilitando la toma de decisiones informadas sobre los resultados obtenidos.

El porcentaje de selección del 30% de los datos originales surgió como resultado de diversos ensayos realizados, especialmente durante la etapa final de entrenamiento.

Inicialmente, la capacidad de generalización de las redes no era óptima debido a la cantidad insuficiente de datos destinados al entrenamiento. A lo largo de varias ejecuciones del proceso completo –que incluyó desde la selección de la estructura óptima hasta el entrenamiento final de la red para pronosticar la demanda eléctrica– se fue ajustando progresivamente el número de registros utilizados en cada etapa hasta alcanzar una proporción que ofreciera un equilibrio adecuado entre los datos asignados al entrenamiento y a la evaluación.

Finalmente, este 30% de los datos originales se dividió equitativamente en subconjuntos para entrenamiento y evaluación. Esta distribución permitió utilizar un grupo independiente de datos para validar el modelo, lo que ayudó a identificar problemas como el sobreajuste (overfitting) o subajuste (underfitting), y contribuyó a garantizar una evaluación precisa y objetiva del desempeño del modelo.

En conclusión, el ajuste adecuado del número de datos en cada etapa del proceso fue fundamental para mejorar la generalización del modelo. Esta estrategia asegura que el modelo sea capaz de generar pronósticos precisos y robustos, mientras que la validación confiable de su desempeño facilita la toma de decisiones informadas sobre su implementación y ajuste.

III. RESULTADOS OBTENIDOS

El diseño y la implementación de un algoritmo destinado a estimar el consumo eléctrico en la Ciudad de Salta, Argentina, constituye una iniciativa de vital importancia en el contexto actual. Con el continuo avance tecnológico y la creciente relevancia de la electricidad en diversas esferas de la vida diaria, la gestión eficaz de la demanda eléctrica se vuelve una prioridad ineludible. Hasta este momento, la falta de herramientas específicas para abordar este desafío en Salta ha sido evidente, lo que resalta aún más la necesidad de contar con un modelo preciso y adaptado a las particularidades locales.

El fundamento de este modelo radica en el empleo de redes neuronales, las cuales son alimentadas con datos clave como el día del año, la hora del día, la temperatura y la humedad relativa. La ejecución del algoritmo se llevó a cabo en una computadora equipada con un procesador AMD Ryzen 5 3500U de 2.10 GHz y 8 GB de memoria RAM, lo que garantizó un rendimiento óptimo en la resolución del problema. Los tiempos de ejecución de las heurísticas utilizadas, tanto el algoritmo genético como la Colonia de Abejas, se situaron en un intervalo de 3 a 5 minutos, lo que evidencia una eficiencia aceptable en la implementación de las soluciones.

La selección de parámetros para el algoritmo genético se presenta en la Tabla I, mientras que los correspondientes al algoritmo basado en colonias de abejas se detallan en la Tabla II. Estos parámetros fueron cuidadosamente determinados mediante numerosos ensayos. Como punto de partida, se consideraron valores promedio reportados en la literatura, los cuales se ajustaron progresivamente hasta alcanzar los valores óptimos descritos.

Tabla I: Parámetros Utilizados Algoritmo Genético

Parámetro	Valor
Número de generación	50
Tamaño de la población	30
Probabilidad de mutación	10%
Número de reproductores	12

Tabla II: Parámetros Utilizados Colonia de Abejas

Parámetro	Valor
Fuentes de alimento	30
Número de abejas observadoras	12
Número de abejas empleadas	20
Error tolerable	0.05
Número de iteraciones	50

Así mismo, en las Tablas III y IV se presentan los resultados obtenidos por el Algoritmo Genético y el algoritmo basado en Colonia de Abejas, respectivamente. La estructura mostrada en dichas tablas corresponde al número de capas ocultas y al número de neuronas ocultas de cada red neuronal, dado que todas comparten la misma configuración en la capa de entrada y salida. La capa de entrada consta de 4 neuronas, una para cada variable mencionada en la sección Variables Involucradas, mientras que la capa de salida incluye una sola neurona, destinada a la estimación de la demanda eléctrica.

El error cuadrático medio mostrado en ambas tablas corresponde al valor de la función objetivo para cada modelo. Cabe resaltar que los resultados presentados en las Tablas III y IV reflejan los valores más óptimos alcanzados para el mes de marzo, lo que resalta la efectividad de los enfoques empleados durante ese período específico.

Tabla III: Resultados Obtenidos para el Algoritmo Genético para el Mes de Marzo

Estructura de Red Interna	Error Cuadrático Medio
17	1.256
2-5-6	2.498
8	2.351
5-8-6	1.727
4-7	2.101

Tabla IV: Resultados Obtenidos para el Algoritmo Basado en Colonia de Abejas para el Mes de Marzo

Estructura de Red Interna	Error Cuadrático Medio
7	1.19
6-2	3.423
3-4	1.64
1-5-3	2.472
8	2.369

En conclusión, el desarrollo e implementación de este algoritmo representan un avance significativo en la gestión eficiente de la demanda eléctrica en la Ciudad de Salta. Los modelos desarrollados no solo proporcionan predicciones precisas del consumo eléctrico, sino que también establecen

una base sólida para futuras investigaciones y aplicaciones en el ámbito de la energía y la planificación del suministro.

Dada la variabilidad del conjunto de datos utilizados, se decidió desarrollar 12 modelos de redes neuronales, uno para cada mes del año. Los modelos correspondientes a los meses de enero, febrero y abril fueron obtenidos mediante la ejecución del algoritmo genético, mientras que los modelos para el resto de los meses del año surgieron de la aplicación del algoritmo basado en colonias de abejas. Este enfoque permitió aprovechar las fortalezas de ambos métodos para generar redes neuronales que aprendieran de manera óptima las características específicas de cada mes, adaptándose a las condiciones cambiantes de cada período.

Para determinar los conjuntos de datos para entrenamiento, y considerando los doce meses del año y la variabilidad climática que existe a lo largo de cada uno de ellos, se aseguró que hubiera suficientes datos representativos de cada patrón climático en cada conjunto de datos. Esto garantizó que cada modelo tuviera la información necesaria para aprender y adaptarse a las diversas condiciones climáticas presentes en cada mes, evitando sesgos y asegurando una mayor precisión en el desempeño de los modelos. Como resultado, la ejecución de estos modelos finales a lo largo de todo el año arrojó un error cuadrático medio promedio de 2.304, lo que demuestra la efectividad del enfoque adoptado y su potencial para aplicaciones futuras.

IV. CONCLUSIONES

En este estudio, se han comparado y evaluado dos enfoques para la optimización de la estructura de redes neuronales: el algoritmo genético y el algoritmo basado en Colonia de Abejas. Ambos métodos fueron investigados con el objetivo de determinar la configuración óptima de una red neuronal para predecir el consumo eléctrico en la Ciudad de Salta, Argentina. Los resultados obtenidos revelan hallazgos significativos que aportan una comprensión más profunda sobre la eficacia y las aplicaciones potenciales de ambas técnicas.

El análisis de las Tablas III y IV destaca que la mejor estructura de red neuronal encontrada para abordar este problema consta de una sola capa oculta con siete neuronas. Este resultado es notable, ya que demuestra que una arquitectura relativamente sencilla puede capturar la complejidad inherente de los datos y proporcionar predicciones precisas del consumo eléctrico en Salta.

En cuanto a los tiempos de ejecución, se observa que estos oscilan entre 10 y 15 minutos, lo cual refleja una eficiencia computacional aceptable considerando la complejidad del problema y el volumen de datos procesados. Además, el algoritmo basado en Colonia de Abejas tiende a presentar tiempos ligeramente inferiores en comparación con el algoritmo genético, lo que sugiere una ventaja en términos de velocidad de convergencia.

A pesar de estos avances, es importante reconocer ciertas limitaciones del estudio. Por ejemplo, los modelos se desarrollaron bajo un conjunto de variables específicas y un alcance geográfico limitado a la Ciudad de Salta. Esto podría restringir la generalización de los resultados a otras regiones o contextos con características significativamente diferentes.

Asimismo, aunque se evaluaron dos técnicas metaheurísticas, la inclusión de otras metodologías o combinaciones de estas podría ampliar la perspectiva sobre las posibles configuraciones óptimas de redes neuronales.

Adicionalmente, los tiempos de ejecución, aunque aceptables, podrían representar una barrera para aplicaciones en tiempo real o escenarios con datos en constante actualización. Aunque no fue el enfoque principal del estudio, la optimización de los tiempos computacionales mediante técnicas como paralelización, procesamiento distribuido o el uso de hardware especializado (por ejemplo, GPU o TPU) podría ser una mejora importante.

Por último, la interpretación y aplicabilidad de los resultados ofrecen oportunidades valiosas para futuras investigaciones. Si bien los modelos desarrollados han demostrado su capacidad para predecir el consumo eléctrico con precisión, existe un gran potencial para explorar cómo estos resultados pueden integrarse de manera efectiva en sistemas de gestión energética actuales. Esto podría incluir aplicaciones prácticas como la planificación del suministro, el diseño de políticas tarifarias más equitativas o el desarrollo de estrategias de sostenibilidad energética.

Asimismo, un análisis más detallado de la sensibilidad de los modelos frente a escenarios futuros, como cambios climáticos, interrupciones en el suministro o transformaciones en los patrones de demanda, permitiría evaluar su robustez y adaptabilidad. Estos aspectos no solo ampliarían el alcance del estudio, sino que también contribuirían a fortalecer su utilidad en la toma de decisiones informadas en contextos dinámicos y en constante evolución.

Para futuras investigaciones, se recomienda abordar estas limitaciones de forma más concreta. Podría explorarse la incorporación de un conjunto de datos más amplio y diverso, así como variables adicionales relacionadas con factores socioeconómicos, climáticos y de infraestructura. También sería valioso profundizar en el análisis interpretativo de los resultados, lo que facilitaría una mejor comprensión de los factores que influyen en el consumo eléctrico y permitiría diseñar políticas energéticas más efectivas.

Finalmente, el estudio subraya el potencial de las metaheurísticas en la optimización de redes neuronales para problemas complejos como la estimación de la demanda eléctrica. Tanto el algoritmo genético como la Colonia de Abejas han demostrado ser herramientas útiles para este propósito, contribuyendo al avance en la toma de decisiones informadas en la gestión energética y la planificación urbana.

REFERENCIAS

- [1] H. Eskandari, I. Maryam, and P. Mohsen Moghaddam, *Convolutional and Recurrent Neural Network based model for Short-term Load forecasting*, Electric Power Systems Research 195 107173, 2021.
- [2] L. Wu, C. Kong, X. Hao, and W. Chen, *A Short-term Load Forecasting Method based on GRU-CNN Hybrid Neural Network Model*, Mathematical problems in engineering, 1-10, 2020.
- [3] L. Weixuan, D. Wu, and Benoit Boulet, *Spatial-temporal Residential Short-term Load Forecasting via Graph Neural Networks*, IEEE Transactions on Smart Grid 12.6, 5373-5384, 2021.
- [4] J. Lin, J. Ma, J. Zhu, and Y. Cui, *Short-term Load Forecasting Based on LSTM Networks Considering Attention Mechanism*, International Journal of Electrical Power & Energy Systems, 137, 107818, 2022.

- [5] Y. Wang, J. Chen, X. Chen, X. Zeng, Y. Kong, S. Sun, and Y. Liu, *Short-term Load Forecasting for Industrial Customers Based on TCN-LightGBM*, IEEE Transactions on Power Systems, 36(3), 1984-1997, 2020.
- [6] Y. Hong, Y. Zhou, Q. Li, W. Xu, and X. Zheng, *A Deep Learning Method for Short-term Residential Load Forecasting in Smart Grid*, IEEE Access, 8, 55785-55797, 2020.
- [7] B. N. Oreshkin, G. Dudek, P. Pelka, and E. Turkina, *N-BEATS Neural Network for Mid-term Electricity Load Forecasting*, Applied Energy, 293, 116918, 2021.
- [8] M. Askari and F. Keynia, *Mid-term Electricity Load Forecasting by a New Composite Method Based on Optimal Learning MLP Algorithm*, IET Generation, Transmission & Distribution, 14(5), 845-852, 2020.
- [9] S. M. Jung, S. Park, S. W. Jung, and E. Hwang, *Monthly Electric Load Forecasting Using Transfer Learning for Smart Cities*, Sustainability, 12(16), 6364, 2020.
- [10] M. R. Kazemzadeh, A. Amjadian, and T. Amrace, *A Hybrid Data Mining Driven Algorithm for Long Term Electric Peak Load and Energy Demand Forecasting*, Energy, 204, 117948, 2020.
- [11] M. K. M. Shapi, N. A. Ramli, and L. J. Awal, *Energy Consumption Prediction by Using Machine Learning for Smart Building: Case Study in Malaysia*, Developments in the Built Environment, 5, 100037, 2021.
- [12] J. Q. Wang, Y. Du, and J. Wang, *LSTM Based Long-term Energy Consumption Prediction with Periodicity*, Energy, 197, 117197, 2020.
- [13] Y. Liu, C. Gong, L. Yang, and Y. Chen, *DSTP-RNN: A Dual-stage Two-phase Attention-based Recurrent Neural Network for Long-term and Multivariate Time Series Prediction*, Expert Systems with Applications, 143, 113082, 2020.
- [14] M. Ilbeigi, M. Ghomeishi, and A. Dehghanbanadaki, *Prediction and Optimization of Energy Consumption in an Office Building Using Artificial Neural Network and a Genetic Algorithm*, Sustainable Cities and Society, 61, 102325, 2020.
- [15] R. Olu-Ajayi, H. Alaka, I. Sulaimon, F. Sunmola, and S. Ajayi, *Building Energy Consumption Prediction for Residential Buildings Using Deep Learning and other Machine Learning Techniques*, Journal of Building Engineering, 45, 103406, 2022.
- [16] J. Bedi and D. Toshniwal, *Deep Learning Framework to Forecast Electricity Demand*, Applied Energy, 238, 1312-1326, 2019.
- [17] G. Surendra, W. Samar, N. Md Tabrez, and P. Suraiya, *Optimizing Hyperparameters in Neural Networks Using Genetic Algorithms*, 2024.
- [18] S. Oh, J. Yoon, Y. Choi, Y. A. Jung, and J. Kim, *Genetic Algorithm for the Optimization of a Building Power Consumption Prediction Model*, Electronics, 2022.
- [19] A. Narmontas, R. Jankevičiūtė, T. Bliujus, E. Vaičekauskas, and V. Abromavičius, *Exploration of Genetic Algorithm-Driven Hyperparameter Optimization for Convolutional Neural Networks*, IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), 2024.
- [20] N. Bacanin, T. Bezdán, E. Tuba, I. Strumberger, and M. Tuba, *Optimizing Convolutional Neural Network Hyperparameters by Enhanced Swarm Intelligence Metaheuristics*, Algorithms, 2020.
- [21] S. Nikbakht, C. Anitescu, and T. Rabczuk, *Optimizing the Neural Network Hyperparameters Utilizing Genetic Algorithm*, J. Zhejiang Univ. Sci. A 22, 407-426, 2021.
- [22] S. Lee, J. Kim, H. Kang, D. Y. Kang, and J. Park, *Genetic Algorithm Based Deep Learning Neural Network Structure and Hyperparameter Optimization*, Applied Sciences, 11(2), 744, 2021.
- [23] M. Abd Elaziz, A. Dahou, and L. Abualigah, *Advanced Metaheuristic Optimization Techniques in Applications of Deep Neural Networks: A Review*, Neural Comput & Applic 33, 14079-14099, 2021.

Análisis de Sentimientos Migratorios en Publicaciones de Venezolanos en Medios Sociales

Livia Borjas, Mauricio Morales, Miguel Zamora

liborjas@ucab.edu.ve, mauricio1999morales@gmail.com, miguelzamora.23.97@gmail.com

Escuela Ingeniería de Informática, Universidad Católica Andrés Bello, Ciudad Guayana, Venezuela

Resumen: El presente trabajo tuvo como propósito realizar análisis de sentimientos y opinión migratoria en publicaciones de venezolanos en medios sociales, con la finalidad de apoyar el proceso de toma de decisiones en cuanto al fenómeno migratorio que sufre la nación venezolana. En primera instancia se llevó a cabo una revisión documental en aproximadamente ocho (08) artículos relacionados a los modelos de clasificación de textos, lo que permitió reconocer a los algoritmos de clasificación y el aprendizaje automatizado como los métodos más utilizados. En la etapa de desarrollo se definieron dos escalas de categorización, una que reflejó las opiniones de los usuarios y la otra los sentimientos asociados a su posición. Finalmente se procedió al proceso de construcción de los modelos de clasificación mediante la implementación de una aplicación móvil, donde los resultados obtenidos del análisis mostraron tendencias hacia la tristeza como sentimiento predominante y a la postura a favor de la migración, en cuanto a la opinión de los individuos.

Palabras Clave: Análisis de Sentimientos; Minería de Opinión; Clasificación de Textos; Algoritmos de Aprendizaje Automatizado; Fenómeno Migratorio.

Abstract: The purpose of this work was to conduct sentiment analysis and migration opinion analysis in posts by Venezuelans on social media, with the aim of supporting the decision-making process regarding the migratory phenomenon affecting the Venezuelan nation. Initially, a documentary review was carried out on approximately eight (08) articles related to text classification models, which allowed for the recognition of classification algorithms and machine learning as the most commonly used methods. In the development stage, two categorization scales were defined: one reflecting users' opinions and the other representing the feelings associated with their positions. Finally, the process of building classification models was carried out through the implementation of a mobile application, where the results obtained from the analysis showed trends towards sadness as the predominant sentiment and a favorable stance towards migration, regarding individuals' opinions.

Keywords: Sentiment Analysis; Opinion Mining; Text Classification; Machine Learning Algorithms; Migratory Phenomenon.

I. INTRODUCCIÓN

El tema del análisis de sentimientos y minería de opinión ha tomado mucho interés en la actualidad. Muchas investigaciones han demostrado el creciente interés sobre el estudio de los fenómenos migratorios aplicando minería de datos con inteligencia artificial sobre publicaciones en redes sociales [1]. La presente contribución gira en torno al tema de analizar la opinión y sentimiento de los venezolanos expresados en sus redes sociales. El documento expone los primeros resultados de este estudio, organizando el aporte de la siguiente manera: la Sección II tiene el planteamiento del problema, la Sección III describe los antecedentes de la investigación, la Sección IV presenta una propuesta de metodología híbrida para aplicar en el caso en estudio, la Sección V describe los resultados, y finalmente la Sección VI presenta las conclusiones y recomendaciones.

II. PLANTEAMIENTO DEL PROBLEMA

El continuo deterioro de la economía venezolana a lo largo de los años, ha generado escasez severa de alimentos y medicamentos básicos que, sumada a la crisis política e institucional, resultaron en un movido flujo migratorio de venezolanos, reseñándose la diáspora de mayor volumen en la historia venezolana. Según Arias, Arias, Morffe, Martínez y Carreño [2], en su artículo "Informe sobre la movilidad humana venezolana II" mencionan que la migración internacional venezolana es uno de los fenómenos con mayor impacto en Latinoamérica, donde no sólo se ve afectada la población venezolana, sino también los países receptores, ya que, si la cifra de migración mantiene el ritmo actual, creará cada vez más problemas presupuestarios, decaimiento de servicios básicos y la atención sanitaria en dichos países.

En el artículo “Refugiados y migrantes de Venezuela” de la Plataforma de Coordinación Interagencial para Refugiados y Migrantes de Venezuela en su reporte de septiembre del 2021 [3] afirma que, la cifra de venezolanos que han emigrado es de aproximadamente 5,66 millones de individuos, que representan alrededor de un 20% de la población total venezolana.

Para llevar a cabo estudios sobre estos flujos migratorios, el Observatorio Proyecto Migración Venezuela, en su artículo “Percepción en redes sociales sobre la migración venezolana” [4] plantea que el análisis de las redes sociales es una de las herramientas más rápidas y eficaces para obtener una visión general de la situación, ya que proporciona información que posteriormente facilita a los encargados del proceso de toma de decisiones, la materialización de sus objetivos.

Según Hütt [5] en su artículo “Las redes sociales: Una nueva herramienta de difusión” el impacto de las redes sociales tiene la capacidad de cambiar la percepción que los usuarios tienen sobre un tema en particular, y en una situación como la crisis migratoria de Venezuela, las ideas y creencias formadas por la opinión pública pueden entorpecer la integración de los migrantes, por ello es importante aclarar las posturas actuales que se tienen sobre este fenómeno.

Para realizar el análisis de mensajes en redes sociales, una de las técnicas más utilizadas es el Análisis de Sentimientos también conocida como Minería de Opinión, que Liu [6] en su artículo “*Sentiment analysis and opinion mining*” define como “el campo de estudio que analiza las opiniones de las personas, sentimientos, evaluaciones, apreciaciones, actitudes y emociones sobre entidades tales como productos, servicios, organizaciones, individuos, cuestiones, eventos, tópicos y sus atributos”.

La relevancia del análisis de sentimientos radica en su capacidad para utilizar técnicas de procesamiento del lenguaje natural, sobre grandes volúmenes de información de forma automatizada, lo que facilita el reconocimiento de tendencias y patrones para la categorización de las opiniones y los sentimientos de la población.

A. Propuesta

Dada la relevancia del tema migratorio venezolano unido con el aporte que puede sumar el análisis de sentimientos en redes sociales, el objetivo del presente trabajo se enfoca en realizar un proceso de análisis de sentimientos y minería de opinión respecto al tema de migración sobre publicaciones de venezolanos en medios sociales.

Para alcanzar este propósito se aplicaron los siguientes pasos: examinar algunos antecedentes relacionados con el análisis de sentimientos y minería de opinión en redes sociales a fin de establecer la manera efectiva de aplicar este procedimiento en el tema migratorio, determinar los criterios sobre los cuales se llevará a cabo dicho análisis, diseñar modelos descriptivos o predictivos de sentimientos y de opinión que se ajusten a los criterios establecidos, para finalmente implementar dichos modelos mediante una aplicación móvil para su publicación, explotación y uso.

III. ANTECEDENTES

El avance de las tecnologías de big data y minería de textos, en la última década, han demostrado los aportes que pueden obtenerse mediante la aplicación de análisis de sentimientos. A continuación, se describen algunas investigaciones que proporcionan base para el desarrollo del trabajo a realizar.

Al momento de realizar el análisis de sentimientos u opiniones, uno de los mayores retos es el de la comprensión de los tonos lingüísticos en un mensaje, en ese apartado Salaz [7], en su tesis doctoral titulada “Detección de patrones psicolingüísticos para el análisis de lenguaje subjetivo en español”, propuso un método para la detección de patrones psicolingüísticos en el análisis de sentimientos y la detección de la sátira en español.

Este método permite, a través de un enfoque automatizado supervisado, clasificar textos como positivo, negativo, neutro, muy positivo o muy negativo y como satíricos y no satíricos. Esta investigación demuestra las aplicaciones de los enfoques supervisados para reconocer aspectos particulares, en este caso el tono lingüístico en los mensajes.

Por su parte, Cestari [8] realizó una tesis de pregrado titulada “Propuesta para automatizar la asociación de emociones a textos en español”, la cual consistía en automatizar la asociación de frases en español a emociones, mediante el análisis de sentimientos. Tomando en cuenta el modelo de emociones de Ekman, se construyó un prototipo capaz de asociar las frases a un conjunto de seis (6) emociones, basado en algoritmos de aprendizaje de máquina, aprendizaje profundo y redes neuronales.

Este estudio demuestra la aplicación del análisis de sentimientos y algoritmos de aprendizaje de máquina a textos en español, así como la flexibilidad que ofrecen estos modelos ya que el número de categorías pueden variar según el propósito para el cual esté dirigido.

Arango y Osorio [9] realizaron un artículo de investigación titulado “Aislamiento social obligatorio: un análisis de sentimientos mediante machine learning”, donde se planteó analizar los sentimientos subyacentes de los comentarios de Twitter relacionados con el aislamiento, identificando los temas y palabras frecuentemente utilizados en el contexto, a través de un algoritmo de machine learning. Se obtuvo como resultado la identificación del miedo como el sentimiento predominante durante todo el periodo de confinamiento.

Este artículo de investigación demuestra que el uso del análisis de sentimientos puede ir más allá que solo clasificar los mensajes en grupos particulares, sino que también permite identificar los temas y palabras características de cada uno de los grupos resultantes.

Abordando más afondo el fenómeno migratorio, se llevó a cabo una investigación sobre crisis migratoria basada en análisis de sentimientos [10] donde se plantea que la dinámica migratoria se ha producido a razón de crisis económicas, políticas y humanitarias. Por medio de un acercamiento desde el análisis de sentimientos se obtuvieron que las preocupaciones radicaban mayormente en las subcategorías de los derechos humanos, seguridad y desempleo. En base a estos resultados se puede validar que el análisis de sentimientos permite obtener opiniones públicas sobre un tema o situación

determinada, para así tomar las medidas preventivas o correctivas correspondientes.

Adicionalmente, al aplicar la técnica de revisión documental, sobre aproximadamente más de ocho (08) referencias, [5][6][8][9][10][11][12][16], cuyas fechas de publicación estén vigentes con una década de diferencia al 2021 (fecha que se realizó la primera fase del estudio), en los cuales se aplica las técnicas de aprendizaje automático para realizar análisis de sentimientos. En este análisis documental se pudo apreciar que todos los trabajos abordaron un procedimiento metodológico basado en el entrenamiento de clasificadores de textos, aplicando las siguientes etapas:

- Etapa de recolección de datos
- Pre procesamiento de textos
- Entrenamiento de los modelos
- Despliegue de los resultados

Para poder llevar a cabo la fase de recolección de datos, en su mayoría se basaron en mensajes de la red social Twitter, y para su extracción se utilizaron, el software estadístico R y en otras ocasiones lo hicieron a través de la API de Twitter con el lenguaje de Python.

De la misma forma para llevar a cabo la construcción de los clasificadores de textos se seleccionaron diferentes algoritmos basados en aprendizaje automático supervisado, entre los más utilizados se encontraron los siguientes algoritmos:

- Naive Bayes
- Máquina de Soporte Vectorial
- Regresión Logística
- Árboles de decisión
- K-vecinos

Para evaluar el desempeño de los modelos construidos, se destacó el uso de las siguientes métricas de rendimiento:

- Precisión
- Recall
- f1-core

En resumen, para llevar a cabo un estudio de análisis de sentimiento y opinión es importante tener presente:

- El uso de la plataforma de Twitter u otra red social como fuente de información, si es adecuado para el tema a indagar
- El uso del lenguaje Python o R para la extracción de textos de las redes sociales
- Las diferentes etapas que constituyen la construcción, entrenamiento y evaluación de modelos de clasificación de textos

El enfoque más utilizado es el de aprendizaje automático, más en concreto los basados en algoritmos de clasificación tradicionales.

IV. METODOLOGÍA

El análisis de sentimientos y minería de opinión son temas que vienen en ascenso debido al impacto que pueden tener a nivel

comercial o social. Debido a esto se decidió abordar la metodología CRISP-DM (Cross-Industry Standard Process for Data Mining o Proceso Estándar Para la Extracción de Datos en Todos los Sectores) [14][15], pero teniendo en consideración que el presente estudio tratará un tipo particular de minería de datos complejos, específicamente mensajes cortos de textos, se aplicará una propuesta de adaptación de autoría propia de la metodología CRISP-DM al caso de datos complejos de textos.

Esta adaptación consistirá de dos etapas generales: una etapa de pre-procesamiento y una etapa de descubrimiento. La primera etapa, está formada por la comprensión del problema, la creación del corpus y la transformación de los textos a algún tipo de representación estructurada o semiestructurada que facilite su posterior procesamiento, mientras que la segunda etapa, está constituida por el proceso de modelado para la clasificación de mensajes, la evaluación de rendimiento y finalmente el despliegue de los conocimientos obtenidos. En la Figura 1 se pueden apreciar las etapas para la metodología CRISP-DM adaptadas al proceso de minería de textos.

A continuación, se describen las etapas que se llevan a cabo, siguiendo la metodología CRISP-DM adaptada al proceso de análisis de sentimientos, como se puede observar en la Figura 1.

A. Comprensión del Negocio

En esta primera etapa, se lleva a cabo la etapa de revisión documental relacionada con el análisis de sentimientos, las técnicas de procesamiento de lenguaje y la construcción de modelos de clasificación, para fortalecer los conceptos fundamentales presentes en el análisis.

B. Definición de los Criterios para el Análisis de Sentimientos

En esta etapa se establecen los parámetros que se tendrán en cuenta para aplicar el análisis de sentimientos, así como las premisas que deben cumplir cada uno de ellos para la selección de sus elementos.

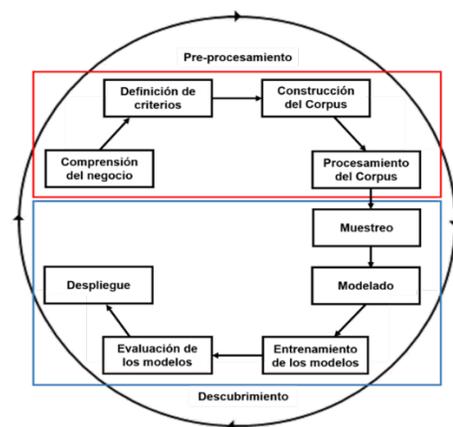


Figura 1: Adaptación de la Metodología CRISP-DM al Proceso de Análisis de Sentimientos

C. Construcción del Corpus

En esta etapa se recopilan la máxima cantidad de textos posibles que cumplan con la serie de criterios establecidos

previamente y se asocia cada uno de ellos con alguna de las categorías de la escala de sentimientos.

D. Procesamiento del Corpus

En esta instancia se llevan a cabo un conjunto de técnicas para simplificar los mensajes a representaciones estructuradas, con la intención de mejorar la interpretación del análisis automatizado y así obtener mayor precisión al momento de entrenar y evaluar los modelos de clasificación.

E. Muestreo del Corpus

En esta etapa se define la proporción de los mensajes del corpus que formarán parte de los datos de entrenamiento, utilizados en la construcción del modelo, y los datos de prueba, utilizados en la etapa de evaluación del modelo.

F. Modelado y Entrenamiento de los Clasificadores

En esta etapa se lleva a cabo el desarrollo de los modelos, a partir de los algoritmos de clasificación seleccionados y la data de entrenamiento establecida.

G. Evaluación de los Modelos de Clasificación

En esta etapa se mide el desempeño de los modelos de clasificación haciendo uso de los datos de prueba. Este proceso se logra mediante el estudio de las métricas de rendimiento que evalúan el índice de éxito del modelo para diferentes situaciones.

H. Despliegue de los Modelos de Clasificación

Finalmente, según los resultados obtenidos, los modelos pueden ser desplegados para su visualización o utilizados como base para otros trabajos con relación al área.

V. RESULTADOS

Los resultados de los objetivos más resaltantes, después de aplicar la metodología, se describen a continuación:

A. Definición de Criterios para el Análisis de Sentimientos

Entre los criterios a tomar en cuenta para el proceso de recolección de textos, que constituye la primera fase para llevar a cabo un análisis de sentimientos, se encuentran:

1) *Selección de la Plataforma.* La plataforma seleccionada para la extracción de los textos de estudio fue Twitter debido su facilidad de acceso a la API para el momento del estudio, año 2021, los formatos al realizar una extracción de datos y como indica Mathews et al. (2016) citado por Saura, Palos y Ríos [16], la red social Twitter, además de generar gran cantidad de datos y mensajes, permiten vincularlos a un evento o característica y posteriormente segmentar audiencias al recopilar mensajes en torno a un hashtag, las cuales son etiquetas ubicadas dentro de los textos para asociarlos a algún tema en particular.

2) *Selección de la Población Objetivo.* En esta etapa se seleccionaron los perfiles de twitter, de los cuales se extraerán los mensajes para el análisis. Para definir a la población se utilizaron los criterios mencionados por Arias, Villasís y Miranda [1] donde establecen que para definir una población es necesario tener en cuenta las características de homogeneidad, la temporalidad y los límites espaciales.

Teniendo en cuenta estos factores, para la selección de la población objetivo se definieron las siguientes premisas:

- Homogeneidad. Los perfiles deben pertenecer a individuos mayores a 18 años de edad. Los perfiles deben tener alguna participación sobre la migración
- Temporalidad. Los perfiles deben haber tenido participación en el año 2021, año donde se realiza la recolección de los datos a través del API
- Definir los límites espaciales. Los perfiles deben pertenecer a individuos venezolanos

Para llevar a cabo esta tarea se utilizó la herramienta web FollowerSearch, la cual es una herramienta de análisis de twitter, especializada en la búsqueda de perfiles al suministrar ciertos parámetros. Para este estudio, el criterio de selección aleatoria, de perfiles de venezolanos y mayores de edad. Haciendo uso de ella se obtuvo un total de 128 usuarios de twitter que cumplían con las características requeridas.

3) *Definición de la Escala de Opiniones y Sentimientos.* Para obtener la clasificación más precisa sobre los mensajes a analizar se determinó necesario el uso de dos escalas de categorización. La primera se basa en una escala de Likert, que según Matas [17], es uno de los instrumentos más utilizados para la medición de opiniones, en este caso se determinó el uso de tres ítems, ya que este plantea que no se encuentran diferencias significativas en las versiones de tres, cinco y siete alternativas, y para esta temática no resulta relevante la subclasificación de las posturas:

- A favor de la migración
- Neutral ante la migración
- En contra de la migración.

La segunda escala hace referencia al sentimiento asociado a la opinión emitida, en esta investigación se tomó en cuenta la clasificación propuesta por Jack, Garrod y Schyns [18] de la Universidad de Glasgow, donde sugieren que las categorías utilizadas para medir las emociones estarán formadas por las siguientes categorías:

- Felicidad
- Tristeza
- Ira
- Miedo.

4) *Enfoque de Análisis.* En el momento del análisis, los métodos de aprendizaje automático tenían mayor relevancia debido a que utilizan las capacidades de un algoritmo de clasificación para que encuentre por sí mismo a qué polaridad pertenecen las palabras y oraciones, a partir de un conjunto de datos de entrenamiento, debido a esto, la investigación se centrará en este tipo de enfoque.

5) Algoritmos de Aprendizaje Automático

Con el fin de aplicar los modelos de aprendizaje automático es necesario realizar la selección de los algoritmos de clasificación que se tomarán en cuenta al momento de construir el modelo ya que cada uno de los algoritmos de clasificación tiene una forma distintiva para llevar a cabo la categorización de los datos. En la Tabla I, se pueden observar

los algoritmos seleccionados y una breve descripción de cada uno de ellos.

Tabla I: Algoritmos de Clasificación para el Análisis de Sentimientos

Algoritmo	Descripción
Naive Bayes (NB)	Es un modelo basado en la identificación de características para cada una de las categorías. Se basa en la predominancia de un grupo de características al realizar una predicción.
Maximum Entropy (MaxE)	Es un modelo basado en regresión logística, utiliza el principio de la máxima entropía, para determinar las características mutuamente dependientes que identifican una categoría.
Árboles de Decisión (DT)	Es un modelo basado en el uso de una serie de reglas simples extraídas de los datos para obtener así una predicción verificando el cumplimiento de alguna de las obtenidas.
Máquina de Vectores de Soporte (SVM)	Es un modelo basado en la identificación de planos que le permitan delimitar las diferentes categorías

B. Construcción del Corpus

Para recopilar los mensajes de la plataforma de Twitter, se utilizó la herramienta Twint, que es una librería del lenguaje Python para el scraping de tweets, la cual tuvo como resultado 5000 tweets, donde se incluían los más recientes de cada uno de los individuos de la población objetivo. Estos mensajes fueron almacenados posteriormente en un archivo .CSV, debido a la facilidad para operar sobre él con las herramientas disponibles en el lenguaje Python.

Posteriormente se llevó a cabo la limpieza del corpus que engloba múltiples etapas, la primera de ellas, es la selección de las columnas que serán utilizadas en el análisis las cuales son las siguientes:

- Fecha de publicación
- Usuario que lo publicó
- Contenido del tweet

Luego se procedió a la eliminación de aquellos tweets que no aportan valor al desarrollo de este trabajo como son los siguientes casos:

- Tweets repetidos
- Tweets que no tienen como enfoque la migración venezolana

Como resultado se descartaron un total de 1254 mensajes, resultando una muestra de 3746 tweets que conformaron el corpus inicial. Finalmente, para concluir esta etapa, se procedió a categorizar cada uno de los mensajes, en las escalas de sentimientos anteriormente descritas. En las Tablas II y III se muestran la distribución de los tweets para cada una de las clasificaciones.

Tabla II: Distribución de los Mensajes Según su Opinión sobre la Migración

Postura sobre la migración	Número de tweets
A favor de la Migración	2702
Neutral ante la Migración	191
En contra de la Migración	853

Tabla III: Distribución de los Mensajes según la Emoción Asociada

Emociones	Número de Tweets
Felicidad	871
Tristeza	913
Ira	878
Miedo	1084

C. Pre-procesamiento del Corpus

Después de obtener el corpus inicial se llevó a cabo el procesamiento del lenguaje natural, aplicando técnicas de análisis sobre los textos de entrada que permiten mejorar la capacidad de interpretación de los algoritmos, en este estudio se llevaron a cabo las siguientes etapas:

1) Tokenización

Se utilizó el paquete de nltk.tokenize para separar cada una de las palabras y signos de puntuación que conforman un mensaje y así analizar cada una de estas por separado en las siguientes etapas.

2) Eliminación de Ruido

En esta etapa haciendo uso del paquete nltk.corpus, se removieron los siguientes casos que afectaban el análisis

- Palabras de Parada
- Espacios múltiples
- Enlaces web
- Menciones a otros usuarios
- Emoticones

3) Normalización

En esta etapa haciendo uso del paquete nltk.stem se aplicaron las siguientes técnicas de normalización

- Conversión de todos los tokens extraídos a letras minúsculas
- Stemming, donde cada una de las palabras es transformada a su raíz.

D. Muestreo del Corpus

Con la finalidad de evaluar cada uno de los modelos de clasificación seleccionados se realizaron tres pruebas, con base en el corpus creado, las cuales fueron variadas en el porcentaje de mensajes utilizados para su entrenamiento. En la Tabla IV, se pueden apreciar las pruebas realizadas y la distribución de los mensajes utilizados en cada apartado.

Tabla IV: Distribución del Corpus para las Pruebas de los Modelos

Nº Prueba	Entrenamiento (%)	Entrenamiento (Totales)	Prueba (%)	Prueba (Totales)
1	70	2622	10	375
2	80	2997		
3	90	3371		

Los mensajes que forman parte de la categoría de entrenamiento son aquellos a partir de los cuales se creará el modelo de clasificación, mientras los que pertenecen a la categoría de prueba son aquellos utilizados para evaluar el desempeño de los modelos posteriormente.

E. Entrenamiento de los Modelos de Análisis de Sentimientos

Una vez construido el corpus y desarrollados los algoritmos de clasificación, se procedió a realizar la etapa de entrenamiento del algoritmo, en la cual se les suministraron a los algoritmos los datos ya procesados y etiquetados, para así obtener los modelos clasificadores que serán los encargados posteriormente de realizar las predicciones y categorizar nuevos elementos.

F. Evaluación de los Modelos de Análisis de Sentimientos

En esta etapa se llevaron a cabo la serie de pruebas descritas en la sección del muestreo del corpus, que permitieron estimar el desempeño de los modelos de clasificación, las cuales variaron el porcentaje de mensajes utilizados para su entrenamiento. Al evaluar los resultados de las métricas de rendimiento descritas en la Tabla V.

Tabla V: Métricas de Rendimiento para Evaluar los Modelos de Clasificación

Métrica	Descripción
Precisión	Representa el porcentaje de mensajes predichos de una categoría que pertenecen a ella realmente
Sensibilidad	Representa el porcentaje de mensajes reales de una categoría que fueron correctamente asignados en la predicción
Valor F1	Es un valor que representa la precisión y la sensibilidad en un único atributo
Exactitud	Representa el porcentaje de casos que el modelo ha acertado de todas las categorías

En la Tabla VI, se muestra el resumen de los resultados obtenidos en la evaluación de rendimiento para cada uno de los modelos de clasificación.

Tabla VI: Rendimiento de los Modelos de Clasificación en la 3era Prueba

Nº Prueba	Precisión (%)	Sensibilidad (%)	Valor-F1 (%)	Exactitud (%)
Naive Bayes	79.23	76.56	77.27	77.07
Maximum Entropy	80.27	79.5	79.76	79.73
Árboles de Decisión	76.96	76.02	76.35	76.27
Máquinas de Vectores de Soporte	83.08	82.74	82.77	82.67

El rendimiento de los modelos de clasificación fue mejorando con el incremento de la data de entrenamiento, lo cual se cumplió para cada uno de los algoritmos a lo largo de las tres pruebas realizadas. Por ello, para el análisis de resultados se tomó como referencia principal la última prueba que utilizó la mayoría de los datos de entrenamiento.

El modelo que obtuvo mejores resultados bajo estas condiciones fue el basado en máquinas de vector de soporte, ver Tabla VII, que obtuvo una tasa de éxito en el 82.67% de los casos, que para la prueba realizada representó un total de 310 predicciones correctas de 375 textos seleccionados. Debido a esto, el modelo de clasificación que se utilizó principalmente en la aplicación desarrollada fue el modelo basado en máquinas de vector de soporte. Sin embargo,

también se contempló la posibilidad de utilizar los otros modelos para observar los distintos comportamientos.

Tabla VII: Resultados para la Escala de Sentimientos con la Data de Entrenamiento al 90%

Categoría	precision	recall	f1-score	confusion matrix
Felicidad	85	78.16	81.44	68 2 5 12
Ira	89.89	90.91	90.4	2 80 1 5
Miedo	87.25	81.65	84.36	2 4 89 14
Tristeza	70.19	80.22	74.87	8 3 7 73
average	83.08	82.74	82.77	accuracy 82.67

Desde el punto de vista de la escala de sentimientos y opiniones, se puede determinar el rendimiento general que tuvieron cada una de las categorías a través de los múltiples modelos de clasificación, y así reconocer las tendencias generales de los modelos. En la Figura 2 podemos apreciar que la opinión a favor de la migración fue la que tuvo el mejor desempeño, mientras que las otras posturas no lograron un buen rendimiento, esto viene dado principalmente a la diferencia en la cantidad de mensajes de entrenamiento disponibles en el corpus inicial, donde hay una predominancia de textos asociados a la categoría a favor de la migración lo que permite un modelo más refinado para detectar este tipo de textos y genera una predisposición a su clasificación.

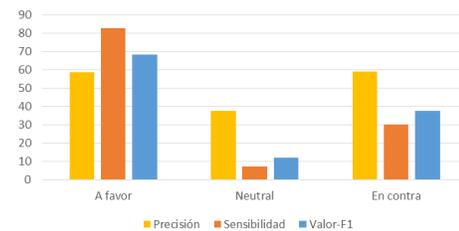


Figura 2: Rendimiento General según la Escala de Opiniones

En la Figura 3 por otro lado, podemos observar que las categorías de la escala de sentimientos se encuentran todas con un desempeño similar, esto se debe principalmente, a que la cantidad de datos de entrenamiento suministrados fueron similares para cada uno de los sentimientos.

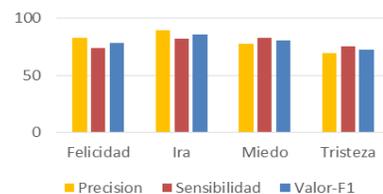


Figura 3: Rendimiento General según la Escala de Sentimientos

G. Resultados del Análisis de Sentimientos

Para obtener una visión general de la situación actual, se decidió analizar la mayor cantidad de registros de data, en vista que mejoraba el comportamiento del modelo, por lo cual se utilizó un total de 17.252 mensajes disponibles referentes al fenómeno de migración. Haciendo uso del modelo de clasificación basado en máquinas de soporte vectorial, que fue

el que obtuvo el mejor de desempeño de los modelos construidos. Los resultados fueron los siguientes:

2. Distribución de Sentimientos General

Con la finalidad de conocer la predominancia de alguna de las categorías de la escala de opiniones y sentimientos, en las Figuras 4 y 5 se disponen la distribución de los resultados obtenidos en la clasificación.

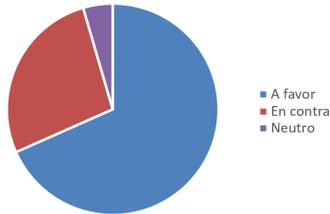


Figura 4: Distribución de la Escala de Opiniones

En la Figura 4 se puede observar que existe una tendencia de opinión clara hacia la postura a favor de la migración, el modelo clasificó aproximadamente un 64.94% de los mensajes a esta categoría, mientras que un 25.77% fueron asignados a la postura en contra y tan solo un 4.28% a la postura neutral.

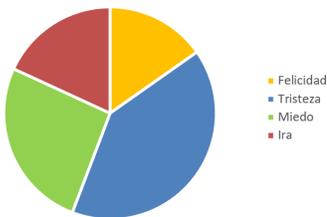


Figura 5: Distribución de la Escala de Sentimientos

Por otro lado, en cuanto a la escala de sentimientos, en la Figura 5 se puede observar una distribución más equilibrada, que se decanta ligeramente hacia la emoción de tristeza con un total de 40.55% de los mensajes, mientras que la distribución de los mensajes asignados a las categorías de miedo, ira y felicidad fueron similares con un total de 26.07%, 18.14% y 15.14% de los textos respectivamente.

3. Temas y Palabras Frecuentes

Los resultados de este estudio nos brindaron una mayor perspectiva sobre cuáles son los pensamientos de la población venezolana teniendo en cuenta su postura, para su obtención se evaluaron dos tipos de variables, los Unigramas y los N-gramas. Los Unigramas, en este caso, están representados por palabras, por ende, este método nos permite conocer aquellos términos que tienen mayor aparición en los textos, en la Figura 6 se pueden observar una lista con los Unigramas más relevantes y su número de apariciones. Por otro lado, los N-gramas están representados en este ámbito por un conjunto de palabras, que proporcionaron una perspectiva más detallada, en la Figura 7 se pueden observar los N-gramas que mayor aparición tuvieron y su número de apariciones.

Para identificar las relaciones que tienen estos términos con cada una de las categorías de la escala de opiniones, en las Figuras 8, 9 y 10 se presentan los N-gramas de las posturas a favor, en contra y neutral ante la migración respectivamente, con la intención de comparar los valores característicos de cada uno.

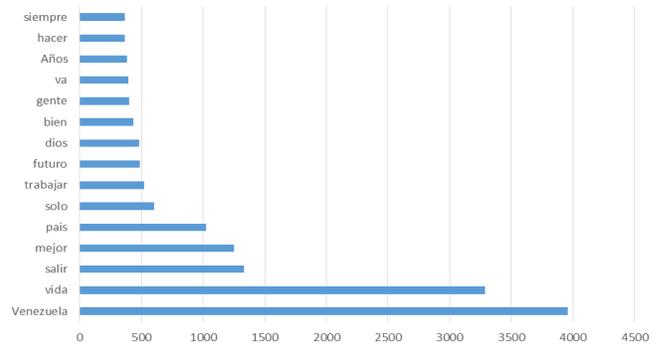


Figura 6: Unigramas con Mayor Aparición en el Análisis de Sentimientos General

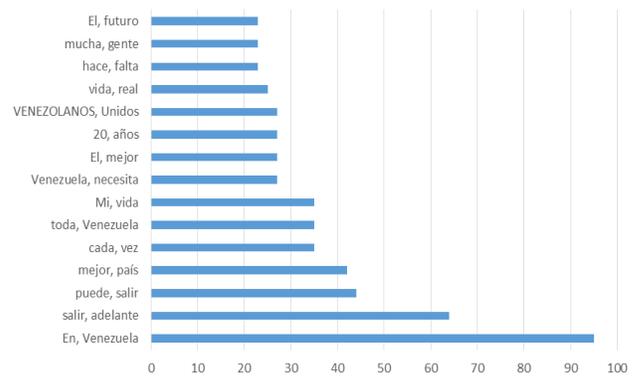


Figura 7: N-gramas con Mayor Aparición en el Análisis de Sentimientos General

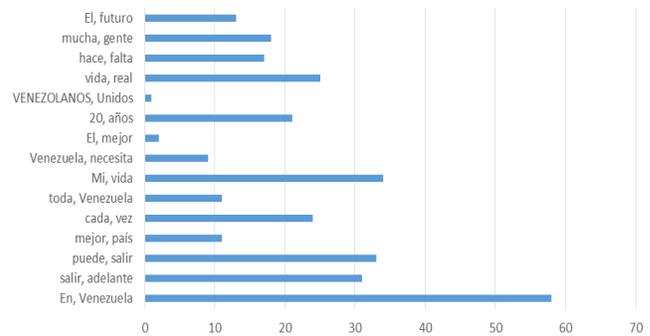


Figura 8: N-gramas con Mayor Aparición en Textos a favor de la Migración

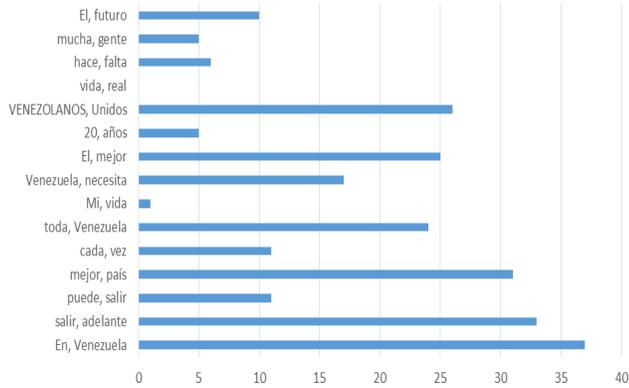


Figura 9: N-gramas con Mayor Aparición en Textos en Contra de la Migración

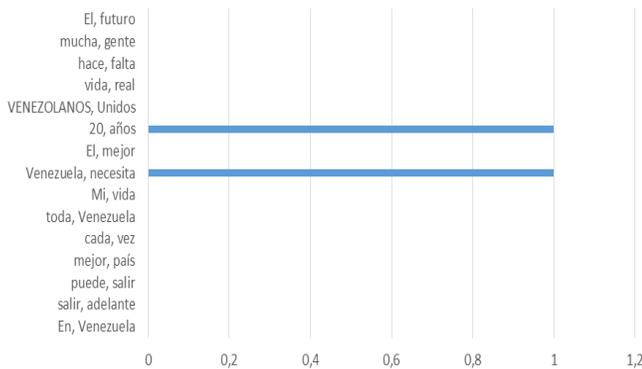


Figura 10: N-gramas con Mayor Aparición en Textos Neutrales ante la Migración

Se puede observar que los términos más relevantes de la postura a favor de la migración se enfocan a mencionar Venezuela, la vida que estos tengan en el país y la posibilidad de salir del mismo. En cuanto a la postura en contra de la migración, esta presenta un enfoque más centrado en la mejora de Venezuela y la contribución del pueblo venezolano para lograrlo. Y finalmente la postura neutral ante la migración solo hace presencia en un par de palabras debido a que el mayor número de los mensajes se ven identificados en las 2 posturas anteriores.

B. Desarrollo de la Aplicación

Para el desarrollo del aplicativo se llevó a cabo una serie de procesos, entre ellos se destacan, la creación de la base de datos, que fue la estructura encargada de almacenar los datos necesarios para la generación y evaluación de los modelos de clasificación, la construcción de una API, que fue la encargada de conectar la aplicación móvil con el servicio de base de datos y finalmente la aplicación que utilizó ambas herramientas para mostrar los resultados del análisis obtenido.

1) Base de Datos

En primera instancia, se diseñó una base de datos capaz de almacenar los valores necesarios para la etapa de entrenamiento y evaluación de los modelos de clasificación, debido a la simple naturaleza de los datos utilizados, el manejador de base de datos seleccionado fue PostgreSQL, una base de datos de tipo relacional, la cual provee un gran manejo y facilidad de acceso desde Python.

2) Application Programming Interface (API)

Para la comunicación entre la aplicación móvil y el servidor de base de datos, se desarrolló un servicio web que serviría de intermediario en este proceso. A través de la Tabla VIII, se describen las rutas para atender a las peticiones de la aplicación.

Tabla VIII: Rutas del Servicio Web Intermediario entre la Aplicación y el Servidor de Base de Datos

Ruta	Método HTTP	Descripción
/tweets	GET	Petición que devuelve todos los tweets de data de entrenamiento o los ya evaluados que estén almacenados en la Base de datos.
/tweets	POST	Petición que permite añadir un registro en la Base de Datos.
/tweets/scraping	GET	Petición que ejecuta la función para poder recolectar los datos de twitter y los procesa para ser almacenados en un archivo csv.
/tweets/classification	GET	Petición que ejecuta los modelos de predicción usando los datos de ejemplo para demostrar el funcionamiento de los mismos.
/tweets/classification/message	POST	Petición que recibe un mensaje y el nombre del usuario, para poder ejecutar los modelos predictivos sobre ese mensaje.
/tweets/kill	GET	Petición que solicita que se vacíen y rellenen la tabla de la base de datos.
/tweets/ngrams	GET	Petición para recolectar la frecuencia de las palabras que estén en los datos de prueba.
/statistics /general_probability_distribution	GET	Petición que devuelve el porcentaje de cuantos mensajes fueron clasificados en cada una de las escalas de sentimientos y los de opinión.
/statistics /distribution_by_date	GET	Petición que devuelve la cantidad de ocurrencias de cada una de las escalas de sentimiento y los de opinión pero esta vez bajo el parámetro de los meses del año.
/statistics /distribution_monograms	GET	Petición que devuelve las 15 palabras más frecuentes con sus datos de sentimientos y opinión según en los mensajes que haya aparecido.
/statistics /distribution_ngrams	GET	Petición que devuelve los 15 ngramas (Bigramas o Trigramas) más frecuentes con sus datos de sentimientos y opinión según en los mensajes que haya aparecido.

3) Aplicación Móvil

Para el desarrollo de la aplicación móvil se utilizó la herramienta de Flutter ya que como expone Herazo [19] es un marco moderno y sencillo para crear aplicaciones móviles, que tiene su propio motor de renderizado y permite ver los resultados en tiempo real.

El desarrollo de la aplicación se dividió en dos secciones principales, el desarrollo de la sección de las estadísticas del análisis de sentimientos y opiniones y la sección de predicciones utilizando los modelos de clasificación desarrollados. Para ambas secciones se hace uso de la API descrita en la Tabla VIII que permite la interacción con el servidor de base de datos.

Para la representación de las gráficas, se utilizó la librería *charts_flutter*, la cual cuenta con un conjunto extensivo de gráficas para datos estadísticos.

VI. CONCLUSIONES Y RECOMENDACIONES

A partir del análisis del problema en perspectiva de los resultados obtenidos en el trabajo, se extrajeron las siguientes conclusiones:

- La revisión del estado del arte indica que, para llevar a cabo el proceso de análisis de sentimientos y minería de opinión, la plataforma Twitter constituye una gran fuente de data, pues es de las redes sociales más utilizadas para la publicación de textos, los cuales pueden ser extraídos y procesados mediante las herramientas que brinda el lenguaje de Python, para la construcción del corpus inicial y posteriormente los modelos de clasificación.
- En la determinación de los criterios para el análisis de sentimientos y minería de opinión se identifican los parámetros que apoyan al proceso de recolección y análisis de los textos, entre ellos se destacan las características de la población objetivo, la escala de los sentimientos y opiniones que categorizan los textos y las métricas de rendimiento para evaluar el desempeño de los modelos que clasifican los textos relacionados al tema de la migración venezolana.
- Para diseñar modelos predictivos en base a textos publicados en redes sociales, es necesario la construcción de un corpus de palabras presentes en las publicaciones de venezolanos para luego determinar los sentimientos y opiniones referentes al tema de la migración con el uso de las técnicas de procesamiento del lenguaje natural y los algoritmos de aprendizaje automatizado. La calidad de los datos recolectados afecta de manera proporcional la calidad y validez de los modelos extraídos.
- Esta investigación muestra el ventajoso uso de los algoritmos de clasificación y la selección de categorías para la escala de sentimientos y opiniones para la construcción de modelos de clasificación que abarquen todas las perspectivas de la población venezolana sobre el fenómeno migratorio. Una vez entrenado el clasificador, se lleva a cabo el proceso de evaluación, haciendo uso de las métricas de rendimiento que permiten determinar su desempeño al categorizar textos relacionados a la migración venezolana.

- El resultado de la evaluación de los modelos de clasificación muestra que el basado en el algoritmo de Máquina de Soporte Vectorial es el que posee el mejor desempeño, obteniendo una exactitud general del 82.67% al momento de clasificar los mensajes de venezolanos sobre el fenómeno migratorio. El uso de este modelo para realizar el análisis de sentimientos y minería de opinión sobre la población venezolana en el año 2021 respecto al fenómeno migratorio venezolano, destaca que el sentimiento predominante es la tristeza, con un porcentaje de aparición en textos del 40.55%, y en cuanto a las opiniones, la postura a favor de la migración prevalece en el 64.94% de los casos.
- La implementación de los diferentes modelos predictivos mediante una aplicación móvil resulta ser útil, ya que permite la publicación de los resultados del análisis de datos, como la distribución general de la escala de sentimientos y opiniones, su distribución a través del tiempo y las palabras clave que identifican a cada una de las categorías, mediante una plataforma potencial de uso masivo, que apoya el proceso de toma de decisiones, suministrando información oportuna, para afrontar el fenómeno migratorio.
- El resultado obtenido en el presente trabajo evidencia la factibilidad del uso del análisis de sentimiento y minería de opinión de publicaciones en redes sociales, para la construcción de un modelo predictivo sobre el fenómeno migratorio venezolano, demostrando la razón del aumento de este tipo de investigaciones para el descubrimiento y explotación de opiniones y sentimientos de una comunidad respecto a un evento en un momento determinado.

En base a los resultados obtenidos en la presente investigación, a continuación, se enumeran una serie de recomendaciones a tomar en cuenta para trabajos futuros:

- Mejorar los modelos de clasificación, explorando la aplicación y/o combinación de otras técnicas de análisis de datos, como otras técnicas de agrupamiento, reglas de asociación, entre otras.
- Refinar el comportamiento de los modelos de clasificación descubiertos, utilizando un conjunto de datos más elevado y equitativamente distribuido entre todas las categorías.
- Explorar el uso de otros medios sociales para la recolección de datos, ya que pueden tener una representación distinta de los diferentes grupos de la sociedad venezolana.
- Experimentar el uso de otras arquitecturas, modelos de entrenamiento, librerías y entornos de desarrollo, con la finalidad de ampliar el conocimiento sobre otras opciones de análisis de sentimientos, y así elegir la más adecuada para un problema en específico.
- Ampliar el alcance de la aplicación móvil implementada, para abarcar la interacción entre usuarios, a modo de red social, permitiendo una mejora en la exposición de información para la toma de decisiones.

REFERENCIAS

- [1] J. Arias, M. Villasis, y M. Miranda. *El Protocolo de Investigación III: La Población de Estudio*. Revista Alergia México, vol. 63, no. 2, pp. 201-206. México. Abril, 2016.
- [2] R. Arias, N. Arias, M. Morffe, C. Martínez, y T. Carreño. *Informe sobre la Movilidad Humana Venezolana II Realidades y Perspectivas de Quienes Emigran (8 de abril al 5 de mayo 2019)*. ISBN: 978-980-7879-02-6, San Cristóbal, Venezuela, Junio 2019.
- [3] Plataforma de Coordinación Interagencial para Refugiados y Migrantes de Venezuela. *Refugiados y Migrantes de Venezuela*. 2021. <https://www.r4v.info/es/refugiadosymigrantes>.
- [4] Observatorio Proyecto Migración Venezuela. *Percepción en Redes Sociales sobre la Migración desde Venezuela*. 2020.
- [5] H. Hütt. *Las Redes Sociales: Una Nueva Herramienta de Difusión*. Reflexiones, vol. 91, no. 2, pp. 121-128. Universidad de Costa Rica San José, Costa Rica. 2012. <https://www.redalyc.org/pdf/729/72923962008.pdf>
- [6] B. Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers. Chicago, United States. 2012.
- [7] M. Salaz. Detección de Patrones Psicolingüísticos para el Análisis de Lenguaje Subjetivo en Español. Tesis Doctoral del Programa Oficial de Doctorado en Informática. Universidad de Murcia. España 2017. https://rua.ua.es/dspace/bitstream/10045/74617/1/PLN_60_10.pdf
- [8] A. Cestari. *Propuesta para Automatizar la Asociación de Emociones a Textos en Español*. Trabajo de Grado. Escuela de Ingeniería Informática, Universidad Católica Andrés Bello, Puerto Ordaz, Venezuela, 2019.
- [9] C. Arango y C. Osorio. *Aislamiento Social Obligatorio: Un Análisis de Sentimientos Mediante Machine Learning*. Suma de Negocios, ISSN-e 2027-5692, ISSN 2215-910X, vol. 12, no. 26, pp. 1-13, 2021.
- [10] M. Arguedas, J. Beita, F. Rodríguez, J. Umaña y M. Vaca. *Crisis Migratoria en Colombia y Costa Rica: Una Visión desde el Análisis de Sentimientos*. Revista humanidades ISSN electrónico: 2215-3934, vol. 10, no. 2, Junio 2020. <https://revistas.ucr.ac.cr/index.php/humanidades/article/view/42238>
- [11] C. González, *Clasificador de Texto Mediante Técnicas de Aprendizaje Automático*. Trabajo de Grado en Ingeniería Informática, Escuela Técnica Superior Universidad Politécnica de València, España 2020.
- [12] A. Mundalik, *Aspect Based Sentiment Analysis Using Data Mining Techniques Within Irish Airline Industry*. MSc Research Project Data Analytics. 10.13140/RG.2.2.13637.40165. Abril 2018. <http://norma.ncirl.ie/3413/1/aishwaryamundalik.pdf>
- [13] E. Páez y A. Monroy, *Implementación de un Modelo de Análisis de Sentimientos con Respecto a la JEP basado en Minería de Datos en Twitter*. Trabajo de Grado. Universidad Católica de Colombia. Facultad de Ingeniería. Programa de Ingeniería de Sistemas. Bogotá, Colombia 2020.
- [14] P. Chapman, J. Clinton, R. Keber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth. CRISP-DM 1.0 Step by Step Bgguide. Edited by SPSS. 2000.
- [15] F. Peralta, *Proceso de Conceptualización del Entendimiento del Negocio para Proyectos de Explotación de Información*. Revista Latinoamericana de Ingeniería de Software, vol. 2, no. 5, pp. 273-306, ISSN 2314-2642, 2014.
- [16] J. Saura, P. Palos-Sánchez y M. Ríos. *Un Análisis de Sentimiento en Twitter con Machine Learning: Identificando el Sentimiento sobre las Ofertas de BlackFriday*. Revista Espacios, vol. 39, no. 42, 2018.
- [17] A. Matas. *Diseño del Formato de Escalas Tipo Likert: Un Estado de la Cuestión*. REDIE ISSN 1607-4041, vol. 20, no. 1, pp. 38-47. http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1607-40412018000100038
- [18] R. Jack, O. Garrod, and P. Schyns. Dynamic Facial Expressions of Emotion Transmit an Evolving Hierarchy of Signals over Time. *Current Biology*, Vol. 24, Issue 2, Pages 187-192, ISSN 0960-9822, 2014. 2014. <https://www.sciencedirect.com/science/article/pii/S0960982213015194>
- [19] L. Herazo. *¿Qué es Flutter y por qué Utilizarlo en la Creación de Apps Móviles?*. 2021. <https://anincubator.com/que-es-flutter-y-por-que-utilizarlo-en-la-creacion-de-apps-moviles/>

Customizing Software Development Methods: A Process Model Approach

Judith Barrios, Jonás Montilva
ijudith@ula.ve, jmontilva@gmail.com

Department of Computer Science, University of Los Andes, Mérida, Venezuela

Abstract: This article presents an instantiation process model for facilitating the understanding and customization activities of software development methods. The process accepts as input a method represented by at least one process model that prescribes the set of software development activities. A method is also defined through a product model that prescribes the set of product parts that can be built following the process model and, a team model that prescribes the set of roles needed to execute the activities proposed in the process model. The proposal integrates the teaching, practical and consulting experience of the authors, which is essential to understand and handle usual difficulties found during the process of adapting software development methods. The main contribution of our proposal is to provide students of systems and software engineering with a global vision of a method and the know-how implicit in its process model description. Therefore, the proposal simplifies users understanding of method background concepts, and guides them whereas customizing it according to a particular software project context. The proposal is illustrated by an example of the White_Watch method customization to cope with a hypothetical software project situation.

Keywords: Software Methods; Instantiation Process; Method Customization; Teaching Practice.

I. INTRODUCTION

Software development methods (SDM) intend to assist project leaders and the whole development team along a development project. A SDM is typically used as a guide to establish main activities as well as technical products that the project team needs to complete to produce a software application. Nevertheless, while following the prescribed guidelines of a method, the project leader -software engineer- has to take instant decisions about the project plan workflow by considering, among other variables, the software product complexity and characteristics, the specific product dynamic requirements, the set of restrictions and needs coming from the current working environment including team size and experience, the developing tools capacities and the essentials of the programming languages, project schedule, and user participation.

In general, a SDM needs some adjustments before being used as a development guide. These adjustments serve to define the project's preliminary schedule and to organize the work that the team must do. Nevertheless, there are still some eventual context factors that may disturb project workflow like technological infrastructure, financial, and human resources complications. A project leader, therefore, needs to frequently adapt the method development process workflow. This problem of tuning a method gets bigger if the project leader/software engineer's background in method understanding and practice is not deep enough. This is the main problem encountered during teaching activities, the students just begin to understand a method when

they must already adapt it to solve a concrete problem. The main motivation of this proposal is, therefore, to assist students to customize a method that they do not completely understand. For that reason, this method instantiation process can be assumed as a teaching practice because it is effective for introducing systems and software engineering students into the context of understanding methods and then adapting them properly. Nevertheless, the proposition may also help software and systems professionals to understand and customize methods and other methodological guidelines.

The process of fine-tuning or customizing a method is called an Instantiation Process (IP). Generally, an instantiation process is done over the set of general concepts prescribed by a method model or just by a broad methodological textual description – tables or a set of steps or phases. A general method model is prescribed, at a high abstraction level, by a comprehensive set of concepts and their relationships which are involved in the process of developing a particular product or service [1].

In this article, we propose an instantiation process (IP) model for properly guiding the customization of a SDM. It means that the IP accepts a SDM represented by at least a process model, which is one of the three formal method description models. The other two are the product model which prescribes the set of product parts that can be built by following method process guidelines, and the team model which prescribes the set of roles and responsibilities that developers need to take for executing the activities prescribed as process guidelines [1][2]. This is a generic proposition; thus, it can also be instantiated to couple

with methods that only have one or two of the method models mentioned. The instantiation process model is illustrated with the customization of a SDM example that describes how to adapt the method for fitting a particular project situation of a hypothetical example.

The proposal integrates the teaching, practical and consulting experience of the authors as the basis of a good understanding of the problems faced by students and professionals during the adaptation of methods. The feedback gained from teaching system engineering students to customize methods to a specific software project and product factors has been essential. The IP model proposal is completed by ways of working derived from specific methods adaptations and characterizations completed in many Venezuelan software organizations.

The main contribution of our proposal is to help systems and software engineering students with method understanding and its customization; especially, the concerns related to a software development method adaptation for satisfying specific project needs. In addition, and considering that the SDM already customized may be applied over and over in similar project contexts, students can enhance their ways of working which are perceived as the enhancement of the quality of their development processes as well as that of the software products or services elaborated.

The article is structured as follows: the next section presents the problem of customizing SDM along with the review of some related works. Section III presents a summary of method engineering and business process model background concepts. Section IV describes, at two levels of detail, the instantiation process model proposed; first, a general contextual level and then, the corresponding detailed workflow level for describing the main activities included in the general process model. Section V shows how to apply the proposed process by instantiating the White_Watch method in a hypothetical software project situation. Section VI concludes the paper and gives some practical recommendations to take advantage of the use of the instantiation process model proposed.

II. PROBLEM AND RELATED WORK

Use this template to prepare your camera ready paper. In this template, all margins, column widths, line spaces, and text fonts are prescribed; please do not alter them.

A. Interpretations of the Problem

The problem of adapting methods is not new, and it is, generally, neglected by software organizations and project leaders since they prefer to solve method adapting problems incidentally [3]. Nevertheless, in preventing to take rush decisions related to tailoring the SDM throughout the execution of a development project, many software organizations define and institutionalize their own development methods to swift project performance and to adequately assure that a high-quality development process shall produce a high-quality software product. But the problem of adapting methods still causes difficulties that disturb project plan work breakdown structure, timelines, costs, and final product or service quality.

The software engineering community early began to work on method selection and its adapting problem. For instance, at the beginning it was identified as a problem of programmers and the knowledge they had of the development cycle along with their

capacities to define what to do, when to do it, and how much effort they needed to complete a prescribed task; so, they could organize their work and, consequently, help project leaders to estimate the required work of the whole team. These were the personal software process model (PSP) and TSP [4][5]. Afterward, the SEI proposal of the CMMI model expected to help organizations and software engineers to understand the whole software development process and, organize team development work by approving a predefined set of business development processes. Accordingly, an organization defines the complete set of software processes that ought to be installed, executed, measured, and monitored to improve organizational performance as well as the corresponding quality of the production process and its products and services [6].

Nevertheless, another customizing problem emerges when a software organization defines or selects, as mandatory, a particular SDM or decides to define and install a set of organizational software processes for elaborating and managing demanded software products. That is, the project leader and the work team must plan and execute each one of the prescribed activities to produce each prescribed technical and management product or document for realizing later, at the end of a project, that some of these activities were not useful because they do not positively contribute to obtaining the final product or service and its documentation, as required by the client. In consequence, there are a lot of time and effort lost, many expenses, and any added value either for the client or for the software organization itself. Many related proposals for software processes assessment and improvement searched to enhance and elevate installed software process performance and quality like SCAMPI, SPICE, ISO 9000 [7][13].

Some other propositions tackled the problem of adapting and defining the required product development activities by assembling a set of generic ones with or without modifications. This kind of solution was triggered by, among main issues, the software development cycle, the product type, the project management schedules according to pertinent project situational or practical variants, etc. The WATCH suite of methods is a good example of this kind of method proposal [8][9] as well as the Crystal method family [10]. These propositions preconize a set of guidelines arranged to manage some of the method adapting problems; as a result, a set of method variants is obtained where they can be selected according to the predefined set of project and product features. Other approaches extend generic activities with practical strategies like RUP (Rational Unified Process) and its agile version RUP agile [11], the ASD (Adaptive Software Development) that repeat iteratively an adapted development cycle as a practical strategy to accomplish software product dynamic or uncompleted requirements [12], [13]; and, the SCRUM project management practical approach that has been used as an agile way for organizing the development work for speedy build functional software products [14].

The SEMAT approach extends the range of the approaches reviewed. The OMG Essence standard has been published as the kernel for software engineering methods. It recommends and organizes, in a generic model, a reduced set of essential elements or concepts which are associated with any software production process. This model may well be applied by the development team to define the practical work to be done along with the

things to be produced and manipulated in a particular development project [15]. There are many related works where practitioners apply and extend the essentials to cope with development process issues [16-18].

Lastly, we add to this variety of methodological research works and practical approaches, the huge number of specialized SDM that have been proposed to help, assist, and guide the development process according to, among other settings:

- the type of product and its complexity,
- the team size, and its experience,
- the tools available and their capacity,
- the time to have a functional version,
- the dynamic change of product requirements, and
- the development technology available.

Some of these propositions recommend their method by including a list of product characteristics, time to have a functional product, team size, and project typical situations where their methods have proved to be effective [19-22]. The work presented in [23] addresses the methods configuration problem but only for the agile development approach.

None of these methodological propositions and strategies explicitly include a user guide, method use guidelines, or a set of tips to better adjust or customize their proposals. Knowledge level and understanding of a SDM or approach as well as sufficient software development experience seem to play a relevant role in the selection and, subsequently, customization of a SDM to fit the project and contextual factors. The literature review presented in [24] discusses the types of reasons behind software engineers' decisions to select and use software development methods – adoption of methods. It includes discussion about method deviations and method operationalization problems where a solid understanding of SDM background concepts along with the awareness of why and how to implement some of the method activities, seems to be a significant factor to avoid method misconception, and consequently, its misalignment to software project intentions.

We also found that many of these methodological and strategical offers are supported by social network broadcasts and an experienced set of users, programmers, and developers who communicate their practices and recommend effective ways of working like [25] and [26].

B. The Problem from Teaching Experience Perspective

To complete the description of the problem, we take some examples from our teaching experience in systems and software engineering. For instance, while working on class projects with our students, we notice that central difficulties of method usage were related to the know-how to apply a method. That is, first, students exposed problems to select, and then, how to execute a prescribed guideline/activity or to choose one of them instead of another among the set of the proposed ones. It is not easy for the students to discern if a particular action is necessary (ought to do it), if it is optional or not required considering the type of product/service or the development tool available, or a particular management exigence of the client/teacher; similar difficulty with an earlier configuration of technical products parts that are needed to complete the final software product or service. A

concern case is when a student or a project course team decides to follow each one of the prescribed guidelines to produce each one of the prescribed products parts or documents (“because the method is explicit so we have to do it”) for realizing later, at the end of the project, that a lot of the diagrams or executed actions were not necessary because they were redundant or do not contribute at all to the final product/service required.

As observed, the problem of customizing a SDM disrupts software engineering work from many perspectives. It is an individual problem, a team problem, a project leader problem, and a business process problem. We may conclude that it is not a method adapting problem but a process domain knowledge problem or a deficient practical experience. It may be true but not completely because the process of learning requires a well-founded background to have an understanding and a comprehensive picture of a method before applying it.

That is one of the reasons why this method instantiation process model was shaped: to assist method understanding and comprehension before instantiating it. From the business process perspective, this problem needs to be solved at the institutional level by defining and installing the required and flexible set of software processes by using a specific process improvement approach or standard as mentioned earlier in this section, but this discussion is out of the scope of this proposal.

We are convinced¹ that a generic instantiation process model, like our proposal, may contribute to facilitate the software engineering task of adjusting a SDM to project context situations, product/service characteristics, and team size and experience issues. It means that before starting a development project, the student, the project leader, or the system/software engineer in charge, may analyze and take decisions about the final product or service, its product parts, and documents that must be produced; besides, what would be the workflow that best fit project context situation and team members' knowledge, skills, experiences, and competencies. Method instantiation process model guidelines assist not only to adjust a method to a particular scenario but to understand what is prescribed by the method before adapting it. This premise persists and is independent of the SDM approach, i.e., if it is disciplined, balanced, or agile.

III. BACKGROUND CONCEPTS

Software methods are formally defined as a set of cohesive and complementary models that represent the software final product and its partial components, the software process/activities which need to be executed to produce each part or component of the product/service, and the competencies and understanding that each member of the software team must attest to, adequately, execute each prescribed activity, and assure that the product part elaborated has the expected quality. These models are the product model, the process model, and the team model, respectively [1][2].

The process of fine-tuning or customizing a method is called an instantiation process (IP). Generally, this kind of process is done over a set of generic concepts prescribed by a method model. This method model is general and is placed at a high abstraction level. It means that a software development model prescribes the complete set of concepts and their relationships which are

¹ After applying several times some others associated learning strategies

involved in the method development process of a software product/service. Some of them may as well include the roles that the members of a software team have to play throughout a development project.

The process of instantiating a method implies the selection, extension, reduction, or modification of any of the concepts included in the general model. In the case of a SDM represented by a process model, a product model, and a team model, the instantiation process ought to be done coherently and consistently over each one of the generic models so the relationships and dependencies between concepts and models may be correctly kept. After an instantiation process, the customized method model has a step lower abstraction level than the general one.

Figure 1 shows the links between the generic method model and the customized model obtained after an instantiation process.

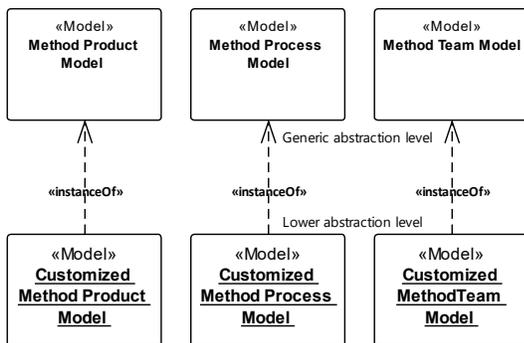


Figure 1: Links between Generic and Instantiated Method Models

A method product model represents the set of product parts and the relationships among them which can be elaborated by applying a particular method process guideline. In the case of a SDM, a method model must include the set of partial technical and management product parts as well as the deliverable ones. Figures 2, 3, and 4 present, at a very high abstraction level, a set of generic method concepts that may be instantiated to build a particular SDM product model. These concepts can be instantiated to fit specific method requirements [8][9].

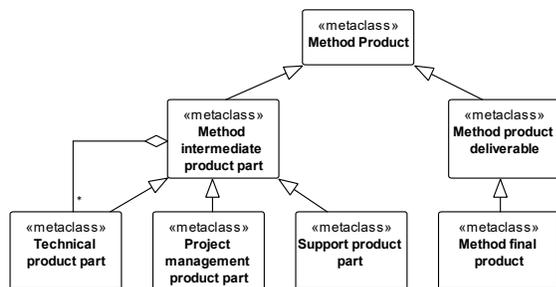


Figure 2: Generic Product Model Concepts. Adapted from [8]

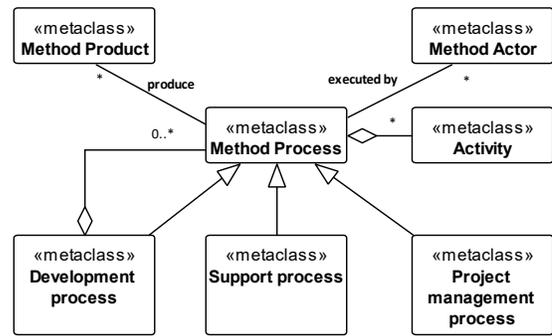


Figure 3: Generic Process Model Concepts. Adapted from [8]

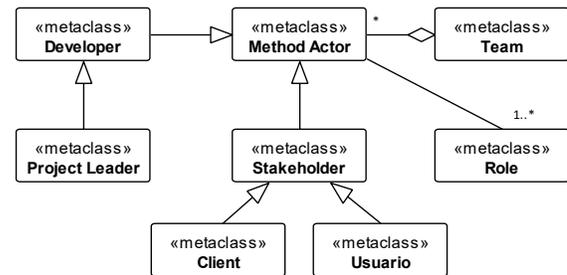


Figure 4: Generic Team Model Concepts. Adapted from [8]

As model concepts are represented at a high abstraction level, they could be chosen and extended for being part of a particular SDM. These meta-models were used to elaborate the Blue, Yellow, and White variants of the WATCH method suite [9].

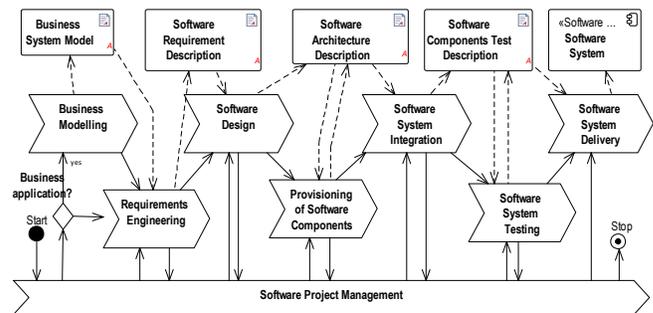


Figure 5: Workflow of the White_Watch Method [27]

Figure 5 presents an example of the results of the instantiation process of the workflow of the White_Watch method [27]. This version of the WATCH suite was defined for assisting software engineering students in their course projects. Notice that product and process generic concepts (Figures 2 and 3) were extended to have a more specific description of the technical products and their corresponding building processes. In this case, a software product is built as an assembled set of reused software components. The intermediate technical products of Figure 2 were extended with a business system model, documents for expressing software requirements, architecture, and testing descriptions. These technical products are represented by using explicit software engineering techniques and UML diagrams where applicable.

A. Process Model

The IP model proposed is graphically represented by a business process model using the UML Business notation [28]. Accordingly, a business process explicitly has a process goal to aim, is executed and supervised by actors (and their roles), has a set of inputs that may be transformed into outputs, generates other outputs, and is regulated by some precise procedures, standards, and rules, and is supported by some business resources (technology, money, documents, etc.). A business process may be complex or simple so it can be decomposed and detailed (into activities and tasks) according to modeling description requirements. A graphical representation like business process models is easy to understand and follow and provides students and engineers with a complete perspective of what they must do for customizing a software development method [29] and [30].

For that reason, the IP model is represented by a general process description diagram and a process decomposition diagram that shows the set of four sub-processes of the general process. Each one of the sub-processes is detailed by using a UML activity diagram.

IV. THE INSTANTIATION PROCESS MODEL

As we explained earlier, an instantiation process implies the selection, extension, reduction, or modification of any method element included in the general method model. This instantiation aims to generate a particular version of the method. Besides, in the case of a SDM represented by a process model, a product model, and a team model, the instantiation process ought to be done coherently and consistently over each one of the general models, so the relationships and dependencies between concepts and models can be respected.

According to the research works [8] and [9], method guidelines suggest starting the instantiation process by first customizing the method product model determining what is going to be produced by the adapted method; then, the selection of the process model elements that describe what must be done to produce the product model elements already selected. The instantiation process ends by defining the actors and their roles that are required for executing those processes/activities expressed by the process model already instantiated. Afterward, a validation process is necessary to assure that the resulting method models are coherent among them. According to these process model guidelines, the student/project leader has by now shaped a set of method models to a particular software project scenario.

Figure 6 shows the Instantiation Process description by using a general-level UML Business diagram [28]. As described in the diagram, the IP is modeled as a business process whose main goal is “to adapt method models according to project and product situational factors”. The IP process accepts general Method Models (Product, Process, and Team) and after processes guidelines, it produces the set of corresponding customized method models. The process is under the responsibility of the project leader (actor) and it may be supervised by a software engineer. There are some rules, standards, and restrictions related to method application, organizational or business domain, and other method features that must be considered while customizing method models. Project documents and other items like initial product requirements, organizational context, and some relevant

technological parameters support decisions related to the selection of model elements. Similarly, if there are any required addition or modification to complete general method model customization.

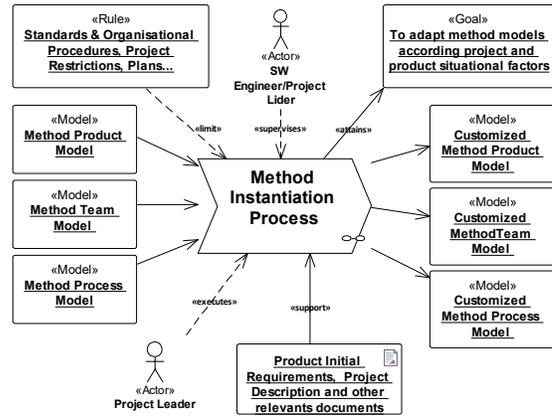


Figure 6: General Description Diagram for the Instantiation Process (IP)

Considering that the IP is a complex process it has been decomposed into four sub-processes as represented in Figure 7.

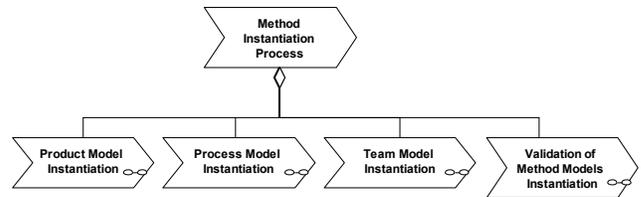


Figure 7: Sub-processes that Compose the IP

To precisely prescribe IP model guidelines, the detailed workflow for each one of the four sub-processes depicted in Figure 7 is presented in Figures 8, 9, 10, and 11, respectively.

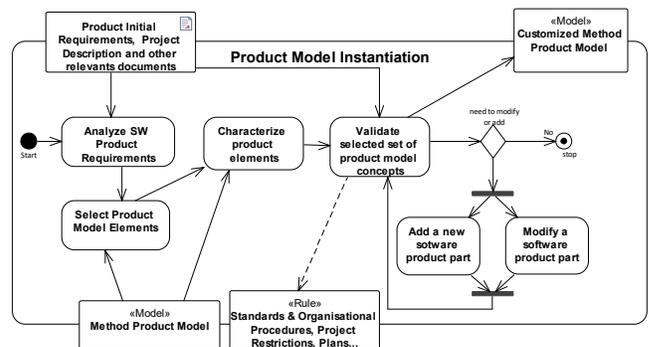


Figure 8: Product Model Instantiation

According to Figure 8, for customizing a method product model, it is necessary to analyze the initial software product requirements document as well as consider project contextual and technological aspects that may influence the selection and characterization of a set of product parts to be produced as represented in the workflow of the figure. It may be necessary to add or modify by extension or reduction one or more product parts of the customized product model. The IP of the method

product model ends after a well-structured validation of the whole set of method product elements.

Once the method product model is customized, the IP continues with the method process model to select the required set of processes/activities that are required for building each one of the product model elements included in the customized product model. If there are some new product parts or some of them have been modified, the corresponding set of method process model elements should be defined or redefined as appropriate. It is possible that the general method process model does not have all the prescribed processes, i.e., maybe it just prescribes the development processes but no the support or the project management processes. Therefore, the IP is done only on the available process model elements as represented in the activity diagram of Figure 9. It is important to have in mind that any modification to current method models must be properly described by using the same formality and/or notation that in the SDM general model.

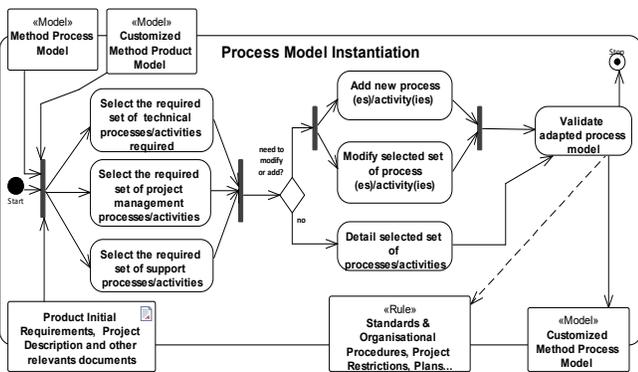


Figure 9: Process Model Instantiation

If a method team model exists in the general SDM models, it is instantiated by using as inputs the already customized method process model. This process consists of identifying and characterizing actors, roles, and responsibilities required for the execution of processes/activities included in the method process customized model as it is represented in Figure 10.

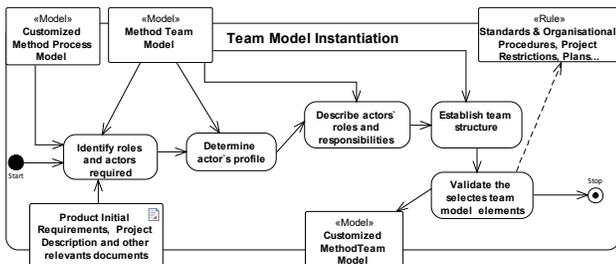


Figure 10: Team Model Instantiation

Finally, a global validation process is done to assure that the set of customized models is coherent and consistent with what is expressed in the SDM general models, and with the project and technological factors delimited by the official project documents, procedures, rules, and restrictions (see Figure 11). For example, in a SDM expressed by the three method models, the process model must use, produce, or complete a product part included in the product model. In the same way, each element

included in the process model must be described or assigned as a responsibility of an actor's role from the team model.

As we mentioned, the proposal is a generic process model, thus it is adjusted to different kinds of method descriptions and formalities. As mentioned before, the proposed IP requires, as input, at least a process model. Most methodological descriptions are represented textually through tables or as a list of steps or phases. This type of representation express activities that are performed to produce a software product (partial or complete). Therefore, the product model is implicitly included in such activities; it is the task of the project leader to extract the products involved, and to determine what part of the product they may represent and whether or not they form part of the deliverable product. The general IP workflow is showed in Figure 12.

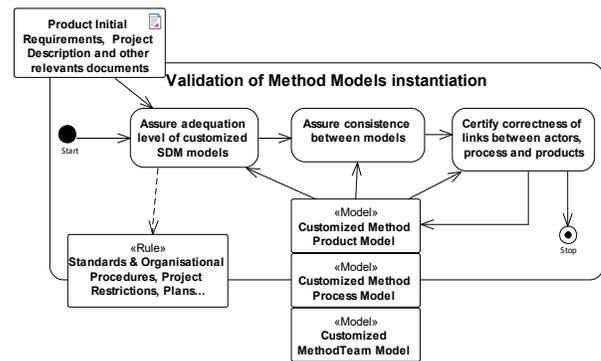


Figure 11: Validation of the Customized Set of Method Models

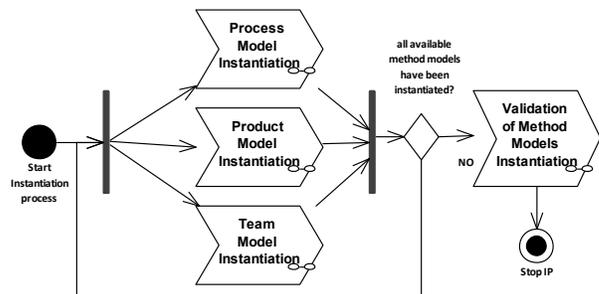


Figure 12: Instantiation Process General Workflow

V. APPLYING THE INSTANTIATION PROCESS MODEL

To illustrate the benefits of our proposal, we present the customization of the White_Watch method [27] for a hypothetical course project where the students work in a team of two software engineers.

A. Hypothetical Project Description

The first part of a software engineering course project consists in detailing the set of business processes involved in an online booking system for a small theater. These processes are part of the organizational or business system model of the small theater; thus, it is necessary to define what are the activities, actors, events, resources and objects involved in the process of booking seats in the plays offered.

B. IP Workflow for Adapting the White_Watch Method

The White_Watch method workflow showed in Figure 5 is detailed through a descriptive table that organize those processes into steps. Each step has a set of prescribed activities along with the set of techniques or notations suggested to elaborate the involved products. In view of that, students should adapt the method from the table description which has four columns: method steps, activities, notations/techniques and products. Table I shows the Business System modeling step of the White_Watch method.

Initially, it is important to state that, according to Figure 6, both students, alternatively, ought to play the role of project leader during the personalization of the process prescribed in the IP model.

Table I: Excerpt of White_Watch Method Model [27]

Steps	Activities	Notations/Techniques	Products
Business System (BS) Modeling	-Modeling BS value chain (if needed)	-Interview with BS and domain experts	-Value Chain
	-Modeling fundamental processes	-Direct observation of BS context	-Hierarchy of processes
	-Modeling Support processes	-Review of technical documentation	-Descriptions of Processes
	-Modeling process' activities	-UML Business value chain diagram	-Diagrams of activities
		-UML Business process description diagram	
		-UML activity diagram	

The relationship between product and process models is explicit and direct. Students have to select and instantiate those products, steps, and activities considering the above mentioned course project requirements; and then they can select the suggested technique or notation to produce them. IP guidelines suggest to start the instantiation process by the product model, then the process model, and finally, if required, the team model.

The outcome of the IP execution should be:

- Product model required (from Table I column “products”): one or more business processes descriptions and the corresponding set of activity diagrams.
- Process Model instantiation (Table I column “activities”): modeling fundamental processes (for the booking process) and modeling process activities (for detailing the booking process). These activities should be performed by applying techniques and notations selected from the “Notations/Techniques” column. In that case, after direct observation of the business context, it is necessary to represent perceived processes and activities by using UML business process description diagrams and UML activity diagrams.
- The adaptation of the Team Model seeks to define the roles that each of the two students has to play to execute the process model already instantiated. The actor roles prescribed by the method are “project leader”, “analyst”, “designer”, “programmer” and “tester”. In this case and as part of the teaching strategy, both students must play the role of analyst at some point in the modeling process.

Table II presents the customization of the White_Watch method after applying the IP to the example introduced above.

Table II: White_Watch Method Model Customized for the Example

Steps	Activities	Notations/techniques	Products
Business System (BS) Modeling	-Modeling fundamental processes	-Direct observation of BS context	-Descriptions of Processes
	-Modeling process' activities	-UML Business process description diagram	-Diagrams of activities
		-UML activity diagram	

VI. CONCLUSIONS

Throughout this article we presented an IP model as a teaching strategy to help students while personalizing a method they do not fully understand. The proposed model provides students with a more complete view of the components of the method and their dependency relationships, which are essential to adapt a method to the situation of a particular project. In fact, it allows to better apprehend what is the purpose of a method and how it is structured, the kind of product that can be built through the execution of a pertinent set of processes and activities, and to detect what are the competencies that developers need to have to properly implement prescribed activities.

As explained, the problem of adapting methods has many relative solutions; some of them are oriented to the implementation of method variants that partially solve the problem of choosing a suitable SDM according to some predefined factors. Others, characterize the methods according to their development approach to then suggest how to select a method from the collection. Our proposal is generic and is based on method engineering concepts so it can be adjusted to different types of method descriptions and formalities, independently of any approach, paradigm and type of method. For instance, it equally works for disciplined, balanced or agile development methods.

This proposal has proven to be an effective teaching practice for introducing system and software engineering students into the conceptual context of the development methods and their adaptation. The IP model may also assist systems professionals to understand and customize other methods and methodological guidelines than SDM.

REFERENCES

- [1] S. Brinkkemper, *Method Engineering: Engineering of Information Systems Development Methods and Tools*. Information and Software Technology, vol. 38, pp. 275-280, 1996.
- [2] J.J. Odell, *A Primer to Method Engineering*. INFOSYS: The electronic newsletter for information systems, vol. 3, no. 19, 1996.
- [3] D. Rivero, J. Montilva, J. Barrios, and M. Murúa, *Un Análisis del Desarrollo de Software en Empresas Venezolanas*. Seventh LACCEI Latin American and Caribbean Conference for Engineering and Technology- LACCEI 2009. San Cristobal, Venezuela, pp. WE1: 1-10, 2009.
- [4] W. Humphrey, *Introduction to the Personal Software Process*, Addison-Wesley, 1997.
- [5] W. Humphrey, *Introduction to the Team Software Process*, Addison-Wesley, 2000.
- [6] SEI, *CMMI for Development, Version 1.3*. Software Engineering Institute. Carnegie Mellon University, USA, Technical Report No.CMU/SEI-2010-TR-033, 2010.

- [7] M. de la Villa, M. Ruiz, and I. Ramos, *Un Estudio Crítico Comparativo de ISO 9001, CMMI e ISO 15504*, CISTI 2006, ISBN: 978-989-20-0271-2, vol. II, pp. 551-573, 2006, https://www.researchgate.net/publication/235661307_Un_estudio_critico_comparativo_de_ISO_9001_CMMI_e_ISO_15504.
- [8] J. Barrios, J. Montilva, and D. Rivero, *The WATCH Method Suite in Practice: Two Complementary Perspectives of Use*, Conference CLEI 2011, Quito, Ecuador, Octubre, 2011.
- [9] J. Barrios and J. Montilva, *Watch: A Suite of Methods for Facilitating Software Development Process Adaptability* in Software Engineering: Methods, Modeling, and Teaching, Sello Editorial Universidad de Medellín, Colombia, 2011.
- [10] A. Cockburn, *Crystal Clear*, Addison-Wesley, 2005.
- [11] S. Ambler, *The Agile Unified Process (AUP)*, 2006, <http://www.ambyssoft.com/unifiedprocess/agileUP.html>.
- [12] J. Highsmith, *Adaptive Software Development: An Evolutionary Approach to Managing Complex Systems*, Dorset House Publishing, 2000.
- [13] R. Pressman, *Software Engineering a Practitioner's Approach*, Seventh Edition, McGraw-Hill, 2010.
- [14] K. Schwaber and J. Sutherland, *The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game*, Scrum.org, October, 2017, <https://www.scrumalliance.org/learn-about-scrum/the-scrum-guide>.
- [15] I. Jacobson, Ng Pan-Wei, P. McMahon, I. Spence, and S. Lidman, *The Essence of Software Engineering: The SEMAT Kernel*, Communications of the ACM, vol. 55, no. 12, pp. 42-49, December 2012.
- [16] OMG, *Essence Kernel and Language for Software Engineering Methods Version 1.0*, November 2014, <http://www.omg.org/spec/Essence/1.0/>.
- [17] V. Savić and E. Varga, *Extending the SEMAT Kernel with the TDD practice - IET Software*, Wiley Online Library, 2018, <https://doi.org/10.1049/iet-sen.2016.0305>.
- [18] J. S. Park, *Essence-Based, Goal-Driven Adaptive Software Engineering*, IEEE/ACM 4th SEMAT Workshop on a General Theory of Software Engineering, 2015, pp. 33-38, DOI: 10.1109/GTSE.2015.12.
- [19] J. Montilva and J. Barrios, *Ingeniería del Software: Un enfoque Basado en Procesos*, Sello Editorial del Vicerrectorado Académico, Universidad de los Andes, Venezuela, August, 2021, Published by Amazon Books, <https://www.amazon.com/-/es/Jon%C3%A1s-Montilva-C/dp/9801120290>.
- [20] J. Barrios and J. Montilva, *A Balanced and Adaptable Method for Software Development in very Small Enterprises: The Blue Watch Variant*, in Carlos M. Zapata et al. (eds), *Software Engineering: Methods, Modeling and Teaching*: pp. 39-54, Medellín, Sello Editorial, 2011.
- [21] J. Barrios and J. Montilva, *Gestión Ágil de Proyectos a Través de la Integración de Blue-WATCH y SCRUM*, X Ibero-American Conference on Software Engineering and Knowledge Engineering – JIISIC 2013, University of Medellín, Medellín, Colombia, November, 2013.
- [22] M. A. Khan, A. Parveen, and M. Sadiq, *A Method for the Selection of Software Development Life Cycle Models Using Analytic Hierarchy Process*, International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014, pp. 534-540, DOI: 10.1109/ICICT.2014.6781338.
- [23] D. Gupta and R. Dwivedi, *Configurable Method Model of Agile Methods for Creating Project-Specific Methods*, International Conference on Software Eng. Research and Practice, SERP'16, ISBN: 1-60132-446-4, CSREA Press, 2016.
- [24] T. Havstorm and F. Karlsson, *Software Developers Reasoning Behind Adoption and Use of Software Development Methods – A Systematic Literature Review*, in International Journal of Information Systems and Project Management, vol. 11, no. 2, article 4, 2023, <https://aisel.aisnet.org/ijispm/vol11/iss2/4>.
- [25] Toptal Engineering Blog, <https://www.toptal.com/developers/blog>.
- [26] SoVisionIT, *Software and APP Development*, <https://www.sovisionit.com/software-app-development/interesting-stuff/blog/>.
- [27] J. Barrios and J. Montilva, *Método W Watch*, Revised version 2020. Internal Report, Group GIDyC, School of Systems Engineering, University of Los Andes, Mérida, Venezuela, 2020, partially published in J. Montilva & J. Barrios [21].
- [28] H. Eriksson and M. Penker, *Business Modeling with UML: Business Patterns at Work*, John Wiley & Sons, 2000.
- [29] K. Figl, *Comprehension of Procedural Visual Business Process Models: A Literature Review*, Business and Information Systems Engineering, vol. 59, no. 1, February 2017, pp. 41–67.
- [30] E. Kornysheva and J. Barrios, *Visual Representation of the TOGAF Requirements Management Process*, ER Workshops 2225, October 2018, LNCS, vol. 11158, pp. 239-248.

Índice de Autores

B

Barrios Judith	20
Borjas Livia	10

Z

Zamora Miguel	10
Zanek Franco	1

M

Montilva Jonás	20
Morales Mauricio	10

REVECOM

Sociedad Venezolana de Computación

La Sociedad Venezolana de Computación está comprometida con el impulso de una nueva generación académica y profesional en nuestra área de saber para el desarrollo del país.

Los conceptos y puntos de vista expresados en los trabajos publicados en este libro representan las opiniones personales de los autores y no reflejan el juicio de los editores o de la Sociedad Venezolana de Computación.

ISSN: 2244-7040



9 772244 704006



<https://svc.net.ve/revecom>