



ISSN: 1646-9895

Revista Ibérica de Sistemas e Tecnologias de Informação
Revista Ibérica de Sistemas y Tecnologías de Información

O u t u b r o 2 0 • O c t o b e r 2 0



©AISTI 2020 <http://www.aisti.eu>

Nº 39

Edição / Edición

Nº 39, 10/2020

Tiragem / Tirage: 1000

Preço por número / Precio por número: 17,5€

Subscrição anual / Suscripción anual: 30€ (2 números)

ISSN: 1646-9895

Depósito legal:

Indexação / Indexación

Academic Journals Database, CiteFactor, Dialnet, DOAJ, DOI, EBSCO, GALE, IndexCopernicus, Index of Information Systems Journals, ISI Web of Knowledge, Latindex, ProQuest, QUALIS, SciELO, SCImago, Scopus, SIS, Ulrich's.

Propriedade e Publicação / Propriedad y Publicación

AISTI – Associação Ibérica de Sistemas e Tecnologias de Informação

Rua Quinta do Roseiral 76, 4435-209 Rio Tinto, Portugal

E-mail: aistic@gmail.com

Web: <http://www.risti.xyz>

Director

Álvaro Rocha, Universidade de Lisboa, PT

Coordenadores da Edição / Coordinadores de la Edición

Gloria Piedad Gasca-Hurtado, Universidad de Medellín, CO

Liliana Machuca-Villegas, Universidad del Valle, CO

Conselho Editorial / Consejo Editorial

Carlos Ferrás Sexto, Universidad de Santiago de Compostela, ES

Gonçalo Paiva Dias, Universidade de Aveiro, PT

Jose Antonio Calvo-Manzano Villalón, Universidad Politécnica de Madrid, ES

Luís Paulo Reis, Universidade do Porto, PT

Manuel Pérez Cota, Universidad de Vigo, ES

Ramiro Gonçalves, Universidade de Trás-os-Montes e Alto Douro, PT

Conselho Científico / Consejo Científico

A. Augusto Sousa, FEUP, Universidade do Porto, PT

Adolfo Lozano-Tello, Universidad de Extremadura, ES

Adrián Hiebra Pardo, Universidad de Santiago de Compostela, ES

Alberto Fernández, Universidad Rey Juan Carlos, ES

Alberto Freitas, FMUP, Universidade do Porto, PT

Alcinia Zita Sampaio, IST, Universidade de Lisboa, PT

Alejandro Peña, Escuela de Ingeniería de Antioquia, CO

Alexandre L'Erario, Universidade Tecnológica Federal do Paraná, BR

Alicia García-Holgado, Universidad de Salamanca, ES

Alma Gomez-Rodríguez, Universidade de Vigo, ES

Ana Amélia Carvalho, Universidade de Coimbra, PT

Ana Isabel Veloso, Universidade de Aveiro, PT

Ana Maria Correia, ISEGI, Universidade Nova de Lisboa, PT

Ana Paula Afonso, Instituto Politécnico do Porto, PT

Anabela Mesquita, Instituto Politécnico do Porto, PT

Anacleto Correia, Escola Naval, PT

Angelica Caro, Universidad del Bío-Bío, CL

Ana Calvão, Universidade de Aveiro, PT

Ana Carla Amaro, Universidade de Aveiro, PT
Ana Melro, Universidade de Aveiro, PT
Ania Cravero, Universidad de La Frontera, CL
Aníbal Zaldivar-Colado, Universidad Autonoma de Sinaloa, MX
António Abreu, ISCAP, Politécnico do Porto, PT
António Coelho, FEUP, Universidade do Porto, PT
Antonio Fernández-Caballero, Universidad de Castilla-La Mancha, ES
António Godinho, ISLA-Gaia, PT
Antonio Jesus Garcia Loureiro, Universidad de Santiago de Compostela, ES
Antonio Jiménez-Martín, Universidad Politécnica de Madrid, ES
António Palma dos Reis, ISEG, Universidade de Lisboa, PT
António Pereira, Instituto Politécnico de Leiria, PT
Armando Mendes, Universidade dos Açores, PT
Arnaldo Martins, Universidade de Aveiro, PT
Arturo J. Méndez, Universidad de Vigo, ES
August Climent Ferrer, La Salle Open University, AD
Beatriz Rodríguez, Universidad de la República, UY
Beatriz Sainz de Abajo, Universidad de Valladolid, ES
Bernabé Escobar-Pérez, Universidad de Sevilla, ES
Borga Bordel, Universidad Politécnica de Madrid, ES
Bráulio Alturas, ISCTE - Instituto Universitário de Lisboa, PT
Brenda L. Flores-Rios, Universidad Autónoma de Baja California, MX
Carlos Alexandre Silva, Instituto Federal de Minas Gerais, PT
Carlos Carreto, Instituto Politécnico da Guarda, PT
Carlos Morais, Instituto Politécnico de Bragança, PT
Carlos Vaz de Carvalho, Instituto Politécnico do Porto, PT
Carmen Galvez, Universidad de Granada, ES
Carlos Rabadão, Politécnico de Leiria, PT
Carlos Rompante Cunha, Politécnico de Bragança, PT
Ciro Martins, Universidade de Aveiro, PT
Cristina M.R. Caridade, ISEC, Politécnico de Coimbra, PT
Daniel Polónia, Universidade de Aveiro, PT
David Fonseca, Universitat Ramon Llull, ES
David Luis La Red Martínez, Universidad Nacional del Nordeste, AR

David Ramos Valcarcel, Universidad de Vigo, ES
Debora Paiva, Universidade Federal de Mato Grosso do Sul, BR
Dora Simões, Universidade de Aveiro, PT
Edna Dias Canedo, Universidade de Brasília, BR
Eduardo Amadeu Dutra Moresi, Universidade Católica de Brasília, BR
Eduardo Sánchez Vila, Universidad de Santiago de Compostela, ES
Enric Mor, Universitat Oberta de Catalunya, ES
Eusébio Ferreira da Costa, Escola Superior de Tecnologias de Fafe, PT
Fábio Marques, Universidade de Aveiro, PT
Fernando Bandeira, Universidade Fernando Pessoa, PT
Fernando Bobillo, Universidad de Zaragoza, ES
Fernando Moreira, Universidade Portucalense, PT
Fernando Paulo Belfo, ISCAC, Politécnico de Coimbra, PT
Fernando Ramos, Universidade de Aveiro, PT
Fernando Ribeiro, Politécnico de Castelo Branco, PT
Filipe Caldeira, Politécnico de Viseu, PT
Filipe Montargil, Politécnico de Lisboa, PT
Filipe Portela, Universiade do Minho, PT
Francisco Javier Lena-Acebo, Universidad de Cantabria, ES
Francisco Restivo, Universidade Católica Portuguesa, PT
Gabriel Alberto García-Mireles, Universidad de Sonora, MX
Gabriel Guerrero-Contreras, Universidade de Cádiz, ES
Gerardo Gonzalez Filgueira, Universidad da Coruña, ES
Gloria Piedad Gasca-Hurtado, Universidad de Medellín, CO
Guilhermina Lobato Miranda, Universidade de Lisboa, PT
Hélder Gomes, Universidade de Aveiro, PT
Hélder Zagalo, Universidade de Aveiro, PT
Hélia Guerra, Universidade dos Açores, PT
Henrique S. Mamede, Universidade Aberta, PT
Higino Ramos, Universidad de Salamanca, ES
Inês Domingues, CI-IPOP, PT
Isabel Pedrosa, Instituto Politécnico de Coimbra, PT
Isidro Calvo, Universidad del País Vasco (UPV/EHU), ES
Ismael Etxeberria-Agiriano, Universidad del País Vasco (UPV/EHU), ES

Ivaldir de Farias Junior, Universidade de Pernambuco, BR
Ivan Garcia, Universidad Tecnológica de la Mixteca, MX
João Paulo Ferreira, ISEC, Politécnico de Coimbra, PT
João Reis, Universidade de Aveiro, PT
João Roberto de Toledo Quadro, CEFET/RJ, BR
Jacinto Estima, Universidade Europeia, PT
Javier Garcia Tobio, CESGA-Centro de Supercomputacion de Galicia, ES
Javier Medina, Universidad Distrital Francisco José de Caldas, CO
Jeimy Cano, Universidad de los Andes, CO
Jezreel Mejia, Centro de Investigación en Matemática (CIMAT), MX
João Balsa, FC, Universidade de Lisboa, PT
João Paulo Costa, Universidade de Coimbra, PT
João Rocha da Silva, FEUP, Universidade do Porto, PT
João Tavares, FEUP, Universidade do Porto, PT
João Vidal de Carvalho, ISCAP, Politécnico do Porto, PT
Joaquim Ferreira, Universidade de Aveiro, PT
Joaquim Reis, ISCTE - Instituto Universitário de Lisboa, PT
Jorge Bernardino, ISEC, Politécnico de Coimbra, PT
Jorge da Silva Correia-Neto, Universidade Federal Rural de Pernambuco, BR
Jose Alfonso Aguilar, Universidad Autonoma de Sinaloa, MX
José Alvarez-Garcia, Universidad de Extremadura, ES
José Borbinha, IST, Universidade de Lisboa, PT
José Carlos Ribeiro, Politécnico de Leiria, PT
José Cascalho, Universidade dos Açores, PT
José Felipe Cocón Juárez, Universidad Autónoma del Carmen, MX
Jose J. Pazos-Arias, Universidad de Vigo, ES
José Luís Pereira, Universidade do Minho, PT
José Luís Silva, Instituto Universitário de Lisboa (ISCTE-IUL), PT
José Paulo Lousado, Instituto Politécnico de Viseu, PT
José Luis Pestrana Brincones, Universidad de Málaga, ES
José Luís Reis, ISMAI - Instituto Universitário da Maia, PT
Jose M Molina, Universidad Carlos III de Madrid, ES
José Machado, Universidade do Minho, PT
Jose Maria de Fuentes, Universidad Carlos III de Madrid, ES

Jose R. R. Viqueira, Universidade de Santiago de Compostela, ES
José Silvestre Silva, Academia Militar, PT
José Torres, Universidade Fernando Pessoa, PT
Josep M. Marco-Simó, Universitat Oberta de Catalunya, ES
Juan Angel Contreras Vas, Universidad de Extremadura, ES
Juan D'Amato, PLADEMA-UNCPBA-CONICET, AR
Juan M. Santos Gago, Universidad de Vigo, ES
Juan Manuel Fernández-Luna, Universidad de Granada, ES
Jugurta Lisboa-Filho, Universidade Federal de Viçosa, BR
Leila Weitzel, Universidade Federal Fluminense, BR
Leonardo Bermon, Universidad Nacional de Colombia, CO
Leticia Morales Trujillo, Universidad de Sevilla, ES
Lucila Ishitani, PUC Minas, BR
Lucila Romero, Universidad Nacional del Litoral, AR
Luis Alvarez Sabucedo, Universidad de Vigo, ES
Luís Bruno, Instituto Politécnico de Beja, PT
Luis Camarinha-Matos, Universidade Nova de Lisboa, PT
Luís Cavique, Universidade Aberta, PT
Luis Chamba Eras, Universidad Nacional de Loja, EC
Luís Ferreira, Instituto Politécnico do Cávado e do Ave, PT
Luis Enrique Sánchez Crespo, Universidad de Castilla-La Mancha, ES
Luis Vilán-Crespo, Universidad de Vigo, ES
Luisa Miranda, Instituto Politécnico de Bragança, PT
Lus Sussy Bayona Ore, Universidad Nacional Mayor de San Marcos, PE
Magdalena Arcilla Cobián, Universidade Nacional de Educación a Distancia, ES
Manuel Fernández-Veiga, Universidad de Vigo, ES
Manuel Jose Fernandez Iglesias, Universidad de Vigo, ES
Marcelo Marciszack, Universidad Tecnológica Nacional, AR
Marcelo de Paiva Guimarães, Universidade Federal de São Paulo, BR
Marco Painho, ISEGI, Universidade Nova de Lisboa, PT
Maria Amelia Eliseu, Mackenzie Presbyterian University, BR
Maria Cristina Marcelino Bento, UNIFATEA, BR
Maria do Rosário Bernardo, Universidade Aberta, BR
Maria Hallo, Escuela Politécnica Nacional, EC

Maria Helena Garcia Ruiz, Universidad de Cantabria, ES
María J. Lado, Universidad de Vigo, ES
Maria João Ferreira, Universidade Portucalense, PT
Maria João Gomes, Universidade do Minho, PT
Maria José Angélico, Instituto Politécnico do Porto, PT
Maria José Sousa, Universidade Europeia, PT
Marisol B. Correia, Universidade do Algarve, PT
Maristela Holanda, Universidade de Brasília, BR
Martín Llamas Nistal, Universidad de Vigo, ES
Martín López Nores, Universidad de Vigo, ES
Matías García Rivera, Universidad de Vigo, ES
Mercedes Ruiz, Universidad de Cádiz, ES
Miguel A. Brito, Universidade do Minho, PT
Miguel Bugalho, Universidade Europeia, PT
Miguel Casquilho, IST, Universidade de Lisboa, PT
Mirna Ariadna Muñoz Mata, Centro de Investigación en Matemáticas (CIMAT), MX
Nelson Rocha, Universidade de Aveiro, PT
Nuno Lau, Universidade de Aveiro, PT
Nuno Melão, Politécnico de Viseu, PT
Nuno Ribeiro, Universidade Fernando Pessoa, PT
Oscar Mealha, Universidade de Aveiro, PT
Patricia Dias, Universidade do estado de Minas Gerais, BR
Patrícia Oliveira, Universidade de Aveiro, PT
Paula Prata, Universidade da Beira Interior, PT
Paulo Martins, Universidade de Trás-os-Montes e Alto Douro, PT
Paulo Pinto, FCT, Universidade Nova de Lisboa, PT
Paulo Rurato, Universidade Fernando Pessoa, PT
Paulo Urbano, FC, Universidade de Lisboa, PT
Pedro Araújo, Universidade da Beira Interior, PT
Pedro Sanz Angulo, Universidad de Valladolid, ES
Pedro Sobral, Universidade Fernando Pessoa, PT
Pedro Sousa, Universidade do Minho, PT
Pilar Mareca Lopez, Universidad Politécnica de Madrid, ES
Ramiro Delgado, Universidad de las Fuerzas Armadas, EC

Ramon Alcarria, Universidad Politécnica de Madrid, ES
Raul Laureano, ISCTE - Instituto Universitário de Lisboa, PT
Renato Maurício Toasa Guachi, Universidad Tecnológica Israel, Ecuador
Ricardo J. Rodríguez, Universidad de Zaragoza, ES
Ricardo Linden, FSMA, BR
Rita Oliveira, Universidade de Aveiro, PT
Rita Santos, Universidade de Aveiro, PT
Roberto Theron, Universidad de Salamanca, ES
Rodolfo Miranda Barros, Universidade Estadual de Londrina, BR
Román Lara, Universidad de las Fuerzas Armadas, EC
Rubén González Crespo, Universidad Internacional de La Rioja, ES
Rui Cruz, IST, Universidade de Lisboa, PT
Rui José, Universidade do Minho, PT
Rui Pedro Marques, Universidade de Aveiro, PT
Rui S. Moreira, Universidade Fernando Pessoa, PT
Samuel Sepúlveda, Universidad de La Frontera, CL
Santiago Gonzales Sánchez, Universidad Inca Garcilaso de la Vega, PE
Sara Balderas-Díaz, Universidad de Cádiz, ES
Saulo Barbara de Oliveira, Universidade Federal Rural do Rio de Janeiro, BR
Sérgi F. Lopes, Universidade do Minho, PT
Sergio Gálvez Rojas, Universidad de Málaga, ES
Sérgio Guerreiro, Universidade Lusófona de Humanidades e Tecnologias, PT
Silvia Fernandes, Universidade do Algarve, PT
Solange N Alves de Souza, Universidade de São Paulo, BR
Telmo Silva, Universidade de Aveiro, PT
Teresa Guarda, Universidad Estatal Península de Santa Elena, EC
Thiago Dias, CEFET-MG, BR
Valéria Farinazzo Martins, Universidade Presbiteriana Mackenzie, BR
Verónica Vasconcelos, ISEC, Politécnico de Coimbra, PT
Vicente Morales, Universidad Técnica de Ambato, EC
Victor Flores, Universidad Católica del Norte, CL
Victor Hugo Medina Garcia, Universidad Distrital Francisco José de Caldas, CO
Vitor Carvalho, Instituto Politécnico do Cávado e do Ave, PT
Vitor Santos, ISEGI, Universidade Nova de Lisboa, PT

Wagner Tanaka Botelho, Universidade Federal do ABC, BR
Xose A. Vila, Universidad de Vigo, ES

El desarrollo de software como habilitador digital para la 4ta revolución industrial

The software development as a digital enabler for the 4th industrial revolution

Gloria Piedad Gasca-Hurtado¹, Liliana Machuca-Villegas^{1,2}

gpgasca@udem.edu.co; liliana.machuca@correounivalle.edu.co

¹ Universidad de Medellín, Carrera 87 No. 30-65, 50026, Medellín, Colombia

² Universidad del Valle, Calle 13 # 100-00, 760032, Cali, Valle del Cauca, Colombia

DOI: 10.17013/risti.39.0

1. Introducción

Los artículos presentados en este número fueron seleccionados de los trabajos aceptados en la Décimo Tercera Edición del Seminario Internacional de Ciencias de la Computación (SICC) 2020, realizado en la Ciudad de Medellín, Antioquia, Colombia el día 13 de octubre del 2020. Este seminario ha tenido una tasa de aceptación del 37.5% de los artículos enviados.

En este número de la Revista Ibérica de Sistemas y Tecnologías de la Información (RISTI), se abordan temas relacionados con el desarrollo de software como habilitador digital para la 4ta revolución industrial. Los trabajos de esta edición son las versiones originales presentadas por los autores en comunicación oral durante el congreso SICC 2020 y que fueron seleccionados después de realizarse un proceso evaluación por pares doble ciego.

2. Desarrollo de software como habilitador digital para la 4ta revolución industrial

En la edición anterior de esta revista relacionada con el Seminario Internacional de Ciencias de la Computación (SICC) se definió 4ta revolución industrial (Gasca-Hurtado & Machuca-Villegas, 2019), a partir de su relación con términos como Fábricas Inteligentes

o Smart Factory, Tecnología Inteligente o Smart Inteligent, sistemas ciberfísicos, transformación digital. Además, se consideraron temas de responsabilidad social a nivel corporativo y ético, con el fin de tener en cuenta aspectos relacionados con el impacto negativo social, ambiental o económico que puede llegar a afectar a las personas y al planeta los desarrollos asociados con la 4ta revolución industrial (Sanchez, 2019).

Continuando con este interés, en esta edición hemos centrado la atención en el desarrollo de software como habilitador digital de la 4ta revolución industrial. Para lo cual se han analizado las diferentes tendencias de la investigación relacionadas con sus habilitadores. Entre los resultados más relevantes de dichos análisis está la agrupación de los principales habilitadores que destaca categorías como los factores humanos, de organización y de tecnología. En este caso y de manera transversal es posible identificar el desarrollo de software como un habilitador transversal que permea las tres agrupaciones mencionadas (Havle & Üçler, 2018).

Sin embargo, un estudio que analiza los avances de las empresas industriales de Europa con respecto a la implementación de la Industria 4.0. logra concluir aspectos importantes a tener en cuenta. Dentro de las conclusiones más relevantes del estudio es la relacionada con los niveles de desarrollo de las estrategias de digitalización. A pesar del impulso e interés de las empresas en Europa por tener un alto nivel de desarrollo de dichas estrategias, aún parece ser bastante superficial (Mogos, Eleftheriadis, & Myklebust, 2019).

Lo anterior, puede estar orientado a que existe un alto potencial de los beneficios de la Industria 4.0 sin explotar. Analizando los resultados de dicho estudio, a la luz del desarrollo de software como habilitador de la Industria 4.0, es posible identificar que los potenciales sin explotar están alrededor de la tecnología. Algunos de ellos como aumentar el uso de las nuevas tecnologías TIC en la industria y el uso de datos como activo principal de una organización (Mogos et al., 2019). Otro factor interesante es la generación de competencias de digitalización del personal, catalogado como un inhibidor de la Industria 4.0, ya que dichas competencias son vitales para el desarrollo de la empresa.

En términos generales, es posible asociar esta nueva era de la industria 4.0 con los robots que se conectan con las computadoras y las máquinas. Es decir, las herramientas están trabajando en algoritmos de aprendizaje de máquinas y ejecutando los sistemas de fabricación física cibernética (Kumar, 2020). Por lo tanto, el desarrollo de software se convierte en uno de los habilitadores esenciales para la adhesión de la Industria 4.0 en los diferentes sectores de la industria, siendo el sector manufacturero el de mayor relevancia.

Por lo tanto, es posible identificar varias clasificaciones de habilitadores entre los que se encuentran las propuestas por Mogos et al. (2019) y Kumar (2020). Sin embargo, a pesar de los diferentes enfoques que pueden tener ambas iniciativas, el común denominador de los habilitadores es el desarrollo de tecnología, tanto hardware como software, necesario para cumplir con el requisito de ejecutar la maquinaria que requiere el mundo en esta era de la Industria 4.0.

Lo anterior motiva a la definición de esta temática para la versión del Seminario Internacional de la Computación SICC 2020, con el fin de conocer los avances tecnológicos que investigadores y académicos proponen como solución a las problemáticas que asume el mundo en esta nueva era. En sintonía con los estudios que definen y clasifican habilitadores de la Industria 4.0, los trabajos presentados en esta edición de la

revista van desde las necesidades en la generación de capacidades de transformación digital en etapas tempranas de la formación como la educación básica y media, hasta propuestas de análisis inteligente de datos, tratamiento y mejoramiento de algoritmos usando aprendizaje de máquina. También se incluyen propuestas de gestión, trabajo colaborativo y análisis emocional que le aportan al mundo interesantes enfoques en esta revolución industrial.

3. Estructura

En el primer artículo los autores presentan una estrategia de transformación para la formación en informática en instituciones de educación básica y media (ByM), orientada hacia el desarrollo de competencias para la Industria 4.0, centradas en el pensamiento creativo y computacional. La propuesta se compone de dos ejes fundamentales: diseño curricular y diseño didáctico, así como de un ecosistema de formación. Esta propuesta se presenta como una alternativa de formación en informática a partir de la educación inicial, con dos ejes integradores: programación y robótica. Los beneficiarios directos son administradores educativos y profesores, dado que la propuesta facilita las acciones de diseño curricular/didáctico y su aplicación práctica en el aula. Asimismo, se define un marco de trabajo que facilita el desarrollo de un *Ecosistema de Formación* en las instituciones de educación ByM que estén incursionando en procesos de renovación curricular. Los autores consiguieron refinar el diseño curricular de la propuesta y probar algunos de los talleres resultantes de la especificación didáctica con grupos de estudiantes de formación básica y media. Adicionalmente, consiguieron dar el primer paso para la conformación del ecosistema de formación en la ciudad de Medellín (Colombia) a nivel de educación básica y media, a través de la divulgación de resultados a profesores y coordinadores de informática de las instituciones educativas.

En el segundo artículo se propone un conjunto de reglas heurísticas a la luz de la simulación como una de las áreas clave de la Industria 4.0. La propuesta se realiza con el fin de obtener código PL/pgSQL a partir de esquemas preconceptuales, como una forma de simular el comportamiento de las aplicaciones que incluyen eventos. Estas reglas se validan mediante un estudio de laboratorio correspondiente a la expansión de una epidemia. Mediante la ejecución de las reglas en el estudio de laboratorio se puede simular el comportamiento futuro de la aplicación y los resultados asociados. Con el código generado fue posible la generación automática de los valores que se predijeron para el modelo. El código se generó completamente a partir de las reglas y fue completamente funcional en el gestor de bases de datos PostgreSQL, de código abierto. Dichas reglas constituyen un ejemplo de la manera en que la industria del software emplea la simulación como un área clave de la Industria 4.0 para predecir el comportamiento de sistemas complejos.

En el tercer artículo se propone un modelo para la detección de paráfrasis empleando un método de aprendizaje no supervisado con el que se extiende el umbral de recuperación de información de una Red Neuronal Recurrente de Hopfield. El modelo emplea, además, la entropía y la temperatura textual para la extracción de características. Esta propuesta se enmarca en la línea de Procesamiento de Lenguaje Natural como un aspecto que caracteriza la 4ta. Revolución Industrial. El resultado de esta investigación es un modelo de búsqueda de similitud semántica entre pares de enunciados que llega a encontrar más

de la mitad de la paráfrasis que puede existir en un corpus lingüístico. El modelo genera un Contexto de Afinidad Libre a partir del mínimo local que converge a través de una Red de Hopfield, que sirve para indicar y cuantificar la distribución semántica que pueda existir en un conjunto de documentos de texto. La utilidad teórica de este trabajo está estrechamente relacionada con la búsqueda y recuperación de información. Su utilidad práctica está dada por el empleo del modelo desarrollado para encontrar paráfrasis.

En el cuarto artículo los autores proponen un enfoque para extraer indicadores de desempeño del servicio al cliente integrando técnicas de inteligencia de negocios y procesamiento del lenguaje natural (NLP), con el propósito de aprovechar fuentes de datos no estructuradas. Para validar la propuesta los autores plantearon un caso de estudio en una empresa que ofrece servicios de salud. A partir de este caso obtuvieron indicadores que muestran la utilidad del enfoque para apoyar las relaciones con los clientes. El enfoque propuesto mostró cómo se puede aprovechar fuentes de datos estructuradas y no estructuradas para obtener indicadores que pueden ser útiles para apoyar las relaciones con los clientes. Apoyar esas relaciones es importante ya que, en la nueva revolución industrial, los clientes pueden convertirse en fuentes que generan valor agregado para las empresas.

El quinto artículo presenta el Internet de las Cosas como un paradigma habilitador de la Industria 4.0 bajo una propuesta para el mejoramiento del algoritmo ADR de tal forma que se optimice el consumo energético en redes LoRaWAN. Dentro de la propuesta se comparan diferentes modelos paramétricos y no paramétricos. Los resultados indican que los métodos basados en máquinas de vectores de soporte (SVM) y en redes neuronales artificiales presentan la mayor exactitud, con un porcentaje por encima del 90% en las estimaciones. Del conjunto de métodos evaluados, se encontró que las SVM son efectivas para la regresión de energía por paquete (EPP) y para ajustar la potencia de transmisión (TP_{final}), en tanto las redes neuronales son efectivas para clasificar el valor del número de bits por símbolo (SF_{final}).

El sexto artículo tiene como propósito el desarrollo de herramientas que ayuden a descubrir patrones ocultos en los datos, siendo un instrumento esencial para la toma de decisiones en lo que respecta a la Industria 4.0. Esta investigación propone una herramienta para la visualización y análisis de conjuntos de datos del paradigma de aprendizaje de múltiples instancias. La propuesta de visualización fue evaluada utilizando el criterio de expertos. Además, se realizaron diferentes pruebas que muestran que una correcta visualización puede ayudar a tomar decisiones sobre el conjunto de datos para mejorar la precisión de la clasificación. La investigación desarrollada permitió encontrar formas de representar estos datos y extraer información relevante para mejorar la precisión en el entrenamiento de un algoritmo MIL (*Multiple Instance Learning*). Además, posibilitó el desarrollo de la herramienta de visualización que, pese a sus debilidades, puede ser de utilidad a los investigadores del área.

El séptimo artículo se centra en el estudio afectivo del contexto social de un usuario, para enriquecer la recomendación de contenidos multimedia musicales más relevantes. La investigación presenta los diferentes componentes asociados al sistema de recomendación, tales como dataset de contenidos musicales, método computacional basado en un clasificador bayesiano encargado de predecir contenidos musicales a

partir del análisis del contexto social del usuario y servicio de música en línea. El estudio permitió realizar una investigación exhaustiva sobre librerías para la clasificación y descarga de contenidos musicales latinos, los cuales van a ser el insumo principal del sistema de recomendación a proponer. Además, la vinculación del análisis lírico para la clasificación por emoción permitió acotar y mejorar la clasificación por polaridad de todos los contenidos que conforman el dataset.

En el octavo artículo se presenta una aproximación a una arquitectura de referencia para plataformas de Big Data, la cual incluye dos componentes: los lineamientos que permiten guiar el proceso de implementación y una fase de gobierno de datos. La arquitectura de referencia para plataformas de servicios de datos pretende ser lo suficientemente general como para implementarse con diferentes tecnologías, paradigmas informáticos y software analítico, dependiendo de los requisitos y propósitos de cada organización. La arquitectura de referencia diseñada es la principal contribución de este trabajo, quedando como trabajo futuro la implementación y respectivas pruebas en algún tipo de industria y dentro de un contexto escalable.

Finalmente, el noveno artículo corresponde a un estudio exploratorio sobre la influencia de la Teoría de Belbin en actividades de gestión de proyectos. Para crear el contexto experimental con tareas de planificación y control, se diseñó un escenario ficticio en el que los sujetos experimentales planificaron y controlaron el desarrollo de un proyecto —construcción de fachadas. Los resultados del experimento mostraron, que, en cuanto a la estimación de costos y obtención de ganancias, los equipos integrados con base en la Teoría de Belbin (EC) no muestran diferencias con los equipos integrados aleatoriamente (ET); sin embargo, en lo que respecta a la variable “penalización”, los EC difieren significativamente de los ET, al haber entregado productos de mejor calidad. Los autores logran concluir, que el uso de la teoría de roles de Belbin para la integración de equipos de trabajo, en lo que respecta a la tarea de gestión, al parecer influye positivamente en los proyectos, al generar equipos más fiables. Como parte de los trabajos en curso, los autores han realizado un par de experimentos en el proceso de Diseño de Bases de Datos en curso que derivarán en resultados y conclusiones futuras.

Agradecimientos

Agradecemos al director de la Revista Álvaro Rocha por otorgar la oportunidad para dar a conocer los trabajos más relevantes presentados en el seminario SICC 2020. Asimismo, expresamos nuestra gratitud a los autores de los trabajos publicados, lo cual hizo posible este número en RISTI. Aunando esfuerzos queremos agradecer a las empresas y universidades aliadas que con su aporte lograron engrandecer la Décimo Tercera Edición del Seminario Internacional de Ciencias de la Computación.

Referencias

Gasca-Hurtado, G. P., & Machuca-Villegas, L. (2019). Era de la cuarta revolución industrial. RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação, (34), xi-xv.

- Havle, C. A., & Üçler, Ç. (2018). Enablers for industry 4.0. In 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (pp. 1–6), Ankara, Turkey.
- Kumar, R. (2020). Espousal of Industry 4.0 in Indian Manufacturing Organizations: Analysis of Enablers. In *Handbook of Research on Engineering Innovations and Technology Management in Organizations* (pp. 245–252). IGI Global.
- Mogos, F. M., Eleftheriadis, R. J., & Myklebust, O. (2019). Enablers and inhibitors of Industry 4.0: results from a survey of industrial companies in Norway. *Procedia CIRP*, 81, 624-629.
- Sanchez, D. O. M. (2019). Sustainable Development Challenges and Risks of Industry 4.0: A literature review. In *2019 Global IoT Summit (GIIoTS)* (pp. 1–6).

Índice / Index

EDITORIAL

El desarrollo de software como habilitador digital para la 4ta revolución industrial	xi
<i>Gloria Piedad Gasca-Hurtado, Liliana Machuca-Villegas</i>	

ARTIGOS / ARTICULOS / ARTICLES

Estrategia de transformación para la formación en informática: hacia el desarrollo de competencias en educación básica y media para la Industria 4.0 en Medellín – Colombia	1
<i>Bell Manrique-Losada, María Clara Gómez-Álvarez, Liliana González-Palacio</i>	
Conversión de eventos desde esquemas preconceptuales en código PL/pgSQL: simulación de software en la cuarta revolución industrial	18
<i>Carlos Mario Zapata-Jaramillo, Juan Sebastián Zapata-Tamayo, Paola Andrea Noreña Cardona</i>	
Detección de paráfrasis basada en la energía, entropía y temperatura textual	35
<i>Alberto Iturbe-Herrera, Armando Rojas-Valdez, Noé Alejandro Castro-Sánchez, Gerardo Sierra</i>	
Un enfoque de Análisis Inteligente de Datos para Apoyar la Relación con los Clientes	52
<i>Oswaldo Solarte Pabón, José Heriberto Torres, Víctor Andrés Bucheli</i>	
Mejoramiento del algoritmo ADR en una red de Internet de las Cosas LoRaWAN usando Aprendizaje de Máquina	67
<i>Mauricio González-Palacio, Lina María Sepúlveda-Cano, Jhon Quiza-Montealegre, Juan D'Amato</i>	
Visualización de conjuntos de datos de múltiples instancias	84
<i>Jorge Eliecer Valencia-Duque, Carlos Mera, Lina Maria Sepúlveda</i>	
Sistema de recomendación para contenidos musicales basado en el análisis afectivo del contexto social	100
<i>Mauricio Sánchez Barragán, Gabriel Elías Chanchí G, Wilmar Yesid Campo M</i>	
Creación colaborativa de una arquitectura de referencia para la implementación de plataformas de servicios de datos	114
<i>Edisson Estelio Gutiérrez Jiménez, Juan Bernardo Quintero, Bell Manrique Losada</i>	
Influencia de la Teoría de Roles en actividades de Gestión: Un experimento controlado con estudiantes de Ingeniería de Software	131
<i>Raúl Aguilar Vera, Omar Gómez Gómez, Julio Díaz Mendoza, Juan Ucán Pech</i>	

Estrategia de transformación para la formación en informática: hacia el desarrollo de competencias en educación básica y media para la Industria 4.0 en Medellín – Colombia

Bell Manrique-Losada¹, María Clara Gómez-Álvarez¹, Liliana González-Palacio¹

bmanrique@udem.edu.co; mcgomez@udem.edu.co; ligonzalez@udem.edu.co

¹ Universidad de Medellín, Carrera 87 No. 30-65, 50026, Medellín, Colombia.

DOI: 10.17013/risti.39.1-17

Resumen: La informática ha logrado avances sustanciales en la sociedad, en razón al rol transformador que tiene la tecnología como eje de desarrollo. A pesar que la educación también se ha visto permeada por las transformaciones de la tecnología y la evolución de la informática, es muy baja la afectación y transformación de los currículos, especialmente de básica y media. Así, en países como Colombia la formación en informática sigue concibiéndose como una asignatura complementaria a otras áreas que busca la formación de consumidores de tecnología, más que la de creadores de ella. En este trabajo se propone una estrategia de transformación para la formación en informática en instituciones de educación básica y media, orientada hacia el desarrollo de competencias para la Industria 4.0, centradas en el pensamiento creativo y computacional. Esta propuesta se compone de dos ejes fundamentales: diseño curricular y diseño didáctico, así como de un ecosistema de formación.

Palabras-clave: pensamiento computacional; pensamiento creativo; formación en informática; diseño curricular; ecosistemas de formación.

Transformation strategy for computer training: towards the skills development in basic and secondary education for Industry 4.0 in Medellín-Colombia

Abstract: Computing has made substantial progress in society, due to the transformative role that technology has as the axis of development. Even though education has also been permeated by the transformations of technology and the evolution of computer science, the affectation and curricula transformation is very low, especially in basic and secondary levels. Thus, in Colombia computer training continues being conceived as a complementary subject to other areas that seek the training of technology consumers rather than its creators. This work proposes a transformation strategy for computer training in primary and secondary education institutions, oriented towards the development of competences for Industry 4.0, focused on creative and computational thinking. This proposal has two key

elements: curricular design, and didactic design as well as a Training Ecosystem in institutions at these levels.

Keywords: computational thinking; creative thinking; computer training; curriculum design; training ecosystems.

1. Introducción

La informática ha logrado hoy en día avances inimaginables en todos los ámbitos de la sociedad, en razón al rol transformador que tiene la tecnología con la adaptación a nuestro entorno y la satisfacción de las necesidades personales, sociales y culturales. Evidentemente, la educación también se ha visto afectada por el cúmulo de transformaciones producidas por la inclusión de la tecnología en estos ámbitos que han generado a su vez cambios en los modelos pedagógicos, en el rol y perfil de los usuarios que se forman, y en los escenarios donde acontece el proceso de enseñanza-aprendizaje, entre otros. A pesar de este tipo de afectaciones en la educación, propiciada por la evolución de la tecnología y la informática, así como por la implementación de teorías pedagógicas como el conectivismo (Goldie, 2016) que surgen para explicar cómo aprendemos en los nuevos contextos tecnológicos actuales, se reconocen debilidades como una baja afectación y rediseño de los currículos (Cortés, 2016), sin orientación hacia el desarrollo del pensamiento creativo y computacional, y la falta de formación e infraestructura adecuada (Buxarrais & Ovide, 2011) por parte de los actores involucrados.

En este sentido y por razones como las descritas, hoy en día la informática, especialmente en los niveles de básica y media (ByM), sigue concibiéndose como una asignatura complementaria a otras áreas que busca la formación de usuarios o consumidores de tecnología, más que de adoptadores o creadores de ella. Los estudiantes usan y conocen muchos recursos y herramientas tecnológicas, pero no saben utilizarlas para la apropiación de conocimiento (Avendaño, 2015).

La ventaja de lograr una formación pertinente en informática desde ByM, fundamentada en el desarrollo del pensamiento creativo y computacional es entonces evidente: permite la transformación de una sociedad formada por meros consumidores de tecnología, en una de potenciales desarrolladores de ésta. Si se orienta adecuadamente la formación en nuevas tecnologías, se puede concebir un sistema educativo reformado e innovador. Experiencias de desarrollo del pensamiento creativo y computacional (PCC), como lo expresa Cortés (2016), logran que los estudiantes despierten su curiosidad e interés por la ciencia, generando nuevas formas de aprender y solucionar problemas del mundo real.

En los últimos años se ha renovado el interés en la enseñanza de pensamiento computacional y programación alrededor del mundo, como una forma de ampliar las oportunidades para los jóvenes y aumentar la competitividad de los países. Desafortunadamente, Colombia se ha mantenido relegada en relación a los países líderes en estos temas como Inglaterra e Israel, y ha persistido en políticas de educación en informática que privilegian la enseñanza en el uso de aplicaciones (Riesco *et al.*, 2014). Actualmente, impulsada por los Ministerios de Educación y de las TIC, se están desarrollando algunas iniciativas relevantes en torno a estos temas, las cuales deben ser fortalecidas y apalancadas en los ámbitos regionales.

En este trabajo se propone una estrategia de transformación de la formación en informática en instituciones de educación ByM en Medellín y su área metropolitana, como una iniciativa local orientada hacia el desarrollo de competencias centradas en el PCC. Esta propuesta se presenta como una alternativa de formación en informática a partir de la educación inicial, con dos ejes articuladores: programación y robótica. Los beneficiarios directos son administradores educativos y profesores, dado que la propuesta facilita las acciones de diseño curricular/didáctico y su aplicación práctica en el aula. Asimismo, se define un marco de trabajo que facilita el desarrollo de un *Ecosistema de Formación* en las instituciones de educación ByM que estén incursionando en procesos de renovación curricular.

El resto del artículo se organiza de la siguiente manera: en la primera sección se presenta la fundamentación conceptual de la investigación. En la segunda sección se muestra un análisis de los trabajos relacionados identificados en el estado del arte. En la siguiente sección se presenta la estrategia propuesta para solucionar el problema y se describen detalles del proceso de implementación y validación. Finalmente, se presentan los resultados de la evaluación de la estrategia y las conclusiones derivadas del presente trabajo.

2. Fundamentación conceptual

En esta sección se abordan los conceptos que fundamentan la investigación de la cual se deriva el presente artículo. Se inicia con la definición de enseñanza-aprendizaje, luego se describe el enfoque conceptual asociado a Diseño Curricular, y finalmente los fundamentos de la formación en informática.

2.1. Enseñanza y aprendizaje

La enseñanza y el aprendizaje forman parte de un proceso unitario que tiene como fin la formación del estudiante (Infante & Miranda, 2017). Es así como convergen la función mediadora del profesor y la acción de apropiación del estudiante.

La enseñanza comprende la transmisión de información por parte del docente mediante una comunicación concreta, que genera como resultado un cambio en el estudiante en forma de conocimientos, habilidades y/o capacidades (Sánchez, 2006). El aprendizaje se centra en la adquisición de un nuevo conocimiento, habilidad o capacidad y se considera efectivo cuando se manifiesta en un tiempo futuro y favorece la solución de problemas (Sánchez, 2006).

El proceso de enseñanza-aprendizaje parte de la interacción/comunicación en el aula o en el espacio donde se está dando el acto educativo, por ejemplo, una plataforma virtual. Además, se enmarca en unas coordenadas espacio-temporales, y se orienta por unos lineamientos metodológicos. Se trata de un fenómeno complejo, en razón a que intervienen un conjunto de factores y características especiales, como lo sugiere Núñez et al. (1998): las diferencias individuales de los actores del proceso y los tipos de interacción que estas diferencias suponen, y el conocimiento y características de los contenidos teóricos o procedimentales.

2.2. Diseño curricular

El término ‘Diseño Curricular’ es empleado para referirse a la estructuración de planes de estudio (a partir de contenidos curriculares), programas e incluso a la instrumentación didáctica del proceso de formación o proceso de enseñanza-aprendizaje (Álava et al., 2019). Los contenidos curriculares pueden ser de diversos tipos: de unidad, de materia o asignatura, de área, de etapa escolar, entre otros. Así, el proceso de diseño curricular se formaliza en un documento que puede reconocerse como un plan instruccional, programa de un curso, programa de área, plan de estudios. Por su parte la estructura curricular determina las etapas de diseño relativas al perfil, objetivos, contenidos y metodología, siguiendo clasificaciones estándar para este diseño, como la taxonomía de Bloom.

Taxonomía de Bloom

Como se mencionó anteriormente, en el acto educativo hay dos procesos indisolubles: la enseñanza y el aprendizaje. En éste último, es fundamental establecer y detallar los objetivos de aprendizaje. Para apoyar esta labor, Benjamín Bloom propuso desde 1950 una taxonomía que permite categorizar a partir de un conjunto de verbos las nuevas habilidades y conocimientos que un estudiante debe adquirir al apropiarse de algún concepto, o área de conocimiento. A lo largo de décadas esta taxonomía ha evolucionado, hasta convertirse en uno de los principales referentes para la definición de competencias (Anderson & Krathwohl, 2001).

Didácticas y estrategias de enseñanza y aprendizaje

La planificación o el diseño del proceso formativo debe ser explicado en el marco de los planteamientos de la pedagogía y la didáctica. Un lugar relevante lo ocupa también la parte aplicada, que se reconoce en función de la metodología, con sus estrategias y técnicas, así: la didáctica es una disciplina pedagógica que estudia los elementos constitutivos de los procesos de enseñanza y aprendizaje (Zambrano, 2005). De otro lado, la metodología hace referencia a la forma de materializar el acto educativo. En términos de la planificación curricular, incluye la forma de enseñar (método), el papel del estudiante (activo, pasivo), las funciones del profesor en el proceso de construcción del conocimiento, las técnicas para desarrollar las tareas, los tiempos, espacios y materiales necesarios y las competencias que se deben lograr (Zambrano, 2005).

2.3. Formación en informática

Esta formación busca entender y tener éxito en la solución de problemas de la vida real. Si bien existen diversas aproximaciones a las TIC en los currículos de niveles básico y medio, el desarrollo de esta área no se ve plasmado de manera adecuada, en la mayoría de los casos debido a que no se concibe como una asignatura con entidad propia, o a su orientación se fundamenta hacia el uso de aplicaciones.

Formación en informática a nivel mundial

Inglaterra fue el primer país (desde el 2014) en incluir de manera obligatoria en el currículo de los niveles ByM (entre 5 y 16 años), asignaturas de informática (UK, 2013). Países como Canadá, Israel y Japón lideran la clasificación de la OCDE donde hay mejores

niveles de educación (OCDE, 2013). Canadá ha definido en su normatividad curricular desde ByM una sección destinada de manera explícita hacia la informática (OME, 2008). Israel tiene definiciones curriculares para ByM de cursos de informática optativos, como fundamentos de algoritmos, programación y diseño de software (Hazzan, Gal-Ezer, & Blum, 2008). Además, tiene incluidos programas de gobierno para preparación de profesores y una línea de investigación en educación en ciencias de la computación. En Japón, se tiene desde secundaria asignaturas obligatorias para desarrollar competencias y capacidades de resolución de problemas y ciudadanía digital (Matsuda, 2010).

En Estados Unidos, las disposiciones para formación y diseño curricular se regulan en cada estado. En la mayoría no se enseña informática o son cursos optativos que no forman parte del núcleo curricular. Dada esta debilidad, desde el 2016 y bajo la iniciativa *Computer Science for All* (Ciencias de la Computación para todos) liderada por el expresidente Barack Obama, se logró promover la expansión de su enseñanza en las escuelas. Por otro lado, se han liderado movimientos privados como Code.org, que buscan introducir la enseñanza de informática en las escuelas y hogares con el apoyo personalidades reconocidas como Bill Gates de Microsoft o Mark Zuckerberg de Facebook.

En América Latina, y de acuerdo al estudio de políticas y prácticas para la enseñanza de las ciencias de la computación en América Latina (Jara, Rodríguez & Claro, 2019), Costa Rica instituyó una asignatura de informática educativa hace treinta años, basada en la enseñanza de programación para niños de primaria, lo cual se amplió luego a secundaria. En la actualidad, esta política está orientada hacia desarrollar el pensamiento computacional mediante la programación, robótica y *Makers*.

Formación en informática en Colombia

En Colombia se establecen por medio de las Leyes 115 de 1994 y 30 de 1992 el contexto y la estructura del servicio educativo en dos tipos: educación formal y no formal, conocida como educación para el trabajo. La educación formal, para el alcance de este artículo, se clasifica en: nivel preescolar (mínimo un grado obligatorio), nivel básico con una duración de nueve grados (básica primaria: cinco grados y básica secundaria: cuatro grados), y nivel medio con una duración de 2 grados. Al final de este ciclo se logra un título de bachiller académico, o en otros casos, bachiller técnico en algún área específica.

En cuanto al área de informática, si bien existen algunos esfuerzos aislados por incorporar fundamentos de programación en el currículo en la educación formal ByM, en la mayoría de instituciones educativas la asignatura de tecnología tiene como contenido básico el uso de herramientas de ofimática. Uno de las experiencias más relevantes de la incorporación de competencias en informática en la educación media, la constituye la incorporación de media-técnica en desarrollo de software, como actividad extracurricular que permite a los estudiantes titularse como bachilleres técnicos en esta área. A la fecha, aproximadamente 110 colegios públicos de la ciudad cuentan con esta modalidad como resultado de trabajo de la Alianza Futuro Digital Medellín (Henao, 2011).

Competencias en Pensamiento Creativo y computacional

Actualmente, para cualquier individuo de la sociedad es evidente que el manejo de tecnologías digitales es una habilidad básica y en algunos innata (nativos digitales).

La expansión del uso y acceso de Internet en todos los entornos sociales, culturales y económicos, la amplia oferta de aplicaciones y el uso masivo de dispositivos inteligentes (como teléfonos móviles) han convertido la informática en un elemento indispensable en la vida cotidiana. Dadas estas condiciones, se ha generado un cambio en el rol del usuario, que, como afirman Riesco et al. (2014) ha dejado de ser un mero usuario de herramientas para convertirse en un *agente* que debe modificarlas y adaptarlas a sus necesidades.

De acuerdo a la ISTE (*International Society for Technology in Education*) el Pensamiento Computacional es un proceso de análisis y solución de problemas que incluye características asociadas a: formular problemas de una forma que permita usar ordenadores y otras herramientas para solucionarlos; organizar y analizar datos de manera lógica; representar datos mediante abstracciones; y automatizar soluciones mediante pasos ordenados (pensamiento algorítmico), entre otros (CSTA & ISTE, 2011).

El pensamiento creativo se concibe como una alternativa privilegiada para la transformación de las condiciones sociales y culturales que se requiere enfrentar en la actualidad y que tienden a complejizarse día tras día (Velásquez, 2017). Así, se sugiere que la resolución de problemas pueda ser modelada por una máquina que procesa información (computador), promoviendo la creatividad de los estudiantes y logrando vincular la teoría computacional y las aplicaciones en el mundo real.

3. Trabajos relacionados y Antecedentes

A continuación, se presentan algunas experiencias exitosas en Latinoamérica, Colombia y el mundo, orientadas a la formación en tecnologías abarcando PCC, mediante los ejes articuladores robótica y programación.

3.1. Colombia - Latinoamérica

Barrera & Montaña (2015) plantean una propuesta para fomentar el pensamiento computacional en la educación media en Chile fundamentada en el desarrollo de recursos para promover competencias lógicas de programación utilizando Scratch. Esta propuesta consta de sesiones diseñadas con objetivos de aprendizaje y actividades de PCC presentes en cada sesión. Sin embargo, no se incorpora al currículo de informática de instituciones de educación ByM, sino que se aborda como un taller complementario y voluntario para estudiantes de séptimo y octavo grado por un período de dos meses.

En este mismo sentido también se encuentra el trabajo de Basogain, Olabe & Olabe (2015) en República Dominicana, quienes desarrollaron durante 6-10 meses un curso de pensamiento computacional mediante la plataforma Egelapi a niños de educación primaria entre 6-12 años, donde los resultados de los exámenes y entrevistas realizadas muestran alto nivel de desempeño y satisfacción de los estudiantes.

A nivel Colombia, se identifican dos experiencias similares: (i) Incorporación del pensamiento computacional en curso de primaria con 66 alumnos entre 7-10 años dividiendo a los alumnos en tres grupos (básico, intermedio y avanzado). En este curso se realizaron pruebas al iniciar y finalizar, evidenciando un aumento del 6.56% en el nivel avanzado (Rodríguez, 2017). (ii) Incorporación de pensamiento computacional en la asignatura Matemáticas mediada por Scratch en niños entre 10-11 años, donde se

comparó el desempeño de un grupo experimental y un grupo de control obteniendo resultados superiores (Galindo Suárez, 2016).

También se reconocen diversas experiencias del uso de robótica para la enseñanza de computación en educación ByM. En Costa Rica, Vega-Moreno, Cufí Solé, Rueda & Llinás (2016) describen el proyecto EDUROVs para fomentar el aprendizaje basado en proyectos de robótica promoviendo la creatividad, el uso de nuevos materiales y el desarrollo de habilidades de programación a bajo costo. En el caso de Colombia, entre otros, Vargas, Guapacho & Isaza (2017) presentan una propuesta para el desarrollo de competencias cognitivas en matemáticas y física mediante el uso de la robótica educativa como elemento integrador de las áreas STEM. Esta propuesta incluye el diseño de un plan de estudios basado en objetivos, actividades, materiales requeridos y métodos de enseñanza-aprendizaje, el uso de diferentes aplicaciones robóticas junto con software interactivo, desde grado 5° a 11°.

3.2. Mundo

En relación a la enseñanza de pensamiento computacional, se encuentran trabajos como: Morrás (2014) en Escocia, forman en fundamentos de programación por 8 semanas a un grupo de 21 alumnos entre 8-9 años de educación primaria. Al evaluar a los estudiantes por medio de rúbricas se obtuvieron resultados satisfactorios en términos de mejora cognitiva con respecto a la clase tradicional. En España, Moreno-León, Robles & Román-González (2015) incorporan pensamiento computacional en un grupo de primaria y otro de secundaria mediada por Scratch. La percepción final de los participantes la reconocen como una herramienta atractiva y de fácil manejo y generó una mejora en las competencias de programación. En Estados Unidos, autores como Yadav, Gretter, Hambruch & Sands (2016) plantean la importancia de incorporar la enseñanza de ciencias de la computación en la educación ByM (K-12), lo que implica mayores esfuerzos en la formación de docentes con estas competencias, cuyos principales retos son: (1) Los licenciados poseen conocimientos en didáctica y pedagogía, pero su formación y experiencia en disciplinas como programación es escasa, (2) Falta de conocimiento de métodos de evaluación del aprendizaje logrado en los estudiantes y (3) Falta de comunidades de práctica o colaboración entre los docentes de educación ByM.

A nivel mundial, también se viene utilizando la robótica como eje de formación en competencias tecnológicas de Industria 4.0, como es el caso de Frieboon-Yesharim & Ben-Ari (2018), quienes proponen esta formación aprovechando la motivación que genera en los estudiantes el trabajo con objetos tangibles como los robots. Estos autores plantearon un curso de robótica para 118 estudiantes con edades entre 7-8 años usando un kit de robótica y una herramienta gráfica de software.

4. Estrategia de transformación

El ecosistema actual de los escenarios de enseñanza evidencia un panorama crítico, caracterizado por la diferencia entre las competencias definidas en los currículos de las instituciones, la formación rígida, fragmentada y orientada a corto plazo (Said, 2015), y los desafíos que los estudiantes deben asumir para el mundo competitivo, interconectado y globalizado (Schmidt, 2010). En este contexto se deben promover

modelos de enseñanza y aprendizaje continuos y flexibles, en los que se estimule el desarrollo de competencias que respondan a las necesidades reales. A partir de esto, el desarrollo de iniciativas que partan desde el diseño curricular, e involucren formación de actores, cobran gran relevancia en el proceso de desarrollo de nuevos modelos para mejorar las condiciones de enseñanza desde los niveles ByM.

La transformación de los procesos de formación en las instituciones educativas puede ser abordada, como lo propone Cortés (2016) desde lo macro a lo micro (cuando los cambios son orientados desde entes del gobierno como Secretarías de Educación o Ministerio de Educación -MEN) o de lo micro a lo macro (donde se consideran las experiencias llevadas a cabo por los docentes/directivos y las condiciones de cada institución, para ser extrapoladas a contextos más amplios), lo cual supone, según la UNESCO (López, Lugo & Toranzos, 2014), identificar el alcance de cada institución para tomar decisiones y realizar planes de acción. Es desde este último abordaje que se ha definido la propuesta que se presenta aquí.

En este apartado se describe la estrategia de transformación propuesta que favorece el papel protagónico de la formación en informática a partir de la educación inicial, centrada en el desarrollo del PCC. Esta estrategia se estructura en dos componentes: Propuesta de diseño didáctico-curricular y Desarrollo del Ecosistema de Formación.

4.1. Propuesta de diseño didáctico-curricular

Los procesos de diseño curricular parten de marcos conceptuales y políticas pedagógicas de referencia, orientados hacia la mejora del aprendizaje de los estudiantes. Dado que todos los contenidos no pueden ser tratados de la misma manera, ni con el mismo rigor, abordaje, ni profundidad, depende en gran medida de las mediaciones pedagógicas de los profesores, de los conocimientos que se quiera impartir, de las formas de gestionar el aprendizaje, de los recursos disponibles, y de la autonomía escolar. En este sentido es importante determinar dos ejes integradores del proceso: lo curricular y lo didáctico.

Diseño Curricular

Bajo el eje de Diseño Curricular se encierra un conjunto de elementos referidos a planes de estudio, programas y cualquier instrumentación didáctica del proceso de formación o proceso de enseñanza-aprendizaje (Álava *et al.*, 2019). En la presente propuesta, este último elemento se aborda en el eje de Diseño Didáctico.

Para el área de formación en informática, de acuerdo a la SEI (2017), en el Diseño Curricular es necesario considerar las competencias en ciencias, tecnología, ingeniería, artes y matemáticas (STEAM, por sus siglas en inglés para *Science, Technology, Engineering, Arts and Mathematics*), alineado con las necesidades que requiere la Industria 4.0 (alto nivel de cualificación en áreas tecnológicas clave). Además, considerar las competencias transversales asociadas con: alta capacidad de adaptación, flexibilidad y aprendizaje continuo y perfiles multidisciplinarios. En este sentido, las competencias genéricas que permean toda la estructura para este diseño curricular se orientan hacia el PCC, con dos ejes articuladores: la programación y la robótica.

Como parte de este componente, en la investigación realizada soportada en la literatura analizada, se define la estructura para el diseño curricular que se presenta en la Tabla 1, en términos de *tipo de habilidad* y *categoría de competencia* a desarrollar desde las áreas del plan de estudio. En la Tabla 2 se presenta la alineación de las categorías de competencia propuestas, con su correspondiente categoría en la Guía 30 del MEN, como estándar de referencia nacional, y con el Framework curricular de Massachusetts DLCS (*Digital Literacy and Computer Science*) como estándar de referencia internacional.

Habilidad		Categoría de Competencia
Tipo	Descripción	
General o mediacional	Agrupar habilidades de comprensión general que faciliten una actitud positiva y motivación para abordar soluciones.	1. Apropiación de las TIC Conocimiento, comprensión y uso de dispositivos de cómputo y contenidos digitales (software, documentos digitales).
Cognitiva	Conjunto de operaciones mentales (atención, percepción, memoria, resolución de problemas, etc.) que permiten que el estudiante integre la información adquirida a través de los sentidos, en una estructura de conocimiento con sentido.	2. Pensamiento computacional Permite desarrollar una solución genérica a un problema, a partir de su descomposición, identificación de variables relevantes y patrones, y la derivación de un procedimiento de solución (algoritmo).
Práctica	Habilidad para la ejecución práctica o desempeño, que implica destrezas motoras y habilidades prácticas o de ejecución. El saber se concretiza en una práctica estratégica.	3. Codificación Integra las habilidades prácticas para escribir y diseñar programas software como aplicaciones de cómputo funcionales
Tecnológico-Práctica	Además de la habilidad práctica base, vincula conocimientos específicos de ejecución y uso de tecnologías	4. Construcción Diseño, desarrollo y puesta en marcha de un producto tangible, que involucre la conexión codificación-hardware.

Tabla 1 – Estructura propuesta para el diseño curricular

Tipo de Habilidad	Categoría de Competencia	Alineación con Estándar	
		Guía 30	DLCS
General o mediacional	1. Apropiación de las TIC	- Naturaleza y evolución de la tecnología - Apropiación y uso de tecnología	<i>Computing and Society</i> <i>Digital Tools and Collaboration</i>
Cognitiva	2. Pensamiento computacional	Solución de problemas con tecnología	<i>Computational Thinking</i>
Práctica	3. Codificación	Solución de problemas con tecnología	Computational Thinking
Tecnológico-Práctica	4. Construcción	Tecnología y sociedad	<i>Computing Systems</i>

Tabla 2 – Estructura de diseño curricular alineado con estándar internacional

Diseño Didáctico

El diseño didáctico está orientado hacia la generación de procesos innovadores de especificación didáctica, en términos de estrategias, actividades, herramientas y recursos de apoyo, que permita poner en marcha el diseño curricular. Para tal fin, el docente debe definir acciones que le permitan guiar a los estudiantes; facilitarles el uso de los recursos y las herramientas que necesitan para explorar y elaborar nuevos conocimientos y destrezas; actuar como orientador y mediador, pero, principalmente, como gestor de los recursos de aprendizaje que fortalezca el papel activo de los estudiantes en el desarrollo de los contenidos (Armengol Casas, 2005).

En la Tabla 3 se presenta un primer abordaje para la especificación de actividades en el marco de estrategias de enseñanza, como propuesta de la estructura de diseño didáctico, que incluye los elementos que proponemos deben considerarse para el diseño didáctico de actividades para el área de tecnología-informática.

Especificación didáctica		
Nombre Actividad:		
Nivel/Grado:	Autor:	Fecha:
Estrategia:		
Categoría	Unidad:	Temática:
1. Descripción General de Actividad:		
2. Objetivos de Aprendizaje: 2.1) competencias específicas; 2.2) competencias generales		
3. Organización y gestión del equipo: 3.1) tamaño del grupo para la actividad propuesta; 3.2) Organización y gestión del grupo: indicar roles en cada equipo y responsabilidades; 3.3) Número de grupos aproximados del tamaño ya indicado.		
4. Recursos: 4.1) Presupuesto; 4.2) materiales (reusables, Consumibles); 4.3) Personal; 4.4) espacios (de diseño, construcción, almacenamiento, operación); 4.5) Software de apoyo		
5. Información para estudiantes: 5.1) Contexto/Antecedentes; 5.2) descripción de la situación problemática; 5.3) desarrollo del ejercicio (pasos de la actividad).		
6. Información adicional: 6.1 Preguntas de reflexión y discusión con los estudiantes; 6.2) alternativas para el ejercicio; 6.3) lecturas y recursos suplementarios.		
7. Bibliografía de referencia		

Tabla 3 – Estructura para la especificación didáctica

4.2. Desarrollo del Ecosistema de Formación

Al hablar de ‘ecosistema’, por la naturaleza del término, se hace referencia a una comunidad de seres o individuos cuyos procesos vitales se relacionan entre sí y se desarrollan en función de los factores físicos de un mismo ambiente. Con este fundamento, un ‘ecosistema de formación’ define la promoción de redes de colaboración entre estudiantes, profesores y profesionales para el desarrollo de aprendizajes en un contexto específico (presencial y virtual) atendiendo a algunas condiciones iniciales

(Álvarez & Rodríguez, 2015). En este sentido, integra componentes físicos y humanos, relaciones e interacciones entre ellos (de poder, liderazgo, creatividad, emprendimiento) y un contexto.

Entre algunas propuestas de ecosistemas de formación, se puede resaltar el ecosistema de aprendizaje de Guetl & Chang (2008), y el EcosisteMap como modelo para aprender a emprender (Álvarez & Rodríguez, 2015), los cuales incorporan en un ecosistema diseño de aprendizaje, recursos humanos, formación para el desarrollo de competencias, sistema de comunicación y diferentes aplicaciones.

Dadas las posibilidades que ofrecen los nuevos escenarios educativos, las instituciones que quieran asegurar su actualidad y continuidad deben desarrollar y hacer parte de comunidades y sistemas de formación permanente, donde se propenda por el desarrollo de habilidades clave como: creatividad, emprendimiento, trabajo en equipo, responsabilidad social, pensamiento crítico y creativo y aprendizaje para toda la vida, que permitan el desarrollo de competencias en el área. Autores como Álvarez & Rodríguez (2019) reconocen las limitaciones y complejidades que conllevan este tipo de intervenciones, porque la enseñanza, el aprendizaje y el currículo ya no son patrimonio de ninguna institución en concreto. Sin embargo, proponer y adoptar iniciativas con una perspectiva sistémica permite que participen de manera organizada los agentes involucrados en la formación en informática en los niveles de ByM.

En este sentido, se propone un marco de trabajo que busca facilitar el desarrollo de un *Ecosistema de Formación* en las instituciones de educación básica y media que estén incursionando en procesos de renovación curricular de la línea de formación en informática, buscando estructurar sus líneas de formación hacia las necesidades actuales del medio y las demandas de la Industria 4.0. Estas demandas deben rescatar aspectos que incorporan la nueva forma de trabajar en esta era, relacionados con la comunicación, la formación, la creación de plataformas en entornos colaborativos, la financiación en innovación y en I+D, y la ciberseguridad, entre otros.

Unidades del Ecosistema

Las unidades del ecosistema se presentan en la Figura 1, y se describen a continuación:



Figura 1 – Unidades del ecosistema de formación

- Infraestructura y Talento Humano, que comprende tres aspectos:
 - Definición e implementación de espacios, como laboratorios, aulas dedicadas y espacios de aprendizaje (presenciales y virtuales).
 - Configuración de aspectos de conectividad y plataformas de soporte.
 - Especificación del talento humano de soporte: profesores, tutores, monitores, directivos, staff de gobierno, entre otros.
- *Soporte*, que involucra la definición de acciones para lograr la sostenibilidad, continuidad y mejora continua de las iniciativas de intervención en las instituciones, priorizando:
 - Estructuración de una red de trabajo con los agentes del ecosistema.
 - Medición de indicadores de avance en intervención (estrategia propuesta).
 - Desarrollo de espacios de co-working para actividades de enseñanza, colaboración, intercambios, o implementaciones, entre otros.
 - Desarrollo de estrategias de motivación (como concursos, premios, juegos, eventos, competencias entre instituciones).
- *Adopción y Uso*:
 - Especificación y adopción de contenidos.
 - Montaje y adopción de aplicaciones de cómputo y plataformas de soporte.
 - Implementación del programa de formación y de acciones de sensibilización/ concienciación entre los actores de cada institución.

Agentes del Ecosistema

Los agentes del ecosistema son todos aquellos actores que intervienen en el Ecosistema de Formación. Estos se visualizan en la Figura 2 y son:

- Organizaciones: Instituciones educativas, con el apoyo de empresas y grupos de interés académico (aliados estratégicos, clusters, autoridades del orden local, regional y nacional), cuyas responsabilidades se fundamentan en sustentar el compromiso para la ejecución de las intervenciones, y el aseguramiento del soporte y recursos básicos para su correcta implementación.
- Comunidades: Proporcionan recursos de apoyo, como el acceso a redes de mentores, inversionistas, proveedores y clientes.
- Promotor: Foco del ecosistema que tiene como meta liderar y visionar planes estratégicos para la implementación de la estrategia y la generación de los productos de cada intervención (de diseño curricular, tecnológicos, puestos de trabajo y otros beneficios económicos para los actores involucrados).



Figura 2 – Agentes del Ecosistema de Formación

5. Evaluación

El proceso de implementación de la estrategia propuesta es el resultado de un proyecto de investigación entre la Universidad de Medellín y un colegio privado de la ciudad de Medellín. A continuación, se resumen las actividades de evaluación.

A partir de la revisión de referentes, se construyó una propuesta preliminar de contenidos (diseño curricular) para la asignatura de informática desde pre-escolar hasta grado undécimo. Esta propuesta fue presentada en una sesión de expertos, donde participaron aproximadamente 35 profesores y coordinadores de informática de colegios públicos y privados de la ciudad de Medellín. Con el uso de herramientas de co-creación, se trabajó en equipos sobre los contenidos presentados por grados con la posibilidad de ajustarlos, cambiar el orden, proponer otros nuevos, entre otros. Como resultado de esta actividad, se generó una nueva versión del diseño curricular, basada no sólo en referentes, sino en la experiencia y el conocimiento del contexto de los participantes.

Se diseñaron y aplicaron talleres, producto del diseño didáctico, para grupos focales en tres grados, así: (i) *Transición*, a tres grupos de 18 niñas (4-5 años), con el fin de promover el seguimiento de instrucciones en un proceso mediante una actividad lúdica; (ii) *Cuarto*, a tres grupos de 30 estudiantes (10-11 años) con el propósito de definir algoritmos en pseudocódigo a través de un lenguaje de alto nivel; y (iii) *Octavo*, a tres grupos de 30 estudiantes (12-14 años) con el fin de implementar algoritmos de complejidad creciente para resolver problemas concretos de ordenamiento.

En el caso de las estudiantes de octavo grado, al finalizar los talleres diligenciaron una encuesta orientada a conocer su percepción sobre el taller y el aprendizaje logrado. En cuanto a ítems como la calidad del material usado, el 100% de los participantes la califican como Muy buena (máxima calificación), el 83% califican la coordinación de la actividad por parte del profesor como Muy Buena y el 75% como Excelente. Frente a la pregunta, ¿cuál es la estrategia a seguir para ganar?, algunas de las respuestas fueron: “aplicar lógica para el seguimiento de instrucciones”, “atención y coordinación en el planteamiento de la solución”. Ahora, frente la pregunta ¿cuáles son los principales conceptos aprendidos? algunas respuestas fueron: “seguimiento de instrucciones”, “programación y algoritmos” y “aprendimos sobre la manera en cómo se programan algunos robots”.

Finalmente, como última actividad de esta etapa, se realizó un evento de cierre del proyecto, con 41 docentes y coordinadores de informática de colegios de la ciudad de Medellín. Este evento se realizó con el propósito de compartir la versión mejorada del diseño curricular, describir algunos ejemplos de la especificación didáctica, sintetizar los resultados de los talleres piloto y dar el primer paso en la conformación del ecosistema de formación con los actores involucrados.

6. Conclusiones

Es necesario resaltar que las demandas actuales de la educación básica y media en cuanto a la renovación e innovación curricular que se está gestando a nivel mundial, no solo incluye la introducción de la informática como área clave y estratégica para el desarrollo del pensamiento creativo y computacional, sino que estimula un cambio organizativo y

metodológico alrededor del proceso de enseñanza-aprendizaje y su articulación en las instituciones. La búsqueda de soluciones que respondan a este cambio depende de las políticas educativas, de la gestión de cada institución y del nivel de intervención que logren los docentes en los procesos de transformación curricular.

La formación en informática de manera incremental y transversal en todo el currículo de los niveles básico y medio, prioriza el aprendizaje desde lo práctico y aplicado, apoyado con metodologías de enseñanza como las basadas en problemas y en proyectos que aborda la solución de retos/problemas. Los estudiantes logran soluciones a partir del conocimiento disciplinar, poniendo en práctica el trabajo colaborativo, el pensamiento creativo y crítico, y la aplicación de nuevas tecnologías, en particular, la programación y robótica.

A lo largo del proyecto de investigación del que se deriva la estrategia propuesta, se evidencia la importancia de evaluar constantemente los resultados intermedios con los actores involucrados. Gracias a este principio, fue posible refinar el diseño curricular propuesto y probar algunos de los talleres resultantes de la especificación didáctica con grupos de estudiantes de formación básica y media. Adicionalmente, la divulgación constante de resultados a profesores y coordinadores de informática de las instituciones educativas constituyó el primer paso para la conformación del ecosistema de formación en la ciudad de Medellín a nivel de educación básica y media.

Referencias

- Álava, Á. F. B., Villamar, J. D. C., Vallejo, J. L. G., Cedeño, G. A. C., Robalino, A. P. G., & Moncayo, R. P. (2019). *El diseño curricular y la didáctica, ejes fundamentales en la educación superior contemporánea* (Vol. 45). 3Ciencias.
- Álvarez Arregui, E., & Rodríguez Martín, A. (2015). Inspirando el cambio en educación: Ecosistemas de formación para aprender a emprender. *Estudios pedagógicos (Valdivia)*, 41(Especial), 9-29.
- Anderson, L.W., & Krathwohl, D. (Eds.) (2001). *A Taxonomy for Learning, Teaching and Assessing: a Revision of Bloom's Taxonomy of Educational Objectives*. Longman.
- Armengol Casas, M. (2005). Nueva universidad ante la sociedad del conocimiento. *Universities and Knowledge Society Journal*, 2(2), 1.
- Avendaño Porras, V. (2015). *Implementación y uso escolar de las tecnologías de la información y la comunicación: en la Meseta Comiteca Tojolabal del estado de Chiapas*. Centro Regional de Formación Docente e Investigación Educativa.
- Barrera, R., & Montaña, R. (2015). Desarrollo del pensamiento computacional con Scratch. In J. Sánchez (Presidente), *Nuevas ideas en informática educativa. Simposio en XX Congreso Internal. de Informática Educativa*. Santiago, Chile.
- Basogain, X., Olabe, M. Á., & Olabe, J. C. (2015). Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje. *Rev. de Educación a Distancia*, 46(Septiembre). <http://revistas.um.es/red/article/view/240011/182851>

- Buxarrais Estrada, M.R., & Ovide, E. (2011). El impacto de las nuevas tecnologías en la educación en valores del siglo XXI. *Sinéctica*, (37), 1-14.
- Cortés Rincón, A. (2016). *Prácticas innovadoras de integración educativa de TIC que posibilitan el desarrollo profesional docente: un estudio en instituciones de niveles básica y media de la ciudad de Bogotá (Col)* (Tesis Doctorado). Univesidad Autónoma de Barcelona. <http://ddd.uab.cat/record/175877>
- CSTA & ISTE (2011). Operational definition of computational thinking for K–12 education. <https://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf>
- Friebroon-Yesharim, M., & Ben-Ari, M. (2018). Teaching Computer Science Concepts through Robotics to Elementary School Children. *International Journal of Computer Science Education in Schools*, 2(3).
- Galindo Suárez, M. (2016). *Efectos del Proceso de Aprender a Programar con “Scratch” en el Aprendizaje Significativo de las Matemáticas en los Estudiantes de Grado Quinto de Educación Básica Primaria*. Repositorio Institucional del Tecnológico de Monterrey. <https://repositorio.itesm.mx/ortec/handle/11285/619592>
- Goldie, J.G.S. (2016). Connectivism: A Learning Theory for the Digital Age?. *Medical teacher*, 38(10), 1064-1069.
- Guettl, C., & Chang, V. (2008). Ecosystem-based theoretical models for learning in environments of the 21st century. *International Journal of Emerging Technologies in Learning*, 7, 1-11.
- Hazzan, O., Gal-Ezer, J., & Blum, L. (2008). A model for high school computer science education: The four key elements that make it!. In *Proceedings of the 39th SIGCSE technical symposium on Computer science education* (pp. 281-285).
- Henao, M.A. (2011). El proyecto pedagógico integrador: Una experiencia en el proceso de articulación con el programa técnico profesional en sistemas. *Cuaderno Activa*, 1, 59-70.
- Infante, R. C. H., & Miranda, M. E. I. (2017). Aproximación al proceso de enseñanza-aprendizaje desarrollador. *Revista UNIANDES Episteme*, 4(3), 365-375.
- Jara, I., Hepp, P., Rodríguez, J., & Claro, M. (2019). Políticas y prácticas para la enseñanza de las Ciencias de la Computación en América Latina. https://www.costadigital.cl/robotica/wp-content/uploads/2019/09/ComputerScienceLatinAmerica_ESP.pdf
- López, N., Lugo, M. T., & Toranzos, L. (2014). *Informe sobre tendencias sociales y educativas en América Latina, 2014: políticas TIC en los sistemas educativos de América Latina*. UNESCO. <https://unesdoc.unesco.org/ark:/48223/pf0000230080?posInSet=20&queryId=975dfdo4-c392-4ef4-a950-9bd80146fe00>
- Matsuda, T. (2010). Instructional materials for “information study” teachers’ professional development. In *Society for Information Technology & Teacher Education International Conference* (pp. 3307-3312). VA, USA.

- Moreno-León, J., Robles, G., & Román-González, M. (2015). Dr. Scratch: análisis automático de proyectos Scratch para evaluar y fomentar el Pensamiento Computacional. *RED, Revista de Educación a Distancia*, 46. http://www.um.es/ead/red/46/moreno_robles.pdf
- Morrás Aranoa, H. (2014). Iniciación a la programación informática en Educación Primaria con Scratch. *Academia-e*. <http://academica-e.unavarra.es/handle/2454/14363>
- Núñez Pérez, J. C., González García, J. A., García Rodríguez, M. S., González-Pumariega Solís, S., Roces Montero, C., Álvarez Pérez, L., & González Torres, M. D. C. (1998). Estrategias de aprendizaje, autoconcepto y rendimiento académico. *Psicothema*, 10 (1), 97-109.
- OCDE (2013). Panorama de la educación 2013: Indicadores de la OCDE. <http://www.oecd.org/education/Panorama%20de%20la%20educacion%202013.pdf>
- OME -Ontario Ministry of Education (2008). The Ontario Curriculum Grades 10 to 12: Computer Studies. http://www.edu.gov.on.ca/eng/curriculum/secondary/computer10to12_2008.pdf
- Riesco Albizu, M., Díaz Fondón, M. Á., Álvarez Gutiérrez, D., López Pérez, B., Cernuda del Río, A., Fuente, J., & Adolfo, A. (2014). Informática: materia esencial en la educación obligatoria del siglo XXI. *ReVisión*, 7 (3).
- Rodríguez, M.A. (2017). Desarrollo del pensamiento computacional en educación primaria: una experiencia educativa con Scratch. *Revista de Ciències de l'Educació*, 1(2), 45-64.
- Said, E. (2015) (Ed.). *Hacia el fomento de las TIC en el sector educativo en Colombia*. Editorial Universidad del Norte.
- Sánchez, A. (2006). Evaluación objetiva de competências: Miscelánea Comillas. *Revista de Ciencias Humanas y Sociales*, 64(124), 321-345.
- Schmidt, J. (2010). *Peer 2 Peer University 2010*. <http://vimeo.com/11158136>
- SEI - Secretaría de Estrategias Industriales (2017). La Digitalización y la Industria 4.0. Impacto industrial y laboral. <https://industria.ccoo.es/4290fc51a3697f785ba14fce86528e10000060.pdf>
- UK (2013). National curriculum in England: computing programmes of study. <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study>
- Vargas, J., Guapacho, J. & Isaza, L. (2017). Robótica móvil: una estrategia innovadora en el proceso de enseñanza y aprendizaje. *Revista Virtual Universidad Católica del Norte*, 52, 100-118.
- Vega-Moreno, D., Cufí Solé, X., Rueda, M. J., & Llinás, D. (2016). Integración de robótica educativa de bajo coste en el ámbito de la educación secundaria para fomentar el aprendizaje por proyectos. *International Journal of Educational Research and Innovation*, 6, 162-175.

- Velásquez, J.C. (2017). Ambientes de aprendizaje para el desarrollo de la creatividad. En R. López (Ed.). *Estrategias de enseñanza creativa: investigaciones sobre la creatividad en el aula* (pp.11-30). Universidad de la Salle.
- Yadav, A., Gretter, S., Hambrusch, S., & Sands, P. (2016). Expanding computer science education in schools: understanding teacher experiences and challenges. *Computer Science Education*, 26(4), 235-254.
- Zambrano, A. (2005). *Didáctica, pedagogía y saber*. Cooperativa Editorial Magisterio.

Conversión de eventos desde esquemas preconceptuales en código PL/pgSQL: simulación de software en la cuarta revolución industrial

Carlos Mario Zapata-Jaramillo¹, Juan Sebastián Zapata-Tamayo¹,
Paola Andrea Noreña Cardona¹

cmzapata@unal.edu.co; jzapatat@unal.edu.co; panorenac@unal.edu.co

¹ Universidad Nacional de Colombia, Carrera 80 No. 65-223 Of. M8A-310, 050036, Medellín, Colombia.

DOI: 10.17013/risti.39.18-34

Resumen: La simulación es una de las áreas clave de la Industria 4.0. En la industria de software, la simulación se aplica en la predicción del comportamiento de las aplicaciones a partir del modelado. Una forma de simulación se relaciona con la obtención de código fuente a partir de los modelos, que en el caso de los eventos se suele realizar desde diferentes esquemas conceptuales hacia lenguajes basados en disparadores y viceversa, sin incluir el PL/pgSQL. Por ello, en este artículo se propone un conjunto de reglas heurísticas para obtener código PL/pgSQL a partir de esquemas preconceptuales, como una forma de simular el comportamiento de las aplicaciones que incluyen eventos. Estas reglas se validan mediante un estudio de laboratorio correspondiente a la expansión de una epidemia. Mediante la ejecución de las reglas en el estudio de laboratorio se puede simular el comportamiento futuro de la aplicación y los resultados asociados.

Palabras-clave: Simulación; Industria 4.0; Eventos; Esquemas preconceptuales; PL/pgSQL.

Event conversion from pre-conceptual schemas to PostgreSQL code: software simulation in the fourth industrial revolution

Abstract: Simulation is one of the key areas of Industry 4.0. Simulation in the software industry is applied to the prediction of the application behavior from modeling. One kind of simulation is related to obtaining source code from models; in the case of events, such simulation can be achieved from some conceptual schemas to trigger-based languages and *vice versa*, but excluding PL/pgSQL. For this reason, in this paper, we propose a set of heuristic rules for obtaining PL/pgSQL code from pre-conceptual schemas, as a way to simulate the behavior of applications that include events. Such rules are validated by using a lab study related to the expansion of an epidemic. We can simulate the future behavior of the application and the associated results by executing the rules in the lab study.

Keywords: Simulation; Industry 4.0; Events; Pre-conceptual schemas; PostgreSQL.

1. Introducción

Baygin, Yetis, Karakose, & Akin (2016) presentan nueve áreas clave de la Industria 4.0. La simulación, objeto de este artículo, es una de ellas y se entiende como la reproducción virtual de los procesos que ocurren en las industrias. En la industria de software, y más concretamente en la ingeniería de software, la simulación puede contribuir en la evaluación del impacto de las aplicaciones y también en procesos de rediseño (Tanir, 2017) a partir del modelado.

La generación de código a partir de modelos es una forma de simulación que puede ser manual o automática (Boulanger, 2018) y siempre requiere la implementación de un cierto número de reglas para asegurar que los requisitos definidos en el diseño se consideren. Algunos modelos se traducen en lenguajes como el *Structured Query Language* SQL (Awiti, Vaisman, & Simányi, 2020; Teorey, Lightstone, Nadeau, & Jagadish, 2005; Chaverra, 2011; Chochlik, Kostolny, & Martincova, 2015). En otras propuestas se hace ingeniería inversa a porciones de código en lenguajes basados en disparadores, como el *Programming Language Structured Query Language* PL/SQL (Methakullawat & Limpiyakorn, 2014; Habringer, Moser, & Pichler, 2014; Fernández *et al.*, 2019), para generar modelos. Los eventos, entendidos como fenómenos que ocurren en los procesos de negocios y que permiten iniciar operaciones, se suelen modelar en diferentes esquemas conceptuales como redes de Petri (Narciso & Piera, 2017), *Business Process Modeling and Notation* BPMN (Object Management Group, 2011) y esquemas preconceptuales (Zapata-Jaramillo, 2011), a partir de los cuales se pueden convertir en diferentes lenguajes de programación, como PL/SQL (El-Hayat, Toufik, & Bahaj, 2020; Zapata-Tamayo, 2019). Sin embargo, en ninguna de estas propuestas se toma en consideración el PL/pgSQL, que es un lenguaje de programación basado en disparadores del gestor de bases de datos PostgreSQL.

Por ello, en este artículo se propone un conjunto de reglas heurísticas que posibilita la generación automática de código en PL/pgSQL a partir de esquemas preconceptuales en los que se modelan eventos, de manera que se pueda estudiar el comportamiento de este tipo de aplicaciones en los procesos de simulación que se realizan en la industria de software, en el contexto de la Industria 4.0. Como forma de validación se selecciona un estudio de laboratorio proveniente de la literatura especializada (Noreña, 2018) correspondiente al comportamiento de una epidemia en expansión.

Este artículo se organiza de la siguiente manera: en la Sección 2 se presenta el marco conceptual de la propuesta; en la Sección 3 se presentan los antecedentes; en la Sección 4 se propone la solución y su consecuente validación con un caso de laboratorio; las conclusiones y el trabajo futuro se discuten en la Sección 5.

2. Marco conceptual

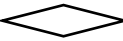
2.1. Simulación en la Industria 4.0

Baygin *et al.* (2016) establecen como uno de los fines de la Industria 4.0 la posibilidad de unir las tecnologías de información con la industria, de modo que se puedan crear fábricas inteligentes. Esa visión se materializa con nueve áreas tecnológicas diferentes:

Big Data y Analítica, Robots Autónomos, Simulación, Integración Horizontal y Vertical de Sistemas, Internet de las Cosas, Ciberseguridad, *Cloud*, Manufactura Aditiva y Realidad Aumentada. Tanir (2017) define la simulación como una técnica de propósito general de gran valor para evaluar espacios de solución de múltiples facetas de manera temprana durante las fases de planeación y diseño sin tener necesidad de un despliegue físico de las posibles alternativas de diseño. Por ello, Tanir (2017) establece que la simulación permite evaluar los impactos que se generen en los procesos de negocios y en las tecnologías de la información, tomando en consideración que se pueden requerir procesos de rediseño. Para Gunal (2019) la simulación puede desempeñar un rol primordial en la realización de la Industria 4.0.

2.2.Representación de eventos en esquemas conceptuales

Luckham (2011) define un evento como algo que ocurre en un sistema o dominio en particular y que sirve para dirigir automáticamente los procesos de negocio. Los eventos se representan en diferentes esquemas conceptuales. Narciso y Piera (2017) emplean transiciones entre los diferentes estados en una red de Petri. En BPMN (*Object Management Group*, 2011), se representan los estados con círculos que encierran diferentes elementos dependiendo de la naturaleza del evento (sobres de carta para eventos de mensaje, relojes para eventos de tiempo, triángulos para eventos tipo señal, etc.). Para efectos de este artículo, se usa la notación de eventos que se define en esquemas preconceptuales (Zapata-Jaramillo, 2011; Noreña (2018), ya que existe ya un conjunto de reglas heurísticas que, a partir de estos esquemas, permiten la traducción a código PL/SQL (Zapata-Tamayo, 2019). Los principales símbolos de los esquemas preconceptuales (incluyendo la notación correspondiente a eventos) se muestran en la Tabla 1.

Símbolo	Explicación
	Concepto: Sustantivo o sintagma nominal del dominio en singular, por ejemplo, <i>patient</i>
	Relación estructural: Verbo “ser” o “estar” en tercera persona del singular
	Relación dinámica: Verbo de acción u operación relacionado con el dominio en tercera persona del singular, por ejemplo, <i>inserts</i>
	Condicional: Contenedor gráfico de expresiones para evaluar su cumplimiento, por ejemplo, <i>Time = 0 weeks</i>
	Instancia: Posible número o texto que se asigna a un concepto, por ejemplo “ <i>stop</i> ”
	Conexión: Flecha para conectar conceptos con relaciones y viceversa
	Implicación: Flecha de tipo causa-efecto para conectar condicionales y relaciones dinámicas
	Relación eventual: Verbo que no necesita agentes, por ejemplo, <i>arrives</i>
	Evento: Combinación de conceptos y relaciones eventuales, por ejemplo, <i>time passes</i>

	Variable: Concepto independiente cuyo valor cambia a lo largo del proceso, por ejemplo, <i>growth rate</i>
	Parámetro: Concepto independiente con un valor fijo a lo largo de la simulación, por ejemplo, <i>population</i>
	Operador: Símbolo que permite incluir operaciones en el esquema preconceptual, por ejemplo +, /, sin, max, etc.
	Línea: Conexión simple para ligar conceptos, operadores e instancias para crear árboles de especificación
	Conjunto concepto-atributo: Nodo para evaluar conjuntos que incluyan un concepto y un atributo particular, por ejemplo, <i>patient.amount</i>
	Especificaciones/condiciones iniciales/restricciones: Contenedor que permite incluir expresiones en forma de árbol. Se debe etiquetar como “condiciones iniciales” para expresar valores que se asignan a las variables, conceptos y parámetros al inicio de la simulación. Expresiones rodeadas por llaves son las restricciones que se deben cumplir
	Conexión punteada: Línea para ligar condiciones iniciales o restricciones o especificaciones con los elementos del esquema preconceptual

Tabla 1 – Símbolos de los esquemas preconceptuales (Adaptado de Zapata-Jaramillo, 2011 y Noreña, 2018)

2.3. PL/pgSQL

Worsley y Drake (2001) definen PostgreSQL como un sistema de gestión de bases de datos objeto relacional de código abierto que incluye los comandos típicos del SQL junto con varias características adicionales, como búsquedas complejas, disparadores (*triggers*) y control de concurrencia multiversión. Además, el usuario puede extender PostgreSQL con lenguaje PL/pgSQL, que es el lenguaje de programación de este sistema de gestión de base de datos. Por ser el tema de este artículo, se muestra seguidamente la estructura de las funciones:

```
CREATE FUNCTION function_identifier () RETURNS opaque AS ‘
DECLARE
    declarations;
    [...]
BEGIN
    statements;
    [...]
END;
‘ LANGUAGE ‘plpgsql’;
```

Los *triggers* se usan para invocar las funciones en el caso de que se requiera su ejecución de manera automática ante un evento. La sintaxis, en ese caso, es la siguiente:

```
CREATE TRIGGER function_identifier
  [AFTER/BEFORE] [CREATE/UPDATE/DELETE] OF field ON table
  FOR EACH ROW
  EXECUTE PROCEDURE function_identifier ();
```

3. Antecedentes

Para Boulanger (2018), una élite de expertos realiza la programación de una aplicación de software, aunque ese proceso se puede automatizar con la asistencia manual de un conjunto de reglas heurísticas o incorporando esas reglas en una herramienta con fines de automatización. Dado que el tema central de este artículo es la generación de código PL/pgSQL desde modelos que incluyan eventos, una revisión sistemática de literatura arroja que no existen propuestas asociadas con ese tópico en particular, pero hay similitudes que son de utilidad para el proceso.

3.1. Desde modelos hacia SQL

Awiti *et al.* (2020) parten de BPMN para obtener expresiones SQL. Teorey *et al.* (2005) y Chochlik *et al.* (2015) parten del diagrama entidad-relación para generar SQL. De manera similar, Chaverra (2011) emplea el esquema preconceptual para generar el diagrama entidad-relación correspondiente y, simultáneamente, la estructura de las tablas en SQL. El diagrama entidad-relación es un esquema conceptual que se usa para representar la estructura del dominio, por lo cual los eventos no se pueden representar en él. En los esquemas preconceptuales se pueden representar los eventos, pero esta porción del esquema no es la que emplea Chaverra (2011) para su proceso de generación de código. Además, si bien SQL es el punto de partida para PL/pgSQL, el SQL tampoco posee sintaxis que permita representar los eventos. Se resalta de estas propuestas la adición de reglas heurísticas para transformar esquemas preconceptuales en sentencias SQL.

3.2. Desde PL/SQL hacia esquemas conceptuales y viceversa

La ingeniería inversa se usa para recuperar esquemas conceptuales cuando existe alguna porción de código legado. Methakullawat y Limpiyakorn (2014) utilizan código legado PL/SQL para generar el diagrama de clases de UML (*Unified Modeling Language*). Las sentencias que incluyen procedimientos sólo las utilizan para generar la parte estructural del diagrama de clases. De forma similar, Habringer *et al.* (2014) usan procedimientos almacenados en PL/SQL para generar el diagrama de flujo de datos, que en cierta forma es también estructural. Ni en el diagrama de clases de UML ni en el diagrama de flujo de datos se toman en cuenta los eventos, aunque el lenguaje PL/SQL sí los soporta. Fernández *et al.* (2019) usan *triggers* de PL/SQL para obtener código Java usando diagramas de clases.

El-Hayat *et al.* (2020) parten de un diagrama de clases enriquecido con OCL para definir *triggers* en PL/SQL. Zapata-Tamayo (2019) parte de esquemas preconceptuales que incluyen eventos y, con base en un conjunto de reglas heurísticas, genera sentencias PL/

SQL con la sintaxis completa, de forma tal que se puede ejecutar en un gestor de Oracle™. Sin embargo, Oracle™ no es un gestor de código abierto y requiere licenciamiento para su uso. En esta propuesta, sin embargo, no se considera la traducción de las reglas a código PL/pgSQL, que sí es de código abierto. Al igual que el PL/SQL, su homólogo de código abierto PL/pgSQL también tiene una sintaxis apropiada para soportar eventos.

4. Solución y validación

Gunal (2019) define para la simulación en Industria 4.0 un “gemelo digital”, que es un software para controlar elementos físicos de una plataforma colaborativa de procesos industriales de negocios y para recolectar datos de esa plataforma que permitan predecir y mejorar su funcionamiento. En las empresas de software no se desarrolla un producto tangible y, por ello, el gemelo digital de Gunal (2019) adquiere su sentido cuando se emplean modelos para simular las aplicaciones definitivas y su comportamiento, como es el caso de la epidemia de influenza que se estudia en este artículo. Por esta razón, el tipo de simulación que se propone con la conversión de un modelo basado en eventos (en este caso el esquema preconceptual) en *triggers* ejecutables en PL/pgSQL posibilitan la adquisición de datos desde etapas preliminares del diseño, actuando como gemelo digital de una aplicación definitiva que permita estudiar un fenómeno desde el punto de vista del modelado.

Zapata-Tamayo (2019) usa las reglas heurísticas de Chaverra (2011) para generar las sentencias SQL correspondientes a la creación de tablas, especialmente aquellas en las cuales se van a almacenar los datos. Además de esas reglas, en este artículo se aprovechan las reglas 1 y 2 de Zapata-Tamayo (2019), en las cuales se generan las tablas correspondientes a las variables (regla 1) y parámetros (regla 2), pues su funcionamiento es idéntico en su traducción a lo que allí se presenta para el lenguaje SQL. Finalmente, se considera el caso de los nombres de variables y parámetros que coinciden con palabras reservadas del lenguaje, adicionando un guión bajo (_) antes del nombre. Adicionalmente, se consideran algunas variaciones a las reglas de Zapata-Tamayo (2019):

- **Sintaxis del PL/pgSQL.** La sintaxis específica del PL/pgSQL tiene algunas diferencias con el PL/SQL. La diferencia fundamental radica en el hecho de que los *triggers* se deben programar como funciones y el código de creación del *trigger* sólo se utiliza para llamar la función correspondiente. En la regla 3 de Zapata-Tamayo (2019) hay que adicionar algunas líneas de la sintaxis específica, como se muestra en la Tabla 2.

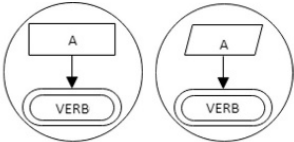
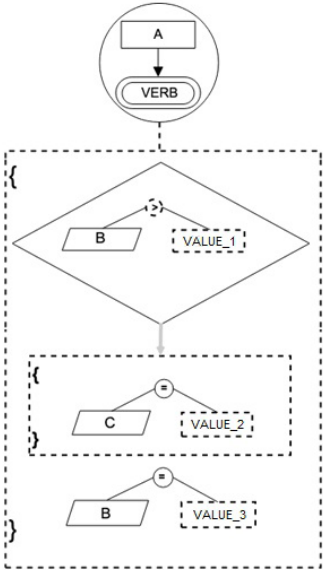
Precondición	Postcondición
	<pre>CREATE TRIGGER -"A_VERB" EXECUTE PROCEDURE "A_VERB" () CREATE FUNCTION -"A_VERB" () RETURNS TRIGGER AS \$"A_VERB" ()\$ END \$"A_VERB" ()\$ LANGUAGE plpgsql</pre>
Nota: La tercera, quinta y sexta línea de la expresión CREATE FUNCTION son necesarias para la ejecución.	

Tabla 2 – Modificación a la regla heurística 3 de Zapata-Tamayo (2019).

- **Efecto de dividir un evento en el modelo en dos *triggers* en lenguaje de programación para evitar los problemas de las tablas mutantes.** Se deben incluir todos los elementos de la condición inicial del evento del modelo en cada uno de los *triggers* en lenguaje de programación, dado que es posible que se presenten problemas de simultaneidad que hagan disparar otro *trigger* con la misma condición final. Ésta es una variación de la Regla 7 de Zapata-Tamayo (2019), tal como se muestra en la Tabla 3.
- **Presencia de dos o más condicionales en paralelo en una especificación de evento.** Hay que adicionar un bloque de código por cada condicional en paralelo que se tenga, tomando la precaución de incluir en el bloque DECLARE todas las variables y parámetros que se requieran en la evaluación de los condicionales. Ésta es una variación a la regla 10 de Zapata-Tamayo (2019), como se ilustra en la Tabla 4.

Precondición	Postcondición
	<pre>CREATE TRIGGER "A_VERB" AFTER INSERT OF "B" ON "VARIABLE" FOR EACH ROW EXECUTE PROCEDURE "A_VERB" () CREATE FUNCTION -"A_VERB" () RETURNS TRIGGER AS \$"A_VERB" ()\$ DECLARE "v_" + B B.TYPE; BEGIN SELECT B FROM [VARIABLE] INTO "v_" + B; IF "v_" + B [CONDICION] VALUE_1 THEN UPDATE [VARIABLE] SET C='VALUE_2'; END IF; RETURN NULL; END; \$"A_VERB" ()\$ LANGUAGE plpgsql CREATE TRIGGER "A_VERB" + "_1" AFTER INSERT OF "C" ON "VARIABLE" FOR EACH ROW EXECUTE PROCEDURE "A_VERB" + "_1" () CREATE FUNCTION "A_VERB" + "_1" () RETURNS TRIGGER AS \$"A_VERB" + "_1" ()\$ DECLARE "v_" + B B.TYPE; BEGIN SELECT B FROM [VARIABLE] INTO "v_" + B; IF "v_" + B [CONDICION] VALUE_1 THEN UPDATE [VARIABLE] SET B='VALUE_3'; END IF; RETURN NULL; END; \$"A_VERB" + "_1" ()\$ LANGUAGE plpgsql</pre>

Nota: Se requiere duplicar el *trigger* porque, como se nota en el modelo, la variable B se invoca tanto para iniciar el evento como para finalizarlo, generando un ciclo infinito. La sintaxis también es similar si se trata del campo de una tabla, pero en ese caso se invocaría en el SELECT la tabla correspondiente y no la tabla *variable*.

Tabla 3 – Modificación a la regla heurística 7 de Zapata-Tamayo (2019).

Con el fin de validar el efecto de las modificaciones a las reglas, se selecciona un caso de laboratorio de la literatura especializada. Noreña (2018) presenta un esquema preconceptual correspondiente al modelo epidémico de una ciudad infectada con influenza. Ese esquema preconceptual incluye los diferentes eventos que controlan el incremento en la cantidad de pacientes que sufren la enfermedad. Como el esquema se encuentra originalmente en el idioma inglés, la aplicación de las reglas se realiza respetando este lenguaje. En el esquema se hacen algunas modificaciones en relación con el original:

Precondición

Postcondición

```

CREATE TRIGGER "A_VERB"
AFTER INSERT OF "B" ON "VARIABLE"
FOR EACH ROW
EXECUTE PROCEDURE "A_VERB" ()

CREATE FUNCTION -"A_VERB" ()
RETURNS TRIGGER
AS $$"A_VERB" ()$

DECLARE
    "v_" + B B.TYPE;
    "v_" + E E.TYPE;
BEGIN
    SELECT B FROM [VARIABLE] INTO "v_" + B;
    SELECT E FROM [VARIABLE] INTO "v_" + E;
    IF "v_" + B [CONDICION] VALUE_1 THEN
        IF "v_" + E [CONDICION] VALUE_2 THEN
            INSERT INTO C (D)
            VALUES (VALUE_4)
        END IF
        IF "v_" + E [CONDICION] VALUE_3 THEN
            INSERT INTO C (D)
            VALUES (VALUE_5)
        END IF
    END IF;
    RETURN NULL;
END;
$$"A_VERB" ()$
LANGUAGE plpgsql
        
```

Nota: Si hubiera más ciclos paralelos, se deberían agregar más sentencias IF después de la línea 18 de la creación de la función. Los árboles de asignación pueden ser tan complejos como se requiera, pero hay que definir variables para cada uno de los elementos (variables, parámetros o atributos) que participen en la asignación.

Tabla 4 – Modificación a la regla heurística 10 de Zapata-Tamayo (2019).

- La constante de integración, que en el modelo de Noreña (2018) se consideraba un parámetro, se considera ahora una variable, pues depende del número de pacientes infectados en la semana cero. El valor inicial de esta variable es cero.
- La tasa de crecimiento, que también se consideraba un parámetro, también se considera ahora una variable, pues depende del número de pacientes infectados en la semana 1 y de la constante de integración. El valor inicial de esta variable es cero.
- Se excluyen de las condiciones iniciales los vectores de tiempo y número de pacientes que se consideran registros iniciales en la tabla paciente, necesarios para el cálculo de la constante de integración y la tasa de crecimiento. En su lugar, se modela un evento denominado *patient amount arrives* (número de

pacientes llega), que sirve para actualizar la tabla de pacientes con los valores correspondientes a los infectados en las semanas 0 y 1, valores que se requieren para el cálculo de las variables *constante de integración* y *tasa de crecimiento* y su actualización con los valores definitivos en dicha tabla.

- Se incluyen en las condiciones iniciales el parámetro *end time* para ponerle fin a la simulación en un tiempo de 6 semanas y la variable *timestamp*, que puede tomar los valores *next* y *stop*, con el fin de controlar la ejecución de los eventos en la simulación. Nótese que al final de cada uno de los eventos hay una asignación diferente del valor de *timestamp*.
- Se reemplazan los vectores tiempo y paciente por los atributos *local time* y *amount*, correspondientes a la tabla *patient*.

El modelo completo, modificado según las consideraciones anotadas, se presenta en la Figura 1. Las flechas de conexión posibilitan la lectura entre los conceptos y las relaciones (por ejemplo “paciente tiene hora local y cantidad”). Las flechas de implicación generan relación causa-efecto (por ejemplo, “cuando el tiempo pasa, la cantidad de pacientes crece”). Las conexiones punteadas permiten detallar los eventos y los operadores permiten complementar las fórmulas (por ejemplo, la parte inicial de “llega cantidad de pacientes” se lee “si la marca de tiempo es igual a pare y el tiempo es menor a dos semanas, se inserta un paciente”). Con base en el modelo de la Figura 1, se inicia la aplicación de las reglas 1 y 2, correspondientes a la creación de las tablas *variable* y *parameter*, como se muestra en la Tabla 5. Además, aplicando las reglas de Chaverra (2011) se crea la tabla *patient*, que sirve para almacenar los datos correspondientes a la cantidad de personas infectadas en la población en cada semana.

El evento *time passes* requiere la variable *timestamp* tanto en el condicional como en el final del evento, por lo cual requiere la aplicación de la regla 7, generando dos *triggers* diferentes tal como se muestra en las Tablas 6 y 7. Además, con la aplicación de la regla 3 se crea el código tanto de cada uno de los *triggers* como de las funciones correspondientes. Ahora, en el caso del evento *patient amount arrives* se debe aplicar la regla 10 para generar las sentencias IF paralelas que se dan en el cuerpo del *trigger*, como se muestra en las Tablas 8 y 9. Dado que el evento termina con la actualización de la variable *timestamp*, que también se necesita en el condicional, se aplica también la regla 7 para generar un segundo *trigger*, pero, por efectos de espacio, ese segundo *trigger* no se incluye en este artículo, al igual que los que corresponden al evento *patient amount grows*, pero su código es similar al que se muestra en los otros eventos.

Una vez se genera la totalidad del código PL/pgSQL mediante la utilización de las reglas, es necesario cambiar los valores en el PostgreSQL para que se inicie la ejecución de los diferentes *triggers*. Como se aprecia en la Tabla 5, antes de iniciar la simulación el valor de *timestamp* está vacío, por lo cual se debe colocar en el valor *stop*, que es la primera condición para ejecutar el proceso. Eso se logra con el siguiente comando en PostgreSQL:

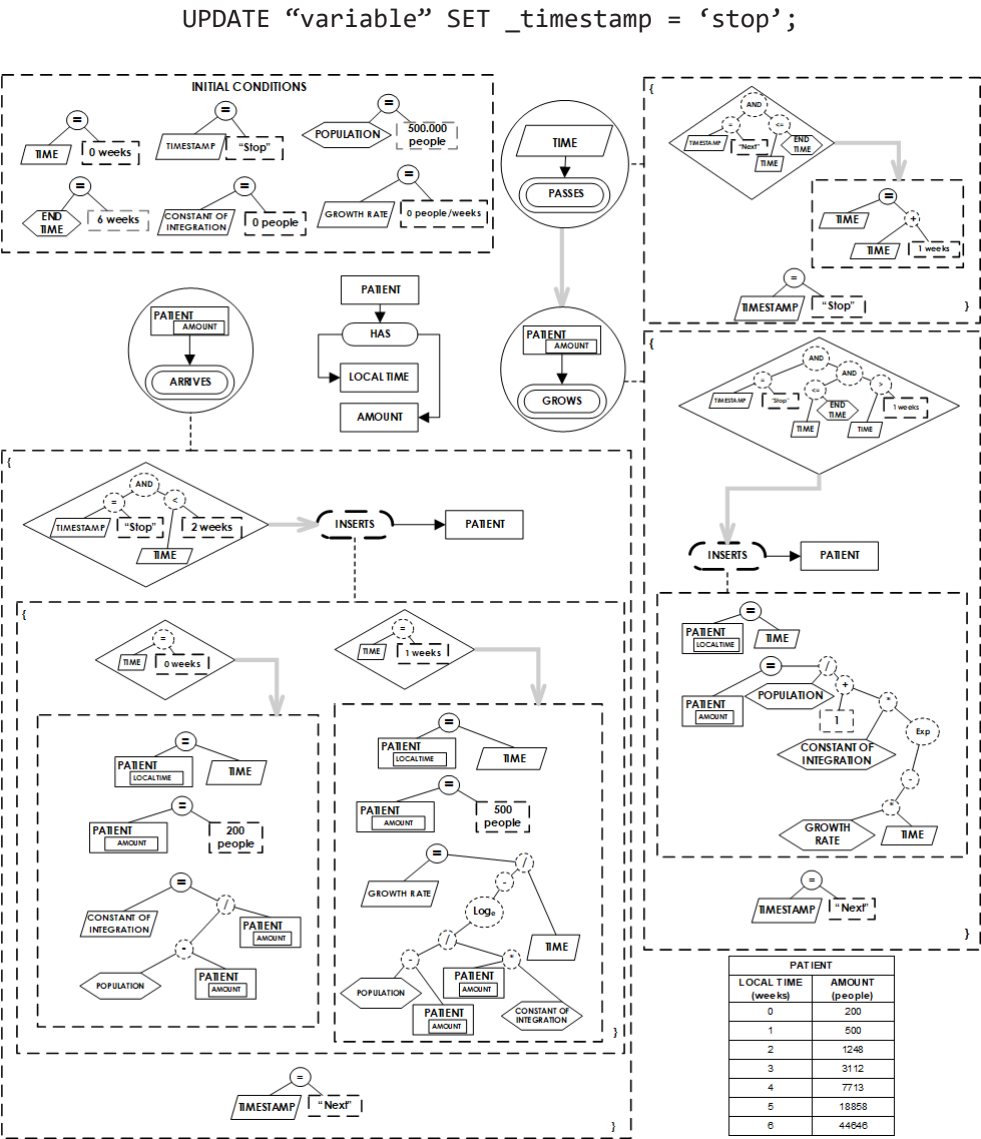


Figura 1 – Adaptación del modelo epidemiológico de Noreña (2018)

Cuando se inicia ese proceso y con los valores que ya se tienen en las tablas *variable*, *parameter* y *patient* se generan los siete valores que se muestran en la Figura 1 para la tabla *patient* en *local_time* 0 a 6.

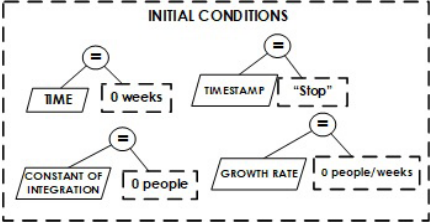
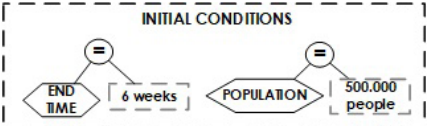
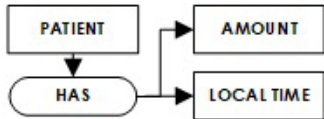
Esquema preconceptual	Código generado PL/pgSQL
	<pre>CREATE TABLE variable (id serial PRIMARY KEY, _timestamp text NOT NULL, _time float NOT NULL, growth_rate float NOT NULL, constant_of_integration float NOT NULL); INSERT INTO variable (_timestamp, _time, growth_ rate, constant_of_integration) VALUES ('', 0, 0, 0);</pre>
	<pre>CREATE TABLE parameter (id serial PRIMARY KEY, _population float NOT NULL, end_time float NOT NULL); INSERT INTO parameter (_population, end_time) VALUES (500000, 6);</pre>
	<pre>CREATE TABLE patient (id serial PRIMARY KEY, local_time float NOT NULL DEFAULT 0, amount float NOT NULL DEFAULT 0);</pre>

Tabla 5 – Aplicación de las reglas 1 y 2 para las tablas *variable* y *parameter* y de las reglas de Chaverra (2011) para la creación de la tabla *patient*.

Ahora, si se hace un UPDATE suficientemente grande del parámetro *end_time*, es posible determinar cuándo la epidemia alcanzará toda la población en la ciudad. Para este caso en particular, se encuentra que en la semana 23 se alcanza un *amount* en *patient* de 499999.1327998074, un número suficientemente cercano al de la población total de la ciudad, que es de 500000. Si se quiere ser suficientemente preciso, en la semana 49 se llega a ese número exactamente.

5. Conclusiones y trabajo futuro

En este artículo se propuso un conjunto de reglas heurísticas que posibilitan la generación de código PL/pgSQL a partir de modelos que incluyen eventos representados en esquemas preconceptuales.

Estas reglas constituyen un ejemplo de la manera en que la industria del software emplea el área clave de simulación correspondiente a la Industria 4.0 para predecir el comportamiento de sistemas complejos, lo cual se pudo comprobar mediante un caso de laboratorio correspondiente a la literatura especializada y en el cual se ejemplifica

la expansión de una epidemia de influenza en una ciudad. Con el código generado fue posible la generación automática de los valores que se predijeron para el modelo. El código se generó completamente a partir de las reglas y fue completamente funcional en el gestor de bases de datos PostgreSQL, de código abierto.

Esquema preconceptual	Código generado PL/pgSQL
Nota: en esta parte de la Tabla se puede apreciar la generación del código de los dos <i>triggers</i> y el código de la primera función, correspondiente a la actualización de la variable <i>time</i> con el incremento de una semana.	<pre> CREATE TRIGGER time_passes AFTER UPDATE OF _timestamp ON variable FOR EACH ROW EXECUTE PROCEDURE time_passes (); CREATE TRIGGER time_passes_1 AFTER UPDATE OF _time ON variable FOR EACH ROW EXECUTE PROCEDURE time_passes_1 (); CREATE FUNCTION time_passes () RETURNS TRIGGER AS \$time_passes\$ DECLARE v_timestamp text; v_time float; v_end_time float; BEGIN SELECT _timestamp FROM variable INTO v_timestamp; SELECT _time FROM variable INTO v_time; SELECT end_time FROM parameter INTO v_end_time; IF (v_timestamp = 'next') THEN IF (v_time <= v_end_time) THEN UPDATE variable SET _time = v_time + 1; END IF; END IF; RETURN NULL; END; \$time_passes\$ LANGUAGE plpgsql; </pre>

Tabla 6 – Aplicación de la regla 7 para la creación de los *triggers* correspondientes al evento *time passes*.

Esquema preconceptual	C3digo generado PL/pgSQL
Nota: en esta parte de la tabla se muestra la aplicaci3n de la regla 7 para generar la segunda funci3n, en la cual se actualiza la variable <i>timestamp</i> para que tenga el valor <i>stop</i>, que es el que dispara la aplicaci3n de otros <i>triggers</i>.	<pre> CREATE FUNCTION time_passes_1 () RETURNS TRIGGER AS \$time_passes_1\$ DECLARE v_timestamp text; v_time float; v_end_time float; BEGIN SELECT _timestamp FROM variable INTO v_timestamp; SELECT _time FROM variable INTO v_time; SELECT end_time FROM parameter INTO v_end_time; IF (v_timestamp = 'next') THEN IF (v_time <= v_end_time) THEN UPDATE variable SET _timestamp = 'stop'; END IF; END IF; RETURN NULL; END; \$time_passes_1\$ LANGUAGE plpgsql; </pre>

Tabla 7 – Aplicaci3n de la regla 7 para la creaci3n de los *triggers* correspondientes al evento *time passes* en su segunda funci3n.

Como líneas de trabajo futuro, se proponen las siguientes:

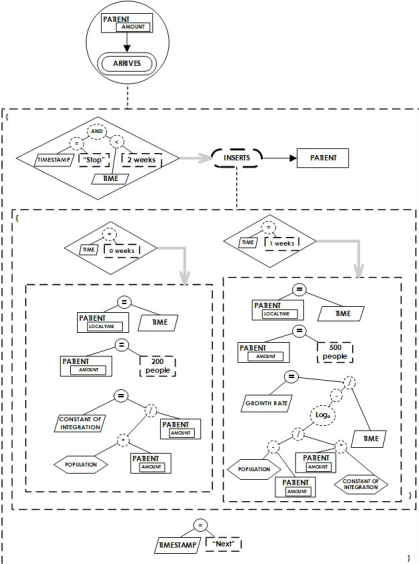
Esquema preconceptual	Código generado PL/pgSQL
 Nota: en esta parte de la Tabla se presenta la aplicación de la regla 3 para generar el código del <i>trigger</i> y el inicio del código de la función. Acá se muestra la declaración de las variables y los comandos SELECT para obtener los diferentes valores de las tablas correspondientes, que se usarán luego en el cálculo.	<pre> CREATE TRIGGER patient_amount_arrives AFTER UPDATE OF _timestamp ON variable FOR EACH ROW EXECUTE PROCEDURE patient_amount_arrives (); CREATE FUNCTION patient_amount_arrives () RETURNS TRIGGER AS \$patient_amount_arrives\$ DECLARE v_timestamp text; v_time float; v_population float; v_constant_of_integration float; v_growth_rate float; v_patient_local_time float; v_patient_amount float; BEGIN SELECT _timestamp FROM variable INTO v_timestamp; SELECT _time FROM variable INTO v_time; SELECT _population FROM parameter INTO v_population; SELECT constant_of_integration FROM variable INTO v_constant_of_integration; SELECT growth_rate FROM variable INTO v_growth_rate; SELECT local_time FROM patient INTO v_patient_local_time; SELECT amount FROM patient INTO v_patient_amount; </pre>

Tabla 8 – Aplicación de la regla 3 para la creación de uno de los *triggers* correspondiente al evento *patient amount arrives*.

- Elaboración de una herramienta que incluya las reglas que se definen en este artículo y las que define Zapata-Tamayo (2019) para la generación automática de código fuente, tanto en PL/SQL como en PL/pgSQL a partir de las representaciones en esquemas preconceptuales.

- Estudio de otros fenómenos relacionados con eventos complejos para determinar reglas heurísticas adicionales en situaciones que así lo requieran para el funcionamiento autónomo de los eventos en este tipo de simulación.
- Exploración de otras maneras en que la simulación basada en eventos pueda contribuir en la Industria 4.0.

Esquema preconceptual

The diagram illustrates a preconceptual schema for a patient management system. It starts with a 'PARENTE (AMOUNT)' entity, which leads to an 'AMBROS' entity. A dashed line connects 'AMBROS' to a complex logic block. This block contains two main paths. The first path involves a decision diamond '10 weeks' leading to an 'INSERT' operation into a 'PARENTE' entity. The second path involves a decision diamond '1 week' leading to a 'GROWTH RATE' calculation, which then leads to a 'CONSTANT OF INTEGRATION' calculation. Both paths eventually lead to a 'BME' entity. The diagram also shows a 'TIMESTAMP' entity at the bottom, which is linked to the 'CONSTANT OF INTEGRATION' calculation.

Código generado PL/pgSQL

```

IF (v_timestamp = 'stop') THEN
  IF (v_time < 2) THEN
    IF (v_time = 0) THEN
      INSERT INTO patient (local_time, amount)
        VALUES (v_time, 200);
      SELECT amount
        FROM patient
      INTO v_patient_amount
      ORDER BY
        id DESC
      LIMIT 1;
      UPDATE variable
      SET
        constant_of_integration = ((v_population - v_patient_
        amount) / v_patient_amount);
      END IF;
      IF (v_time = 1) THEN
        INSERT INTO patient (local_time, amount)
          VALUES (v_time, 500);
        SELECT amount
          FROM patient
        INTO v_patient_amount
        ORDER BY
          id DESC
        LIMIT 1;
        UPDATE variable
        SET
          growth_rate = (- ln(((v_population - v_
          patient_amount) / (v_patient_amount * v_constant_
          of_integration)))) / v_time;
        END IF;
      END IF;
      RETURN NULL;
    END;
    $patient_amount_arrives$
  LANGUAGE plpgsql;

```

Nota: en esta parte de la tabla se aprecia la aplicación de la regla 10 para los dos ciclos IF que se generan. Inicialmente, se hace el cálculo de *constant_of_integration* en la semana 0 y luego se calcula *growth_rate* en la semana 1.

Tabla 9 – Aplicación de la regla 10 para la creación de uno de los *triggers* correspondiente al evento *patient amount arrives*.

Agradecimiento

Este artículo es parte del proyecto de Maestría titulado “Generación Semi-automática de Código PL/SQL a partir de Representaciones de Eventos Basadas en Esquemas Preconceptuales”, patrocinado por la Facultad de Minas de la Universidad Nacional de Colombia, sede Medellín.

Referencias

- Awiti, J., Vaisman, A., & Simányi, E. (2020). Design and implementation of ETL processes using BPMN and relational algebra. *Data and knowledge engineering*, 129, 1-14. 10.1016/j.datak.2020.101837
- Baygin, M., Yetis, H., Karakose, M., & Akin, E. (2016). An Effect Analysis of Industry 4.0 to Higher Education. In *15th International Conference on Information Technology Based Higher Education and Training (ITHET)*. Istanbul.
- Boulanger, J. (2018). *Certifiable software applications 3: Downward cycle*. Elsevier Ltd.
- Chaverra, J. J. (2011). *Generación Automática de Prototipos Funcionales a Partir de Esquemas Preconceptuales*. (Tesis de Maestría). Departamento de Ciencias de la Computación y de la Decisión, Universidad Nacional de Colombia, Medellín.
- Chochlik, M., Kostolny, J., & Martincova, P. (2015). Metamodel describing a relational database schema. *Central European Researchers Journal*, 1(1), 94-102.
- El-Hayat, S., Toufik, F., & Bahaj, M. (2020). UML/OCL based design and the transition towards temporal object relational database with bitemporal data. *Journal of King Saud University—Computer and information sciences*, 32, 398-407.
- Fernández, C., Molina, J., Bermúdez, F., Hoyos, J., Sevilla, D., & Cuesta, B. (2019). Developing a model-driven reengineering approach for migrating PL/SQL triggers to Java: a practical experience. *The journal of systems and software*, 151, 38-64.
- Gunal, M. (2019). Simulation and the Fourth Industrial Revolution. In *Simulation for Industry 4.0: past, present, and future* (pp. 1-17). Springer.
- Habringer, M., Moser, M., & Pichler, J. (2014). Reverse Engineering PL/SQL Legacy Code: An Experience Report. In *IEEE International Conference on Software Maintenance and Evolution*. Victoria, Canada.
- Luckham, D. (2011). *Event Processing for Business: Organizing the Real-time Enterprise*. John Wiley & Sons.
- Methakullawat, N., & Limpiyakorn, Y. (2014). Reengineering Legacy Code with Model Transformation. *International Journal of Software Engineering and Its Applications*, 8(3), 97-110.
- Narciso, M., & Piera, M. A. (2017). Modeling Discrete Event Systems Using Petri Nets, in *Robust Modelling and Simulation* (pp. 49-85). Springer.
- Noreña, P. (2018). An Extension to Pre-conceptual Schemas for Refining Event Representation and Mathematical Notation. In *21st Conferencia Iberoamericana en Software Engineering (CIbSE)* (pp. 589-596). Bogotá, Colombia.
- Object Management Group (2011). About the business process modelling and notation specification version 2.0 [Internet]. <http://www.omg.org/spec/BPMN/2.0>
- Tanir O. (2017). Simulation-Based Software Engineering. In *Guide to Simulation-Based Disciplines. Simulation Foundations, Methods and Applications*. Springer.

- Teorey, T., Lightstone, S., Nadeau, T., & Jagadish, H. V. (2005). *Database Modeling and Design (4th Edition)*. Morgan Kaufmann.
- Worsley, J. & Drake, J. (2001). *Practical PostgreSQL*. O'Reilly.
- Zapata-Jaramillo, C. M. (2011). *The UNC-Method revisited: elements of the new approach*. Lambert Academic Publishing.
- Zapata-Tamayo, J. S. (2019). *Generación Semiautomática de Código PL/SQL a partir de Representaciones de Eventos Basadas en Esquemas Preconceptuales*. (Tesis de Maestría). Departamento de Ciencias de la Computación y de la Decisión, Universidad Nacional de Colombia, Medellín.

Detección de paráfrasis basada en la energía, entropía y temperatura textual

Alberto Iturbe-Herrera¹, Armando Rojas-Valdez¹,
Noé Alejandro Castro-Sánchez¹, Gerardo Sierra²

**Iturbe@cenidet.edu.mx; armando.rojas17ce@cenidet.edu.mx;
noe.cs@cenidet.tecnm.mx; gsierram@iingen.unam.mx**

¹ Tecnológico Nacional de México - Cenidet, Interior internado Palmira S/N, Col. Palmira, 62490, Morelos, México.

² Instituto de Ingeniería, Universidad Nacional Autónoma de México, Av. Universidad, 3000, Coyoacán, 04510, Ciudad de México, México.

DOI: 10.17013/risti.39.35-51

Resumen: La paráfrasis es la reformulación de un texto utilizando un vocabulario distinto para plasmar la idea original con nuestras propias palabras. En esta investigación se presenta un método para la detección de paráfrasis incorporando los conceptos de Entropía y Temperatura Textual a un modelo previo que centró su contribución en la implementación de las redes neuronales recurrentes de Hopfield para generar una medida de distancia llamada Energía Textual. Utilizando la Entropía y la Temperatura se generó un Contexto de Afinidad Libre, basándose en el Model Ising, lo que permitió medir la distribución semántica entre pares de oraciones. Este modelo fue evaluado utilizando el recurso *Microsoft Research Paraphrase Corpus*, permitiendo superar los resultados del modelo anterior y logrando identificar más de la mitad de la paráfrasis de la muestra analizada.

Palabras-clave: detección de paráfrasis; energía textual; entropía; temperatura textual; red de Hopfield.

Paraphrase detection based on Energy, Entropy and Textual Temperature

Abstract: Paraphrases are the reformulation of a text using different vocabulary to capture the original idea in our own words. In this research, a method for paraphrase detection is presented. We incorporate two variables, Entropy and Textual Temperature, into a previous model which implemented a Hopfield Network to generate a distance measure called Textual Energy. A Context of Free Affinity was generated using Entropy and Temperature based on the Ising Model, which allowed us to measure the semantic distribution between pairs of sentences. Our model was evaluated using Microsoft's Research Paraphrase Corpus improving the results of the previous model and was able to identify more than half of the paraphrases presented in the analyzed sample.

Keywords: Paraphrase detection; Textual Energy; Entropy; Textual Energy; Hopfield Network.

1. Introducción

Las características que definen la 4ta. Revolución Industrial se basan en desarrollos tecnológicos que incorporan principalmente Inteligencia Artificial, y en la creciente cantidad de datos que se generan y los algoritmos que se requieren para procesarlos. En noviembre del 2018, la empresa Seagate (líder global en soluciones de almacenamiento) y la consultora IDC (principal proveedor mundial de inteligencia de mercado, servicios de asesoramiento y eventos para las TIC), publicaron un informe titulado *The Digitalization of the World From Edge to Core* (Reinsel, Gantz & Rydning, 2018) donde concluyen que en el año 2025 se habrán generado 5 veces más datos que en el 2018, un aproximado de 175ZB, de los cuales, el 79% se encontrarán en formato de texto.

Bajo este escenario, las soluciones que ofrece el área de Procesamiento de Lenguaje Natural cobran vital importancia. Entre las diferentes tareas que aborda se encuentra la detección de paráfrasis. Se entiende como tal a la expresión de un enunciado utilizando palabras distintas, pero conservando la misma idea; por lo tanto, cuando se habla de paráfrasis se sabe que existe un enunciado original y al menos una versión distinta de él que mantiene el mismo significado.

La detección de paráfrasis es útil en tareas como la generación automática de resúmenes, la traducción automática, la búsqueda de respuestas e incluso para resolver problemas de índole legal, como la detección de plagio, delito que se ha incrementado en los últimos años debido a la gran cantidad de datos existentes en formato de texto y su fácil acceso a través de Internet.

En este trabajo se propone un modelo para la detección de paráfrasis empleando un método de aprendizaje no supervisado con el que se extiende el umbral de recuperación de información de una Red Neuronal Recurrente de Hopfield, el modelo emplea además la Entropía y la Temperatura textual para la extracción de características. El resultado es un modelo de búsqueda de similitud semántica entre pares de enunciados que llega a encontrar más de la mitad de la paráfrasis que puede existir en un corpus lingüístico.

El artículo se divide en las siguientes secciones: en el capítulo 2 se mencionan los trabajos relacionados a la detección automática de la paráfrasis, el capítulo 3 introduce la definición de los conceptos clave que permiten tener una mejor contextualización teórica del trabajo, en el capítulo 4 se explica el modelo de solución propuesto para la identificación de la paráfrasis, el capítulo 5 abarca los experimentos y resultados y finalmente en el capítulo 6 se señalan las conclusiones obtenidas en este trabajo.

2. Trabajos relacionados

Para detectar paráfrasis en textos se puede recurrir a la intervención manual, y cuando la intención es resolver un uso ilegal de ella, a los peritos correspondientes. Atenderse de esta manera implica una inversión de tiempo y dinero, además de que, como en todo trabajo manual, no está exenta de incurrir en errores. A lo largo de los años

han surgido diferentes propuestas para resolver esta tarea de forma automática. Los trabajos publicados tradicionalmente se han dividido en dos grandes categorías según las técnicas que emplean: por una parte, se encuentran trabajos que se basan en el uso de funciones de similitud para resolver si en un par de oraciones existe paráfrasis, y, por otro lado, se tienen los trabajos que utilizan aprendizaje automático, donde a partir de un corpus de entrenamiento extraen características y generan un modelo que predice la clase correspondiente de un par de oraciones.

En (Mohamed & Wael, 2019), se hace un estudio que abarca trabajos en una ventana de tiempo de 10 años, considerando métodos más recientes y los clasifican en enfoques no supervisados y enfoques supervisados, en este último incluyen enfoques clásicos de aprendizaje automático y enfoques de aprendizaje profundo.

Socher, Huang, Pennington, Ng & Manning (2011) presentan un método para la detección de paráfrasis basado en *autoencoders* recursivos (RAE por sus siglas en inglés). Los RAE no supervisados aprenden vectores de características para frases en árboles sintácticos. Estas características se utilizaron para medir la similitud entre palabras y frases entre dos oraciones. Además, los autores introducen una nueva capa de agrupación dinámica que calcula una representación de tamaño fijo a partir de las matrices de tamaño variable, la cual, se utiliza como entrada para un clasificador.

Dentro de los trabajos relacionados en la detección de paráfrasis, Milajevs, Kartsaklis, Sadrzadeh & Purver (2014) realizan un estudio comparativo entre representaciones neuronales de palabras y espacios vectoriales tradicionales basados en recuentos de concurrencia, en una serie de tareas de composición. En éste se utilizan tres espacios semánticos diferentes y se implementan siete modelos de composición basados en tensores, evaluados en tareas que involucran desambiguación de verbos y similitud de oraciones.

Por otro lado, Qiu, Kan & Chua (2006) proponen un modelo que no se centra en la detección de la paráfrasis, sino en identificar las diferencias entre las oraciones, su motivación se basa en que oraciones similares comparten una cantidad sustancial de información, y si existe información adicional en las oraciones, su eliminación no sería significativa. Su modelo se implementa en dos fases, en la primera usan un detector de similitud, donde buscan información similar entre oraciones, y en la segunda aplican un clasificador de diferencias, donde valoran si la información adicional es significativa.

En Mihalcea, Corley & Strapparava (2006) se presenta un método para medir la similitud semántica de los textos, utilizando medidas de similitud basadas en el corpus y el conocimiento. El trabajo previo sobre este problema se ha centrado principalmente en documentos grandes (clasificación de texto, recuperación de información) o palabras individuales (pruebas de sinonimia).

Enfoques más actuales que implementan soluciones basadas en redes neuronales se pueden observar en Lan & Xu (2018) quienes analizan cinco diferentes implementaciones de redes neuronales para el modelado de pares de sentencias, lo cual es fundamental para resolver diversas tareas de Procesamiento de Lenguaje Natural, como la similitud semántica, la identificación de paráfrasis, la inferencia de lenguaje natural, preguntas-

respuestas y la comprensión automática. Sus experimentos los realizaron en 8 conjuntos de datos: dos conjuntos NLI, tres PI, un conjunto de datos STS y dos sobre QA.

3. Marco conceptual

Los fundamentos teóricos en los que se basa este trabajo abarcan diversos conceptos. El primero de ellos es la energía textual, que surge a partir del análisis de las redes neuronales recurrentes de Hopfield, las cuales basan su comportamiento en el modelo termodinámico de Ising. Se propone el uso de dos variables, la Temperatura textual y la entropía, que sirven para reforzar la capacidad de la Red Neuronal Recurrente de Hopfield. Este planteamiento permite medir la distribución semántica entre pares de oraciones. A continuación, todos estos conceptos se definirán en el orden en que se mencionaron.

3.1. Energía Textual

La energía textual es un concepto acuñado por el grupo de investigadores del Laboratorio de Informática Aplicada de la Universidad de Aviñón (Fernández, SanJuan & Torres-Moreno, 2007), quienes identificaron que el comportamiento de las Red Neuronal Recurrente de Hopfield puede ser utilizado para distintas tareas de procesamiento de lenguaje, tales como agrupamiento semántico (Torres-Moreno, Molina y Sierra, 2010). La base que toman es el modelo vectorial, lo que permite interpretar los elementos de una matriz documento-EL (Entidades Léxicas) como los espines o neuronas de una Red Neuronal de Hopfield, definiendo a los documentos como secuencias de neuronas, activando o desactivando la neurona si la aparece o no en la frase respectivamente.

Los autores hacen énfasis en la importancia de la interacción de las unidades consigo mismas desde el punto del procesamiento del lenguaje, a diferencia del modelo de Hopfield donde dicha interacción no es considerada.

3.2. Red Neuronal Recurrente de Hopfield

El concepto de Red neuronal artificial (RNA) se define, según Fausett (1994), como *“un sistema de procesamiento de información que posee ciertas características de rendimiento en común con las redes neuronales biológicas”*. Se aplica para resolver una gran variedad de problemas, como almacenar y recuperar datos o patrones, clasificar patrones y en la agrupación, además de la búsqueda de soluciones a problemas de optimización restringidos (Copeland, 2016).

Goodfellow, Bengio & Courville (2016) señalan que una de las RNA de mayor popularidad son las Redes Neuronales Recurrentes (RNR, utilizadas comúnmente para procesamiento y aprendizaje de datos secuenciales). Las RNR son máquinas de aprendizaje que tienen como objetivo utilizar información secuencial. Se utilizan en diversos contextos donde la dependencia temporal de los datos es una característica implícita de alto impacto. Calculan de forma recursiva nuevos estados implementando funciones de transferencia a estados y entradas previas.

En la Figura 1 se muestra la arquitectura de una RNR simple.

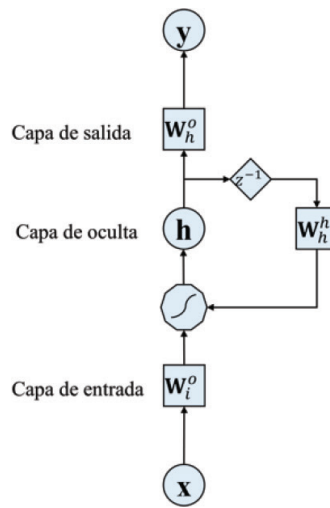


Figura 1 – Arquitectura de una RNR simple

Los datos de entrada se representan con x , la variable y representa los nodos de salida y corresponde a los nodos ocultos. w_i^o y w_h^o son matrices que representan los pesos de entrada, ocultos y salida, respectivamente. El polígono representa la transformación no lineal realizado por las neuronas. Finalmente, z^{-1} corresponde al operador de demora de la unidad.

Hopfield (1982) propuso una RNR denominada Red de memoria asociativa. Es eficiente para resolver problemas de optimización, memorias asociativas, procesamiento de lenguaje, partición de gráficos, visión estéreo, entre otras. Asocia dos valores de activación binario (0, 1) o bien bipolar (1,-1), que se determinan si las unidades superan o no un determinado umbral. Por lo tanto, la definición posible para la unidad de activación es:

$$a_i \leftarrow \begin{cases} 1 & \text{si } \sum_j w_{ij} s_j > \theta_i \\ 0 & \text{en caso contrario.} \end{cases}$$

Donde: w_{ij} representa la fuerza del peso de la conexión de la unidad j a la unidad i , denominado peso de conexión; y , s_j y θ_i corresponden al estado y al umbral de la unidad i , respectivamente.

La Red Recurrente de Hopfield posee un valor escalar asociado a cada estado de la red conocido como Energía (E) de la red, que se define de la siguiente forma:

$$E = -\frac{1}{2} \sum_{i,j} w_{ij} s_i s_j + \sum_i \theta_i s_i$$

Este valor recibe el nombre de Energía, debido a que la definición indica que, si las unidades son elegidas al azar para actualizar los valores de activación, la red convergerá a estados que son mínimos locales de la función de energía.

3.3. Modelo de Ising

Ernst Ising propone un modelo físico para estudiar el comportamiento de materiales ferromagnéticos. Consta de variables discretas que representan momentos dipolares magnéticos de espines atómicos que pueden presentar uno de dos estados (+1 o -1). Los giros se organizan en un gráfico, generalmente una red (donde la estructura local se repite periódicamente en todas las direcciones), permitiendo que cada giro interactúe con sus vecinos. Los giros vecinos que coinciden tienen una energía menor que los que no coinciden; el sistema tiende a la energía más baja, pero el calor perturba esta tendencia, creando así la posibilidad de diferentes fases estructurales (Wolf, 2000).

El modelo permite la identificación de transiciones de fase, como un modelo simplificado de la realidad. El modelo Ising bidimensional de celosía cuadrada es uno de los modelos estadísticos más simples para mostrar una transición de fase.

3.4. Temperatura Textual

La Temperatura Textual es una medida de distancia entre un par de enunciados de acuerdo con su estructura sintáctica (Stepanov, 2012). Tiene su origen en el Modelo de Ising el cual es la base del diseño del modelo computacional propuesto en esta investigación. Para llevar a cabo el cálculo de la temperatura textual se decidió emplear el Coseno Suave, dado que esta variable está estrechamente relacionada con la entropía que tiene como objetivo reducir el ruido (desorden) que es generado en una fuente de información de acuerdo con la Temperatura de la misma. Dicho de otra manera, la Entropía es constante a su Temperatura.

3.5. Entropía

Una de las acepciones de la palabra entropía, según la Real Academia de la Lengua Española, es *Medida del desorden de un sistema* (Real Academia Española, 2020). Considerándola desde el punto de vista de datos, se define como la medida del desorden que existe dentro de una fuente de información, la cual indica el nivel de entropía en un par de oraciones (Grzymala-Busse, 1993). De esta manera, y considerando el orden alfabético, la cadena “GDCBAEF” cuenta con un nivel de entropía alto, pero la cadena “ABCDEF” tiene un nivel de entropía bajo.

3.6. Distribución semántica

También conocida como semántica distribucional (Firth, 1968) es un área de investigación que tiene como objetivo desarrollar y estudiar teorías y métodos para cuantificar y clasificar las similitudes semánticas entre elementos lingüísticos según sus propiedades distribucionales en grandes muestras de datos. La idea básica de la distribución semántica se resume en la hipótesis distribucional (Sahlgren, 2008), la cual tiene origen en la teoría semántica del uso lingüístico, es decir, palabras que se usan y aparecen en los mismos contextos tienden a transmitir significados similares.

Los métodos más empleados para generar una Distribución Semántica son TF-IDF, *K-means*, *Naive-Bayes*, Shannon ID3, *Word Embeddings*, entre otros.

4. Método de solución

En este trabajo se propone incorporar las variables de Entropía y Temperatura textual para reforzar la capacidad de la Red Neuronal Recurrente (RNR) de Hopfield que se utiliza en el modelo de la Energía textual. La salida generada por esta red es un vector que contiene una cantidad indefinida de ruido, por lo que se considera que las variables que se proponen harían posible reducirlo. Esto finalmente permitiría aumentar el umbral de recuperación de información de la red neuronal.

El modelo que se propone (Figura 2) recibe pares de oraciones, una considerada original y otra como posible paráfrasis. Para fines de la experimentación se utilizó el recurso *Microsoft Research Paraphrase Corpus*.

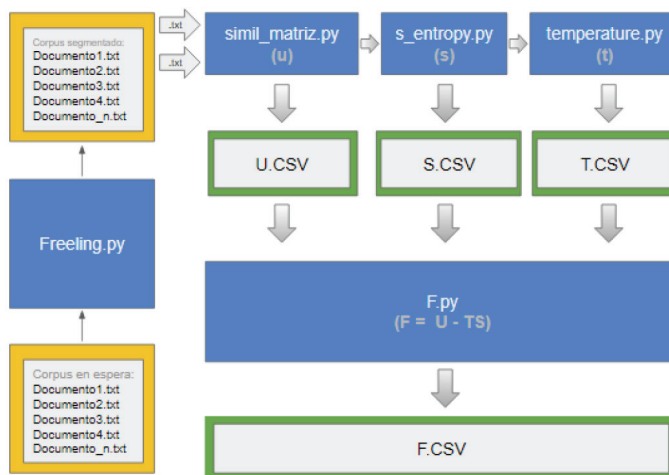


Figura 2 – Método de solución

Ambas oraciones son procesadas con el etiquetador morfosintáctico Freeling, a través del cual se extraen los lemas y se eliminan *stopwords*.

Con esta información se emplea la Red Neuronal Recurrente de Hopfield, de la cual se obtiene una matriz diagonal que permite comparar dos oraciones para identificar los *términos (palabras)* comunes.

Considérense como ejemplo los siguientes dos enunciados:

- Enunciado 1: “Bienvenidos a la Casa de la Primavera, donde tienen el mejor clima y a las mejores personalidades.”
- Y enunciado 2: “Bienvenidos a la Ciudad de la Eterna Primavera, donde tienen el mejor clima y a las mejores personalidades.”

En la matriz diagonal (Figura 3) se coloca el índice “1” a los términos repetidos en ambas oraciones y “0” si no coinciden.

	bienvenido	ciudad	eterna	primavera	donde	tener	mejor	clima	mejor	personas
bienvenido	1	0	0	0	0	0	0	0	0	0
casa	0	0	0	0	0	0	0	0	0	0
eterna	0	0	1	0	0	0	0	0	0	0
primavera	0	0	0	1	0	0	0	0	0	0
donde	0	0	0	0	1	0	0	0	0	0
tener	0	0	0	0	0	1	0	0	0	0
mejor	0	0	0	0	0	0	1	0	0	0
clima	0	0	0	0	0	0	0	1	0	0
mejor	0	0	0	0	0	0	0	0	1	0
personas	0	0	0	0	0	0	0	0	0	1

Figura 3 – Matriz de espines (Energía Textual)

Se obtiene el tamaño de la diagonal (en este caso 10 elementos) y la frecuencia de ocurrencia de los índices con valores positivos (en este caso 9 elementos). Dicho tamaño de la diagonal representa el nivel de semejanza de un par de enunciados. Por lo tanto, la similitud semántica entre un par de enunciados es proporcional a la cantidad de valores que existen dentro de la diagonal principal respecto a su cantidad de índices con valores positivos.

La Energía Textual se genera mediante el conteo de los valores de activación positivos dentro de la diagonal principal respecto a la totalidad de elementos existentes dentro de la misma. El resultado es el mínimo local que converge la diagonal principal de la matriz bidimensional que emplea la Red de Hopfield, basándose en el comportamiento de los valores de activación del Modelo de Ising. Retomando el ejemplo, el cálculo del valor de la Energía Textual (**U**) devuelve un resultado de **0.90%**.

Posteriormente se calcula la Entropía Textual (**S**) entre un par de cadenas de texto según la frecuencia de ocurrencia de un token y su respectivo lema. Esta información permite generar un *árbol de decisión* que se usa para indicar la semejanza de acuerdo con la Entropía de Shannon, como se muestra en la Tabla 1. Las columnas Enunciado 1 y Enunciado 2 muestran los términos de dichos enunciados, la columna Lemas, el término lematizado y **m** y **n** indican con el valor 1 que el término aparece en ambas oraciones y con el valor 0, lo contrario.

Enunciado 1	Lema	m	Enunciado 2	Lema	n
Bienvenidos	bienvenido	1	Bienvenidos	bienvenido	1
casa	casa	0	ciudad	ciudad	0
eterna	eterna	1	eterna	eterna	1
:	:	:	:	:	:
clima	clima	1	clima	clima	1
mejores	mejor	1	mejores	mejor	1
personas	persona	1	personas	persona	1

Tabla 1 – Entropía de información

De tal manera que la entropía entre el par de enunciados debe ser una medida de distancia que se genera haciendo uso de un árbol de decisión basado en la teoría de la información de Shannon (Figura 4).

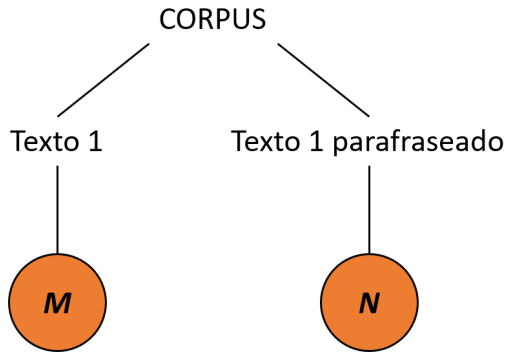


Figura 4 – Árbol de decisión ID3

En este caso, el valor de **S** es igual a **0%**, ya que para medir la distancia se identifica cuál de los dos valores (**m** ó **n**) es mayor y es ocupado como el minuendo para realizar una sustracción, siendo el valor de **n** el sustraendo.

La Temperatura Textual (**T**) se calcula utilizando el coseno suave definido de la siguiente fórmula:

$$\text{soft_cosine}_1(a, b) = \frac{\sum_{i,j}^N s_{ij} a_i b_j}{\sqrt{\sum_{i,j}^N s_{ij} a_i a_j} \sqrt{\sum_{i,j}^N s_{ij} b_i b_j}}$$

En este caso, el Coseno Suave un valor de **0.90**. Al invertir el sentido de su valor se obtiene una medida de similitud.

$$T = \text{Coseno_Suave}(\text{Enunciado 1}, \text{Enunciado 2})$$

$$T = 1 - T'$$

$$T = 1 - 0.90$$

$$T = \mathbf{0.10}$$

Estos valores pertenecen al Contexto de Afinidad Libre (basado análogamente en la Energía Libre del Modelo de Ising), y que en términos de asociación se definen como:

$$F = U - (T \cdot S)$$

Y finalmente, en términos de disociación como:

$$F = 1 - F'$$

Tal contexto proporciona dos variables para generar una Distribución Semántica, la primera es la Asociación, la cual es un índice de similitud textual y la segunda es la Disociación, medida que sirve para estimar la distancia en un par de piezas de texto. A continuación, se muestra un ejemplo en la Tabla 2.

Entropía (S)	Energía (U)	Temperatura (T)	F (asociación)	$F = U - (T \cdot S)$
0	0.9	0	1	0

Tabla 2 – Contexto de Afinidad Libre

Por lo tanto, el resultado en términos de Asociación es $F = 0.90$ y en términos de Disociación, $1 - F = 0.10$. Este método computacional genera finalmente un Contexto de Afinidad Libre que es útil para detectar paráfrasis.

5. Experimentación y resultados

Para la evaluación del modelo se usó el *Microsoft Research Paraphrase Corpus*. Este corpus fue originalmente evaluado por dos jueces humanos que determinaban si las oraciones eran lo suficientemente cercanas en el sentido semántico como para ser consideradas paráfrasis. Un tercer juez intervenía para resolver desacuerdos existentes. Los pares considerados paráfrasis se denotan con el valor 1, y en caso contrario con el valor 0.

En este trabajo se tomaron sólo 2,000 oraciones que se dividieron en los grupos I y J, cada uno con 1000 oraciones, debido al costo computacional que implica el cálculo de las variables de Energía, Entropía y Temperatura textual.

Una vez obtenidos los grupos de evaluación, se procedió con el cálculo respectivo de las variables de la ecuación $F = U - (T \cdot S)$. Donde U es la Energía Textual, T representa la Temperatura Textual y S corresponde a la Entropía Textual. En la Tabla 3 se muestran resultados de U , T y S con oraciones comparadas contra sí mismas y contra otras dos oraciones distintas.

#	Oración original	Oración candidata	U	T	S
1.1	PCCWs chief operating officer Mike Butcher and Alex Arena the chief financial officer will report directly to Mr So.	PCCWs chief operating officer Mike Butcher and Alex Arena the chief financial officer will report directly to Mr So.	1	1	0
1.2	PCCWs chief operating officer Mike Butcher and Alex Arena the chief financial officer will report directly to Mr So.	Rich media doubled its share increasing from 3% in Q2 2002 to 6% in Q2 2003.	0	0.05	0.235
1.3	PCCWs chief operating officer Mike Butcher and Alex Arena the chief financial officer will report directly to Mr So.	Thursday afternoon the Standard & Poors 500-stock index was trading up just 148 points or 01 percent to 104959.	0.075	0.103	0.101

#	Oración original	Oración candidata	U	T	S
2.1	<i>Ms Laffertys lawyer Thomas Ezzell told a Kentucky newspaper: My understanding of this is that there is a lower percentage of successful impregnations with frozen.</i>	<i>Ms Laffertys lawyer Thomas Ezzell told a Kentucky newspaper: My understanding of this is that there is a lower percentage of successful impregnations with frozen.</i>	1	1	0
2.2	<i>Ms Laffertys lawyer Thomas Ezzell told a Kentucky newspaper: My understanding of this is that there is a lower percentage of successful impregnations with frozen.</i>	<i>Last year Congress passed similar though less expensive buyout legislation for peanut farmers ending that Depression-era program.</i>	0	0.044	0.16
2.3	<i>Ms Laffertys lawyer Thomas Ezzell told a Kentucky newspaper: My understanding of this is that there is a lower percentage of successful impregnations with frozen.</i>	<i>Oracle on Friday launched a \$51 billion hostile takeover bid for PeopleSoft.</i>	0	0.107	0.083

Tabla 3 – Ejemplos de los valores de Energía (U), Temperatura (T) y Entropía (S)

En la Tabla 4 se muestra el valor resultante F para cada par de oraciones. Dado que representa la asociación, también fue necesario calcular el nivel de disociación definido como $1 - F$, el cual es una medida basada en términos de diferencia o distancia vectorial.

#	Entropía	Energía	Temperatura	F (asociación)	1-F (disociación)
1.1	0	1	1	1	0
1.2	0.235	0	0.05	-0.01175	1.01175
1.3	0.101	0.0756	0.103	0.065197	0.934803
2.1	0	1	1	1	0
2.2	0.16	0	0.044	-0.00704	1.00704
2.3	0.083	0	0.107	-0.008881	1.008881

Tabla 4 – Valor de F para cada par de oraciones

Por lo tanto, los pares de oraciones 1.1, 2.1 y 3.1 con valores de asociación y disociación de 1 y 0, respectivamente, determinan que la similitud (asociación) es bastante alta y la disociación es nula. Es decir, a valores de F más cercano a 1, los pares de oraciones tienden a ser más parecidos, caso contrario tienden a no serlo.

Los resultados se evaluaron utilizando las métricas de precisión y cobertura:

Precisión: esta métrica se calcula a partir del número de oraciones clasificadas correctamente por el sistema a partir del total de oraciones dentro del corpus. Es la fracción de instancias relevantes entre el total de las instancias recuperadas.

$$\text{precisión} = \frac{| \{ \text{oraciones relevantes} \} \cap | \{ \text{oraciones recuperadas} \} |}{| \{ \text{oraciones recuperadas} \} |}$$

Cobertura: también conocida como exhaustividad, es la fracción de la cantidad total de instancias relevantes que realmente se recuperaron, es decir, es la cantidad de oraciones clasificadas correctamente a partir del total de oraciones clasificadas.

$$\text{cobertura} = \frac{| \{ \text{oraciones relevantes} \} \cap | \{ \text{oraciones recuperadas} \} |}{| \{ \text{oraciones relevantes} \} |}$$

Los grupos I y J fueron sometidos a evaluación utilizando como referencia los resultados de la Energía Textual del modelo propuesto originalmente por Fernández, SanJuan & Torres-Moreno (2007), para comparar los resultados implementando la Energía Textual con la propuesta del uso de la Entropía y Temperatura Textual.

En la Tabla 5 se muestran los valores de precisión (P) y cobertura (C) para ambas implementaciones (modelo original y nuestra propuesta) para los grupos I y J, con diferentes valores de corte (Corte).

Grupo I					Grupo J			
Original		Propuesto			Original		Propuesto	
Corte	P	C	P	C	P	C	P	C
1	61	17	60	35	61	17	60	35
2	61	17	61	36	61	17	61	36
3	62	18	61	38	62	18	61	38
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
7	64	20	66	40	64	20	66	40
8	69	22	72	41	69	22	72	41
9	69	21	61	40	69	21	61	40

Tabla 5 – Umbral con mejores resultados

Se observa que los resultados de precisión y cobertura son más altos para un valor de corte igual a 8.

En la Tabla 6 se muestran resultados de experimentos, pero considerando valores de corte en el rango de 0.80 a 0.89. Se observa que los valores 0.80, 0.81, 0.87 y 0.88 fueron los umbrales con mayor recuperación de información útil para ambos grupos.

Grupo I					Grupo J			
Corte	Original		Propuesto		Original		Propuesto	
	P	C	P	C	P	C	P	C
0.80	69	20	72	41	0.80	69	20	72
0.81	68	20	72	41	0.81	68	20	72
0.82	68	20	71	40	0.82	68	20	71
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.86	69	22	59	40	0.86	69	22	59
0.87	70	22	55	40	0.87	70	22	55
0.88	70	22	54	40	0.88	70	22	54

Tabla 6 – Umbral con mejores resultados

En conclusión, se determinó que implementar entropía y temperatura textual a una Red Neuronal Recurrente de Hopfield mejora significativamente el umbral de recuperación de información, tal y como se muestra en la Tabla 7.

Energía Textual (mínimo local)			
	Precisión	Cobertura	Umbral
<i>Grupo I - Grupo J</i>	70	22	46%
<i>Grupo I - Grupo J</i>	72	41	56.6%

Tabla 7 – Umbral original de la RNR de Hopfield

$$Threshold = \frac{(precision * recall)}{2}$$

Tomando como base la información de la tabla anterior es posible determinar que el umbral de ganancia de información (Gain' Umbral F-E) incrementó un 10.5% como se muestra en la Tabla 8. El valor de umbral se encuentra dado por:

Modelo	Umbral de detección de paráfrasis
<i>Modelo previo (Energía Textual (E))</i>	46%
<i>Modelo propuesto (F = U - (T · S))</i>	56.5%

Tabla 8 – Incremento del umbral

La experimentación con el enfoque propuesto representa una mejora sobre el modelo original, que debe contrastarse con trabajos relacionados. En la Tabla 9 se muestra una comparativa entre el modelo propuesto y trabajos que hicieron uso del mismo corpus. Las medidas que se usaron para realizar esta comparación son

Exactitud (*accuracy*) y medida F1, las cuales se definen de la siguiente manera:

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Modelo	Exactitud	F1
(Socher et al., 2011)	76.8%	83.6%
(Milajevs et al., 2014)	73.0%	82.0%
(Mihalcea et al., 2006)	65.4%	75.3%
(Qiu et al., 2006)	72%	81.6%
Nuestro trabajo	48%	53.98%

Tabla 9 – Comparación de resultados

La comparación muestra que aún es necesario realizar mayor experimentación bajo el enfoque propuesto para obtener resultados competitivos. El incremento en la ganancia de información respecto al modelo anterior resulta alentador pues define una posible línea de investigación que puede continuar explorándose.

6. Conclusiones

El método propuesto permitió desarrollar un Contexto de Afinidad Libre (CAL), capaz de detectar paráfrasis y clasificar una colección de documentos de acuerdo con su género contextual. Se proponen dos nuevos patrones de comportamiento, entropía y temperatura textual respectivamente, que mejoran el proceso de detección de paráfrasis mediante una Red Neuronal Recurrente de Hopfield.

El modelo genera un Contexto de Afinidad Libre a partir del mínimo local que converge a través de una Red de Hopfield, que sirve para indicar y cuantificar la distribución semántica que pueda existir en un conjunto de documentos de texto. Se logró detectar el 56.5% de la paráfrasis existente en un corpus de pares de oraciones, generando un umbral de precisión de 72% y una cobertura del 41%. Lo que refiere, en suma, un aumento de 10.5% con respecto a la versión que utiliza únicamente Energía Textual.

La utilidad teórica de este trabajo está estrechamente relacionada con la Búsqueda y Recuperación de Información. Respecto a su utilidad práctica, el empleo del modelo desarrollado para encontrar paráfrasis se recomienda únicamente para textos cortos (como títulos o snippets), pues calcular la Energía, Temperatura y Energía Textual tiene un costo computacional elevado. Tareas de Procesamiento de Lenguaje Natural implementadas en esta investigación (por ejemplo, etiquetado PoS, lematización, detección de entidades nombradas, entre otras) tienen una complejidad cuadrática ($O(n^2)$). Por otro lado, la complejidad del coseno suave (entropía) es cuadrática para las operaciones entre vectores dispersos dado que solo toma en cuenta las dimensiones distintas a cero. Finalmente, el cálculo de la configuración de energía en el modelo de Ising se considera NP-Hard. Respecto a la complejidad de espacio, se demandan 1.3 GB y 804.82 MB de memoria para las ejecuciones del algoritmo completo, omitiendo las tareas de PLN.

La experimentación del modelo con las 2000 oraciones demoró 17.3 horas en un equipo con procesador AMD Ryzen 5 y 16 GB de RAM sin GPU dedicada.

Agradecimientos

Parte de esta investigación fue patrocinada por CONACYT (becas de posgrado y SNI), proyecto A1-S-27780, PRODEP, TecNM.

Referencias

- Copeland, M. (2016). The Difference Between AI, Machine Learning, and Deep Learning?: NVIDIA Blog. Recuperado el 8 de Octubre de 2020, de <https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/>
- Fausett, L. (1994). *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. Prentice-Hall, Inc.
- Fernández, S., Sanjuan, E., & Torres-Moreno, J.-M. (2007). Énergie textuelle de mémoires associatives. In *Actes de la 14ème conférence sur le Traitement Automatique des Langues Naturelles* (pp. 25–34). Toulouse: Association pour le Traitement Automatique des Langues. <https://hal.archives-ouvertes.fr/hal-01320370>
- Firth, J. R. (1957). A Synopsis of Linguistic Theory 1930-55. In *F. R. Palmer (ed.). 1968. Selected papers of J.R. Firth 1952-59*, (pp. 1-32). <https://www.semanticscholar.org/paper/A-Synopsis-of-Linguistic-Theory-1930-1955%22-in-in-Firth/32f140fbb9514fd3ead5177025c467b50896db30>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT press. <https://www.deeplearningbook.org/>
- Grzymala-Busse, J. (1993). Selected Algorithms of Machine Learning from Examples. *Fundam. Informaticae*, 18, 193-207.

- Hopfield, J. (5 de 1982). Neural Networks and Physical Systems with Emergent Collective Computational Abilities. In *Proceedings of the National Academy of Sciences of the United States of America* (vol.79, pp. 2554-8). <https://doi.org/10.1073/pnas.79.8.2554>
- Lan, W., & Xu, W. (2018). Neural Network Models for Paraphrase Identification, Semantic Textual Similarity, Natural Language Inference, and Question Answering. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 3890–3902). Association for Computational Linguistics. <https://www.aclweb.org/anthology/C18-1328>
- Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-Based and Knowledge-Based Measures of Text Semantic Similarity. In *Proceedings of the 21st National Conference on Artificial Intelligence* (vol. 1, pp. 775–780). AAAI Press.
- Milajevs, D., Kartsaklis, D., Sadrzadeh, M., & Purver, M. (2014). Evaluating Neural Word Representations in Tensor-Based Compositional Settings. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 708–719). Association for Computational Linguistics. <https://doi.org/10.3115/v1/D14-1079>
- Mohamed, I., & Gomaa, W. (2019). Exploring the Recent Trends of Paraphrase Detection. *International Journal of Computer Applications*, 182, 1-5. <https://doi.org/10.5120/ijca2019918317>
- Qiu, L., Kan, M.-Y., & Chua, T.-S. (2006). Paraphrase Recognition via Dissimilarity Significance Classification. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (pp. 18–26). Association for Computational Linguistics.
- Real Academia Española (2020). Entropía. En *Diccionario de la lengua española* (23.^a ed.). Recuperado el 8 de octubre de 2020, de <https://dle.rae.es/entrop%C3%ADa>
- Reinsel, D., Gantz, J., & Rydning, J. (2018). The digitization of the world from edge to core. <https://www.seagate.com/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf>
- Sahlgren, M. (2008). The distributional hypothesis. *Rivista di Linguistica (Italian Journal of Linguistics)*, 20, 33-53.
- Socher, R., Huang, E. H., Pennington, J., Ng, A. Y., & Manning, C. D. (2011). Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Proceedings of the 24th International Conference on Neural Information Processing Systems* (pp. 801–809). Curran Associates Inc.
- Stepanov, I. (2012). Exact solutions of the one-dimensional, two-dimensional, and three-dimensional Ising models. *Nano Science and Nano Technology: An Indian Journal*, 6, 118-122.

- Torres-Moreno, J.-M., Molina, A., & Sierra, G. (2010). La energía textual como medida de distancia en agrupamiento de definiciones. *Proceedings of 10th International Conference of Statistical Analysis of Textual Data* (pp. 215-226).
- Wolf, W. P. (2000). The Ising model and real magnetic materials. *Brazilian Journal of Physics*, 30(4), 794–810. <https://doi.org/10.1590/s0103-97332000000400030>

Un enfoque de Análisis Inteligente de Datos para Apoyar la Relación con los Clientes

Oswaldo Solarte Pabón^{1,2}, José Heriberto Torres², Víctor Andrés Bucheli²

oswaldo.solartep@alumnos.upm.es; jose.heriberto.torres@correounivalle.edu.co;
victor.bucheli@correounivalle.edu.co

¹ Universidad Politécnica de Madrid, Campus de Montegancedo, 28660, Madrid España.

² Universidad del Valle, Calle 13 # 100-00, 760032, Cali, Valle del Cauca, Colombia

DOI: 10.17013/risti.39.52-66

Resumen: El análisis inteligente de datos es un componente importante en la cuarta revolución industrial. En particular, los datos generados por los clientes se pueden usar para mejorar los productos y servicios. Sin embargo, el análisis del cliente se ha enfocado más hacia el uso de datos estructurados. La mayoría de las empresas aún no aprovechan todo el potencial que ofrecen las fuentes de datos no estructuradas. Este artículo propone un enfoque para extraer indicadores de desempeño del servicio al cliente integrando técnicas de inteligencia de negocios y procesamiento del lenguaje natural (NLP), con el propósito de aprovechar fuentes de datos no estructuradas. Para validar la propuesta se realizó un caso de estudio en una empresa que ofrece servicios de salud. Se obtuvieron indicadores a partir del análisis de llamadas telefónicas y correos electrónicos escritos en español. Los resultados obtenidos muestran la utilidad del enfoque para apoyar las relaciones con los clientes.

Palabras-clave: Inteligencia de Negocios; Procesamiento del Lenguaje Natural (NLP); Gestión de relaciones con los clientes (CRM); Análisis inteligente de datos.

An Intelligent Data Analysis Approach to Support Customer Relationship

Abstract: Intelligent data analysis is an important component of the fourth industrial revolution. In particular, data generated by customers can be used to improve products and services. However, customer analysis has focused more on the use of structured data. Most companies are still not taking advantage of the full potential of unstructured data sources. This paper proposes an approach to extract customer service performance indicators integrating business intelligence and natural language processing (NLP) techniques to take advantage of unstructured data. To validate the proposal, a study case was carried out in a company that offers health services. Indicators were obtained from the analysis of phone calls and emails written in Spanish. The results obtained show the utility of the approach to support customer relationships.

Key words: Business Intelligence; Natural Language Processing (NLP); Customer Relationship Management (CRM); Smart data analysis.

1. Introducción

El análisis de datos es un componente fundamental en la cuarta revolución industrial ya que la generación de grandes volúmenes de datos está creando oportunidades para mejorar la eficiencia de los sistemas productivos (Mittal, Khan, Romero, & Wuest, 2019), (Peres, Rocha, Leitao & Barata, 2018). Adicionalmente, en esta nueva revolución los clientes y socios comerciales se integran a los procesos productivos con el fin de aportar valor agregado a las industrias (Thoben, Wiesner & Wuest, 2017).

En ese sentido, uno de los grandes desafíos para las empresas es brindar una experiencia positiva a sus clientes (Hassan, Nawaz, Lashari & Zafar, 2015). Una de las estrategias más usadas para mantener las relaciones con los clientes es la utilización de sistemas CRM (*Customer Relationship Management*). Además, la integración de varios canales de comunicación y la sincronización de diferentes fuentes de datos crea oportunidades y retos para la mejorar los sistemas CRM (Picek, Peras, & Mekovec, 2018).

Con el objetivo de apoyar las relaciones con los clientes, se han presentado varias propuestas que integran capacidades de análisis de datos a los sistemas CRM, tal como se describe en (Rodrigues Chagas et al., 2019) y (Soltani & Navimipour, 2016). Sin embargo, muchas propuestas se han enfocado solo en el análisis de datos estructurados. Las empresas aun no aprovechan todo el potencial que ofrecen las fuentes de datos no estructuradas. De acuerdo con Müller, Debortoli, Junglas & Vom-Brocke, (2016), el 80% de los datos que se generan actualmente son datos no estructurados y frecuentemente se basan en texto. Adicionalmente, las propuestas que explotan fuentes textuales en su mayoría se han enfocado en el idioma inglés. Además, la extracción de información también representa sus propios desafíos en otros idiomas, además del inglés (Névél, Dalianis, Velupillai, Savova, & Zweigenbaum, 2018).

En este artículo se propone un enfoque para extraer indicadores de desempeño del servicio al cliente integrando técnicas de inteligencia de negocios y procesamiento del lenguaje natural (NLP). Este enfoque saca ventaja de fuentes de datos estructuradas y de fuentes no estructuradas como documentos de texto escritos en español. El enfoque permite extraer información de productos, servicios, preferencias y las opiniones que el cliente tiene sobre estos. Los indicadores obtenidos se integran en un esquema dimensional orientado al soporte de consultas para el análisis de datos. Para validar la propuesta se aplicó un caso de estudio en una empresa que ofrece una red de servicios de salud para conectar a los consumidores y productores de estos servicios.

El resto de este documento está organizado de la siguiente manera: la Sección 2 describe algunos trabajos relacionados. En la sección 3 se presenta detalladamente el enfoque propuesto. En la Sección 4 se describe un caso de estudio que se realizó para validar la utilidad del enfoque. Finalmente, la Sección 5 incluye conclusiones y trabajos futuros.

2. Trabajos relacionados

En este artículo las propuestas que usan análisis de datos para apoyar las relaciones con los clientes se agrupan en tres categorías: las que integran sistemas CRM con bodegas de datos, CRM con Minería de datos y las que integran CRM con técnicas de NLP.

2.1. Integración de sistemas CRM con bodegas de datos

La integración de bodegas de datos con los sistemas CRM proporciona información integrada que facilita el análisis de datos del cliente. Una de las primeras propuestas para integrar los sistemas CRM con una bodega de datos se describe en (Cunningham, Song & Chen, 2004), donde se propone el uso del modelo dimensional para crear herramientas que soporten consultas del servicio al cliente. Continuando con el trabajo anterior, en (Cunningham & Song 2007) se proponen los requerimientos que debe cumplir una bodega de datos orientada al análisis de datos del cliente. De acuerdo con Khan, Rao & Shahzad (2019), la integración de los sistemas CRM con una bodega de datos reduce el costo de adquirir nuevos clientes y del proceso de ventas. Una bodega de datos orientada al cliente crea ventaja competitiva ya que aumenta la cantidad de clientes y la tasa de retención (Handzic, Ozlen & Durmic, 2014).

En Ishaya & Folarin (2012), se propone un enfoque para integrar varios sistemas operacionales con información de los clientes en una empresa de telecomunicaciones. En este enfoque, la información se analiza en tres pasos: la identificación del perfil de los clientes, segmentación y la identificación de campañas de marketing. Similarmente, en Maji & Sen (2016), se propone un enfoque basado en inteligencia de negocios para analizar el impacto que tienen las campañas de marketing en los clientes.

Por su parte, en Rodríguez, Flores & La Rosa (2016) se propone un enfoque de inteligencia de negocios para una empresa que distribuye y comercializa refrescos. El enfoque permite obtener indicadores comerciales para conocer las zonas y rutas donde se realizan más ventas y así poder asignar los vendedores de una forma más eficiente.

2.2. Integración de técnicas minería de datos y sistemas CRM

En los últimos años el uso de técnicas de minería de datos para analizar información de los clientes se ha expandido rápidamente (Rodrigues Chagas et al., 2019). En Farajian & Mohammadi (2010) se propone una estrategia para analizar el comportamiento del cliente usando algoritmos de *clustering* y reglas de asociación en el sector financiero. De manera similar, en Devi & Rajagopalan (2012) se analiza el comportamiento del cliente en el proceso de ventas.

En Kadiyala & Srivastava (2011) se propone un marco de trabajo para identificar las técnicas de minería de datos relacionadas con los sistemas CRM. Por su parte Ahn, Ahn, Oh, & Kim, (2011) proponen un modelo de clasificación para apoyar la venta cruzada en el dominio de las telecomunicaciones móviles. De acuerdo con Rahman & Kan (2017), las técnicas de minería de datos son usadas comúnmente por las empresas de telecomunicaciones, ya que este es un mercado donde los clientes se pueden cambiar

fácilmente de operador. Finalmente, en Bahari & Elayidom (2015), se propone un modelo predictivo aumentar la retención de clientes en el sector financiero.

2.3 Integración de técnicas de NLP y sistemas CRM

Con el objetivo de extender el análisis de datos a fuentes no estructuradas, en los últimos años se han propuesto enfoques que integran técnicas de NLP con los sistemas CRM. En Gutfreund (2017) se propone un enfoque que usa minería de texto para analizar la gestión de relaciones con clientes de Elsevier¹. Este enfoque busca predecir y reducir el abandono prematuro de los clientes. Por su parte, en Müller et al., (2016) se aborda el concepto de Social CRM (SCRM), que se refiere a la capacidad de integrar información de redes sociales con los sistemas CRM. En esta propuesta se desarrolló un prototipo para el servicio de tecnologías de información de la universidad de la Florida. El prototipo analiza descripciones textuales de casos reportados por los usuarios a través correo electrónico, llamadas telefónicas y redes sociales.

Una propuesta similar se describe en Zhou et al. (2018), donde se presenta un modelo para analizar opiniones que hacen los clientes a través de aplicaciones móviles de diferentes productos. El modelo sugiere las opiniones más relevantes que los desarrolladores de productos deberían tener para ajustar sus diseños y productos a las necesidades de los clientes. De igual manera, en Khan, Rao & Shahzad (2019) analizan datos de Twitter para extraer opiniones acerca una compañía aérea. Por último, en Kauffmann et al. (2019) se extrae opiniones de los clientes de comentarios escritos en Amazon.

La Figura 1 muestra un resumen de las propuestas descritas anteriormente. Para cada propuesta se muestra el proceso que se quería analizar (segmentación de clientes, retención de clientes, adquirir clientes, etc) y el dominio (telecomunicaciones, financiero, transporte, etc).

3. Enfoque para apoyar la relación con los clientes

La Figura 2 muestra el enfoque propuesto para apoyar la relación con los clientes. Este enfoque combina técnicas de inteligencia de negocios y procesamiento del lenguaje natural para extraer indicadores de desempeño de diferentes fuentes de datos. Este enfoque está formado por cuatro componentes: Selección de fuentes de datos, Procesamiento de datos estructurados, Procesamiento de datos no estructurados y el componente de Integración de datos.

3.1. Seleccionar fuentes de datos

Este componente permite la identificación de fuentes de datos tanto estructuradas como no estructuradas que se han generado a partir de la comunicación con los clientes. Los datos estructurados provienen principalmente de las bases de datos relacionales asociadas al CRM de la organización, hojas de cálculo u otros archivos estructurados. Para los datos no estructurados esta propuesta se enfoca en analizar información escrita en lenguaje natural como correos electrónicos, transcripción de registros de llamadas telefónicas, y otros documentos de texto escritos en español.

¹ <https://www.elsevier.com>

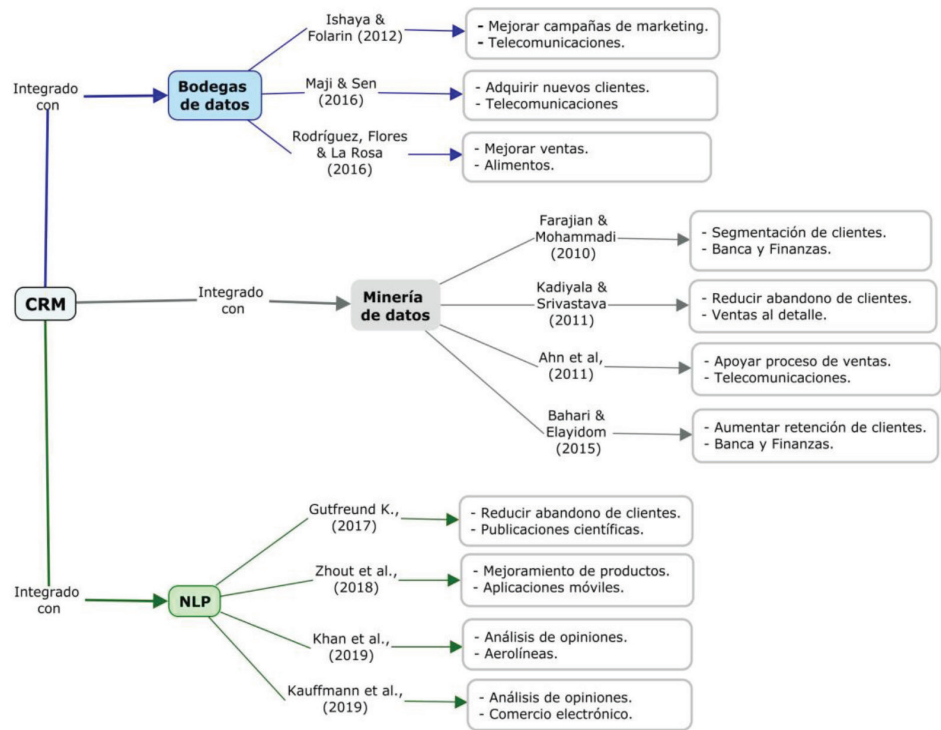


Figura 1 – Propuestas que utilizan análisis de datos para apoyar la relación con los clientes

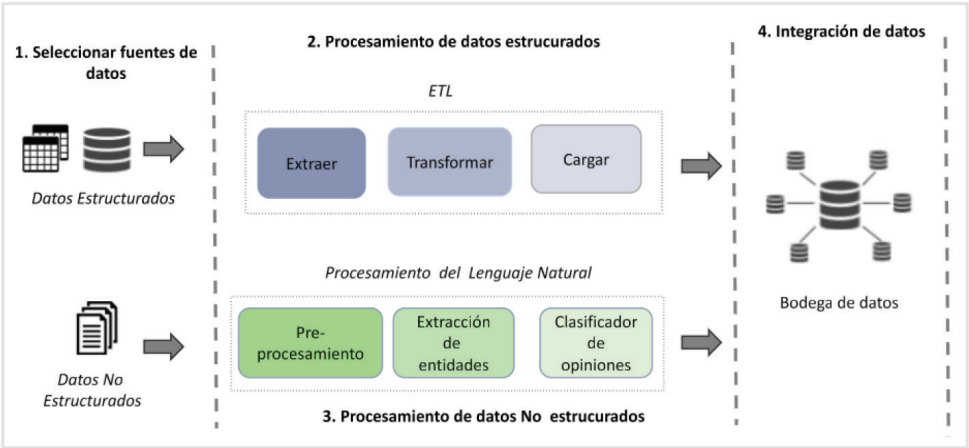


Figura 2 – Enfoque propuesto

3.2. Procesamiento de datos estructurados

En esta fase se aplican técnicas de inteligencia de negocios y bodegas de datos con el fin de obtener indicadores asociados a la relación con los clientes. Como entrada se toman diferentes registros estructurados y estos se transforman en indicadores orientados al análisis de información. La Figura 3 muestra los pasos que se ejecutan en el procesamiento de datos estructurados.

Los indicadores obtenidos se integran y almacenan en un modelo dimensional. Este modelo se describe detalladamente en la sección 3.4. Algunos indicadores que se obtienen a partir del análisis de datos estructurados son:

- Número de transacciones que se resuelven en el primer contacto o primera llamada.
- Número de transacciones que son escaladas a otro equipo que tomará la responsabilidad de resolver la transacción.
- Número de llamadas telefónicas en las que se obtuvo una venta.
- Tiempo promedio que lleva resolver una transacción.

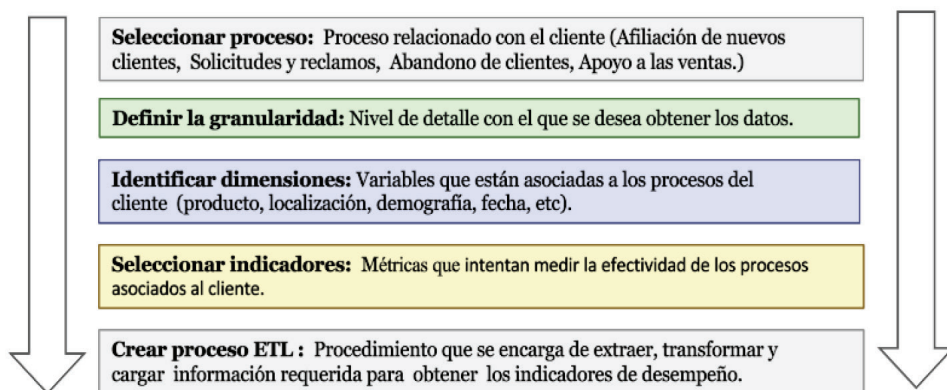


Figura 3 – Procesamiento de datos estructurados

3.3. Procesamiento de datos no estructurados

En este componente se aplican técnicas de NLP para extraer información de fuentes de datos basadas en texto, tal como se muestra en la Figura 4. Este componente puede procesar correos electrónicos, llamadas telefónicas previamente transcritas a texto, o cualquier otro archivo basado en texto.

De acuerdo con la Figura 4, este componente se divide en tres fases: Pre-procesamiento de datos, Extracción de entidades nombradas y un clasificador de opiniones. Los resultados obtenidos se almacenan en un modelo dimensional que sirve como base para el análisis de datos.

Pre-procesamiento de datos

En esta fase se aplican técnicas para limpiar y normalizar los documentos de texto:

- *Tokenización*: divide el texto en sentencias de texto y tokens. Un token es un elemento atómico de una sentencia como palabras, números o siglas.
- *Eliminar Stop Words*: se eliminan palabras que son muy frecuentes en el texto y no aportan valor semántico al texto (ejemplo: la, los, de, del, etc)
- *Normalización*: convierte el texto a minúsculas y luego elimina acentos.

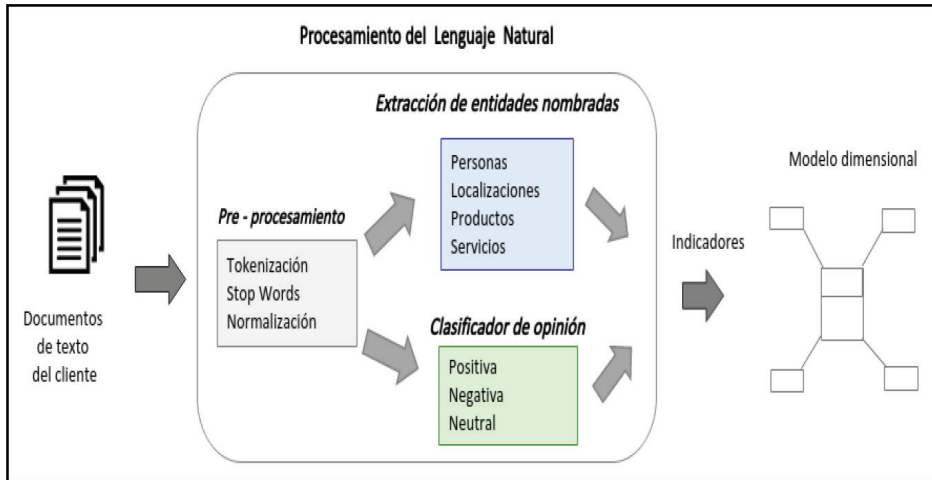


Figura 4 – Procesamiento de datos no estructurados

Reconocimiento de entidades nombradas

En esta fase se aplica el reconocimiento de entidades nombradas (Named Entity Recognition o NER) con el objetivo de extraer personas, productos, servicios o localizaciones relacionadas con los clientes. Para hacer esta tarea se utilizó el estándar de clasificación de entidades propuesto en CoNLL-2003 para el idioma español (Sang & De Meulder, 2003). En el enfoque propuesto las entidades extraídas se clasifican usando las siguientes etiquetas:

- **PER**: etiqueta usada para hacer referencia a personas.
- **ORG**: identifica las organizaciones mencionadas por el cliente.
- **MISC**: productos y servicios ofrecidos por la organización
- **LOC**: localizaciones y ciudades mencionadas por el cliente.

Para extraer entidades se utiliza *Freeling*², una librería de código abierto para el procesamiento de lenguaje natural y que tiene soporte para el idioma español. La Figura

² <http://nlp.lsi.upc.edu/freeling/index.php/node/1>

5 muestra un ejemplo donde se toma como entrada un correo electrónico enviado por el cliente y la salida es un conjunto de entidades extraídas del texto.

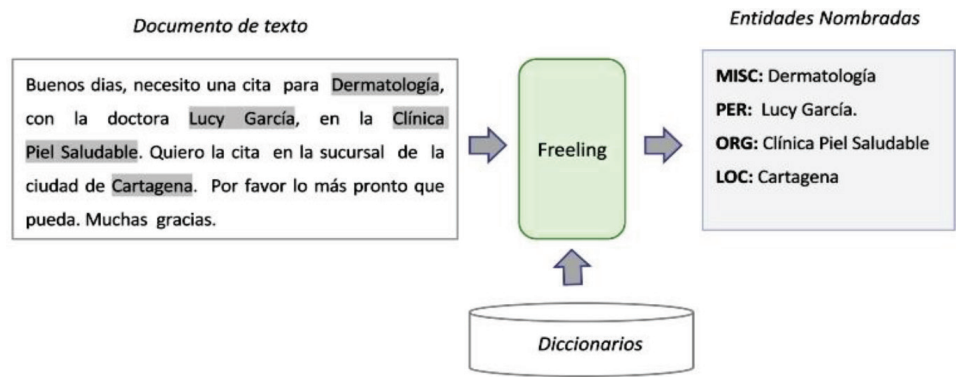


Figura 5 – Extracción de entidades nombradas

Clasificador de opinión

En esta fase se extrae la opinión que los clientes tienen respecto a los productos o servicios que se identificaron en la fase anterior. Para extraer esta opinión se usa un clasificador automático que etiqueta un texto en tres categorías: positivo, negativo o neutral. El clasificador se basa en un modelo que se implementó usando los algoritmos Naive Bayes y Árboles de decisión.

El modelo de clasificación se entrenó con un conjunto de 3200 textos escritos en español. Estos textos fueron extraídos de correos electrónicos y llamadas telefónicas del centro de atención al cliente de la organización Previser, que ofrece servicios de salud a sus clientes. En esta organización se aplicó un caso de estudio del cual se hablará con detalle en la sección 3.4. El conjunto de datos fue anotado y validado manualmente por cinco personas expertas del área de servicio al cliente. Al finalizar el proceso de anotación se hizo un consenso entre todos los anotadores para determinar la etiqueta final. La Tabla 1 muestra tres ejemplos de textos con su respectiva etiqueta de opinión.

Texto	Etiqueta
No ha sido posible contactar con usted, esta muy mala la comunicación. Para pedir una cita siempre suena ocupado, está pésimo el servicio.	Negativo
Para pedir una cita médica con el cardiólogo. Tenemos disponible con el doctor Pedro García. Perfecto , el siempre me ha atendido amablemente .	Positivo
Señora, buenas tardes. Su cita se agendó para el viernes a las 2:00 PM. Esperamos su confirmación y llegar 10 minutos antes.	Neutral

Tabla 1 – Textos anotados manualmente con su respectiva etiqueta

3.4. Integración de datos

En este componente se integran los indicadores obtenidos en las fases anteriores. La integración se basa en el uso de elementos del modelo dimensional como *datamarts*, tablas de hechos y dimensiones. Con estos elementos se modela una bodega de datos que contiene información de fuentes estructuradas y no estructuradas.

El componente de integración de datos busca extender la capacidad de análisis de datos usando el potencial de los datos no estructurados. La integración se hace a través de dimensiones compartidas, es decir aquellas dimensiones que aparecen tanto en los datos estructurados como en los no estructurados, tal como se muestra en los siguientes ejemplos. La Figura 6 muestra un *datamart* que representa el proceso “Apoyo a las ventas”. Este *datamart* se genera a partir de datos estructurados y se pueden obtener indicadores como el porcentaje de ventas por productos, por localización, productos más y menos vendidos, etc.

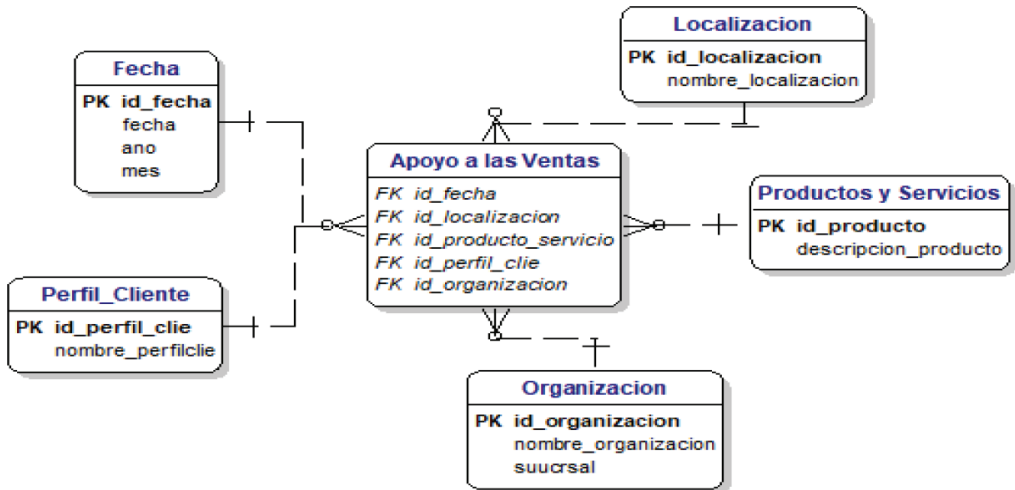


Figura 6 – *Datamart* para el proceso de apoyo a las ventas

Los indicadores obtenidos a partir de la Figura 6 describen cantidades y porcentajes de los productos y servicios vendidos, sin embargo, dichos indicadores son limitados ya que no permiten conocer los factores que podrían estar influyendo en el proceso de ventas.

Por su parte, la Figura 7 muestra un *datamart* que representa el proceso de “Quejas y reclamos del cliente”, el cual se genera a partir del análisis de datos no estructurados. Este *datamart* contiene indicadores de las opiniones que el cliente ha expresado acerca de productos, servicios o del proceso de ventas en general. Estos indicadores se obtienen mediante la extracción de entidades nombradas y el clasificador de opiniones. De este

datamart se pueden obtener indicadores como el porcentaje de opiniones positivas y negativas, productos con más quejas o reclamos, quejas más comunes mencionadas por los clientes, productos mejor calificados, etc.

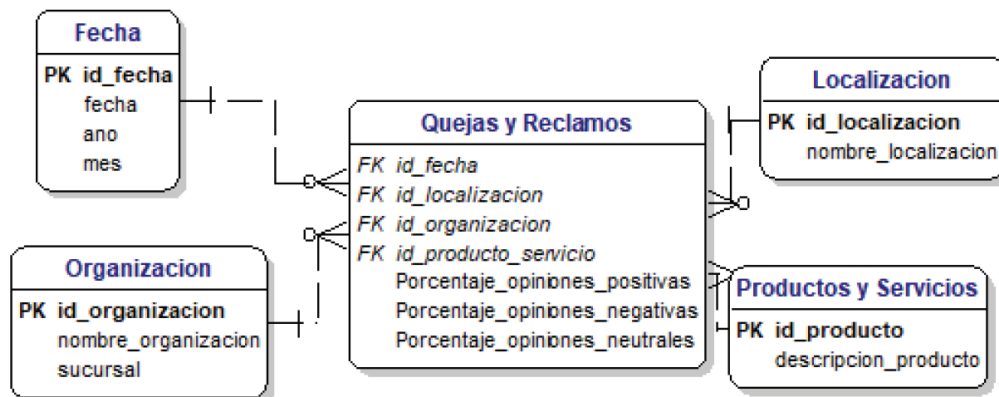


Figura 7 – *Datamart* para el proceso quejas y reclamos

Los *datamart* de las Figuras 6 y 7 se pueden integrar mediante el uso de dimensiones compartidas como productos, servicios o localización. Estas dimensiones aparecen tanto en el análisis de datos estructurados como no estructurados. A través de esta integración se extiende el proceso de apoyo a las ventas ya que al analizar las opiniones del cliente se conocen ciertos factores que podrían estar afectando el proceso de ventas. Usando la extracción de entidades nombradas y el análisis de opiniones se pueden establecer relaciones entre las opiniones del cliente y los resultados en el proceso de ventas. En este sentido, la integración de datos no estructurados permite obtener no solamente resultados de ventas, sino también los factores que están influyendo en esos resultados.

Así como el ejemplo descrito anteriormente, el enfoque propuesto permite modelar diferentes procesos asociados al cliente aprovechando el potencial de los datos no estructurados. Además, los indicadores obtenidos por medio del enfoque se pueden utilizar en diferentes tareas clave en la industria 4.0 como son el mejoramiento continuo de productos y servicios, la personalización de productos, la personalización de campañas de marketing y el constante mejoramiento de las relaciones con los clientes.

4. Caso de Estudio

El caso de estudio se realizó en la organización Previser³, una compañía colombiana que ofrece una red de servicios que conecta a los consumidores finales con diferentes proveedores de productos de salud. Previser tiene un área de servicio al cliente que cuenta con un sistema CRM. Además, diariamente reciben cientos de llamadas telefónicas y correos electrónicos con las solicitudes de los clientes. Para validar el enfoque se hicieron

³ <https://www.previser.com.co>

dos experimentos. El primero para evaluar el reconocimiento de entidades nombradas y el segundo, el clasificador de opiniones.

4.1. Experimento 1: extracción de entidades nombradas

- **Objetivo:** validar el reconocimiento de entidades nombradas en textos asociados al cliente. Como *dataset* de prueba se tiene un conjunto de 1000 textos previamente etiquetados, que corresponden al 20 % de corpus de entrenamiento que contiene alrededor de 5000 documentos de texto.
- **Resultados:** la Tabla 2 muestra los resultados obtenidos separados por cada etiqueta. El reconocimiento de entidades se hizo con la herramienta *Freeling*. De acuerdo con la Tabla 2, el rendimiento para extraer las diferentes categorías de entidades es similar en todas las etiquetas. Una de las dificultades al desarrollar el experimento fue no tener un corpus disponible en español para el dominio del servicio al cliente. Por lo tanto, la primera decisión fue construir un corpus de prueba. El tamaño de este corpus influyó en los resultados obtenidos, a medida que el corpus crezca, la capacidad para encontrar diferentes entidades asociadas al servicio al cliente puede mejorar.

	PER	ORG	MISC	LOC
Precision	0,89	0,86	0,90	0,87
Recall	0,88	0,88	0,87	0,86
F-score	0,88	0,87	0,89	0,85

Tabla 2 – Resultados Experimento 1

4.2. Experimento 2: clasificador de opiniones

- **Objetivo:** validar el clasificador de opiniones que categoriza la polaridad del texto en: positivo, negativo o neutral. Se usó el mismo conjunto de datos utilizado en el primer experimento.
- **Resultados:** la Tabla 3 muestra los resultados obtenidos separados por cada etiqueta. Las pruebas se realizaron con dos clasificadores, Naive Bayes y *Decision Tree*. De acuerdo con la Tabla 3, el algoritmo basado en árboles de decisión tiene un rendimiento superior en la tarea de extracción de opiniones. Los parámetros utilizados para el árbol de decisión fueron (*criterion='entropy', min_samples_split=20,min_samples_leaf=5,max_depth=4*).

La mayor dificultad en la extracción automática de opiniones del cliente, se presentó en textos demasiado largos. En este caso, el clasificador por lo general tiende a etiquetarlos como neutros. Por otro lado, en el idioma español se pueden encontrar gran variedad de expresiones para opiniones similares expresadas por el cliente. Por lo tanto, el mejoramiento del corpus de entrenamiento es un paso fundamental para mejorar los resultados obtenidos.

	Árboles de decisión	Naive Bayes
<i>Precision</i>	0,89	0,86
<i>Recall</i>	0,87	0,85
<i>F-score</i>	0,88	0,85

Tabla 3 – Resultados Experimento 2

Los resultados que se muestran en las Tablas 2 y 3 muestran una métrica F-score mayor al 85% en ambos casos, esto indica que las técnicas de NLP utilizadas fueron útiles para extraer información relacionada con los clientes. Aunque estos resultados se pueden mejorar, el principal objetivo del caso de estudio era probar la utilidad del enfoque propuesto para extraer entidades y opiniones asociadas al cliente.

En el experimento 1 se extraen las entidades mencionadas por el cliente (*e.g* productos o servicios). En el experimento 2 se evalúa la opinión que el cliente tiene sobre dichas entidades. De esta manera se forma la pareja “entidad: opinión”. Esta pareja se usa para extender la información obtenida con los datos estructurados, los cuales solo permiten visualizar la cantidad de servicios solicitados, pero no la opinión de los clientes sobre dichos servicios. Analizar datos no estructurados permite obtener indicadores adicionales como la identificación de servicios con un alto número de opiniones negativas, los servicios mejor calificados o las quejas más comunes expresadas por el cliente. La Figura 8 muestra los servicios con más opiniones negativas que se obtuvieron en el caso de estudio. Esta información se puede aprovechar para entender mejor los resultados del proceso de ventas y así generar estrategias de mejoramiento del servicio.

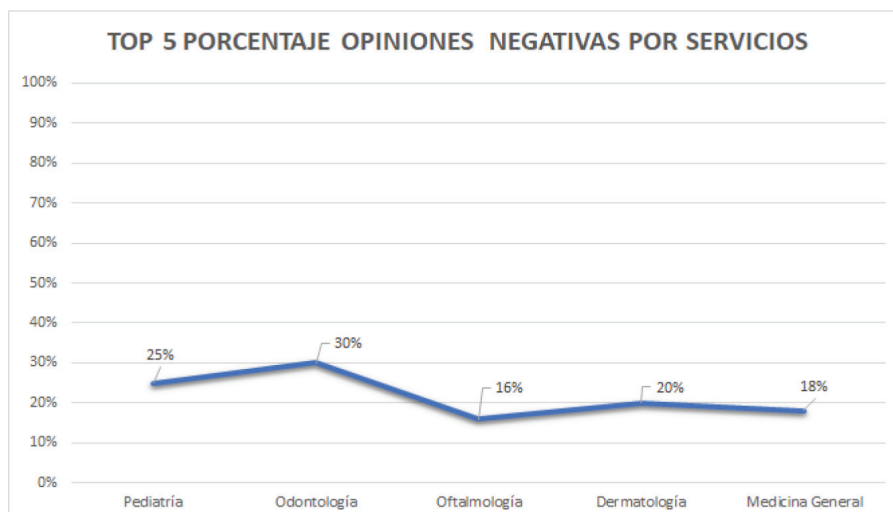


Figura 8 – Servicios con más opiniones negativas

La extracción de entidades y el clasificador de opiniones son técnicas del NLP utilizadas para extraer información relacionada con los gustos y opiniones expresados por los clientes en documentos de texto. Estas dos técnicas se complementan ya que los indicadores extraídos se integran en la bodega de datos, tal como se describió en la sección 3.3. Estos indicadores se pueden usar para el mejoramiento de procesos y productos. Así, el análisis de datos se transforma en ventaja competitiva, lo cual va en el camino de la cuarta revolución industrial.

5. Conclusiones y trabajo Futuro

En este artículo se propuso un enfoque para extraer indicadores de desempeño del servicio al cliente integrando técnicas de inteligencia de negocios y NLP. En particular, las técnicas de NLP aprovechan el potencial que tienen las fuentes de datos basadas en texto. Estas fuentes de datos contienen información que revela los gustos y opiniones del cliente. Extraer esta información e integrarla con otras fuentes de datos puede ser una ventaja competitiva que pueden aprovechar las empresas para mejorar sus procesos.

El análisis de diferentes fuentes de datos es primordial para apoyar el desarrollo de la cuarta revolución industrial. El enfoque propuesto mostró cómo se puede aprovechar fuentes de datos estructuradas y no estructuradas para obtener indicadores que pueden ser útiles para apoyar las relaciones con los clientes. Apoyar esas relaciones es importante ya que, en la nueva revolución industrial, los clientes pueden convertirse en fuentes que generan valor agregado para las empresas.

La gran variedad de datos que se generan actualmente implica que se deben combinar diferentes técnicas y estrategias para cumplir con los requerimientos de las empresas. En este artículo se propuso un enfoque que combina técnicas de inteligencia de negocios, bodegas de datos y NLP para tratar el problema de los datos no estructurados. Las técnicas de la inteligencia artificial como el NLP son importantes para el desarrollo de aplicaciones en la cuarta revolución industrial. Como trabajo futuro se planea probar otros algoritmos para el reconocimiento de entidades y opiniones con el fin de aumentar el rendimiento de nuestro enfoque. También seguir probando otras técnicas de NLP que se puedan aplicar al análisis de datos de los clientes.

Referencias

- Ahn, H., Ahn, J. J., Oh, K. J., & Kim, D. H. (2011). Facilitating cross-selling in a mobile telecom market to develop customer classification model based on hybrid data mining techniques. *Expert Systems with Applications*, 38(5), 5005–5012. <https://doi.org/10.1016/j.eswa.2010.09.150>
- Bahari F., & Elayidom S., (2015). An Efficient CRM-Data Mining Framework for the Prediction of Customer Behaviour. *Procedia Computer Science*, 46, 725–731. <https://doi.org/10.1016/j.procs.2015.02.136>
- Cunningham, C., Song, I. Y., & Chen, P. P. (2004). Data warehouse design to support customer relationship management analyses. *DOLAP: Proceedings of the ACM Int. Workshop on Data Warehousing and OLAP*, 17(June), 14–22. <https://doi.org/10.4018/978-1-60566-058-5.ch042>

- Cunningham, C., & Song, I.-Y. (2007). A Taxonomy of Customer Relationship Management Analyses for Data Warehousing. *The 26th International Conference on Conceptual Modeling*, 83, 97–102. Retrieved from <http://dl.acm.org/citation.cfm?id=1386972>
- Devi, P. I., & Rajagopalan, S.P. (2012). Analysis of Customer Behavior using Clustering and Association Rules. *International Journal of Computer Applications*, 43(23), 19–26.
- Farajian, M. A., & Mohammadi, S. (2010). Mining the Banking Customer Behavior Clustering and Association Rule Methods. *International Journal of Industrial Engineering & Product Research*, 21(4), 239–245. <http://ijiepr.iust.ac.ir/article-1-241-en.html>
- Gutfreund, K. (2017). Big Data Techniques for Predictive Business Intelligence. *Journal of Advanced Management Science*, 5(2), 158–163. <https://doi.org/10.18178/joams.5.2.158-163>
- Handzic, M., Ozlen, K., & Durmic, N. (2014). Improving Customer Relationship Management Through Business Intelligence. *Journal of Information & Knowledge Management*, 13(2).
- Hassan, R. S., Nawaz, A., Lashari, M. N., & Zafar, F. (2015). Effect of Customer Relationship Management on Customer Satisfaction. *Procedia Economics and Finance*, 23, 563–567. [https://doi.org/10.1016/s2212-5671\(15\)00513-4](https://doi.org/10.1016/s2212-5671(15)00513-4)
- Ishaya, T., & Folarin, M. (2012). A service oriented approach to Business Intelligence in Telecoms industry. *Telematics and Informatics*, 29(3), 273–285. <https://doi.org/10.1016/j.tele.2012.01.004>
- Kadiyala, S. S., & Srivastava, A. (2011). Data Mining for Customer Relationship Management. *International Business & Economics Research Journal*, 1(6), 61–70.
- Khan, D. M., Rao, T. A., & Shahzad, F. (2019). The Classification of Customers' Sentiment using Data Mining Approaches. *Global Social Sciences Review*, IV(IV), 146–156. [https://doi.org/10.31703/gssr.2019\(iv-iv\).19](https://doi.org/10.31703/gssr.2019(iv-iv).19)
- Kauffmann, E., Peral, J., Gil, D., Ferrández, A., Sellers, R., & Mora, H. (2019). Managing marketing decision-making with sentiment analysis: An evaluation of the main product features using text data mining. *Sustainability*, 11(15), 1–19. <https://doi.org/10.3390/su11154235>
- Maji, G., & Sen, S. (2016). Data warehouse based analysis on CDR to retain and acquire customers by targeted marketing. 5th *International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*, Noida, India. <https://doi.org/10.1109/icrito.2016.7784955>
- Mittal, S., Khan, M. A., Romero, D., & Wuest, T. (2019). Smart manufacturing: Characteristics, technologies and enabling factors. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 233(5), 1342–1361. <https://doi.org/10.1177/0954405417736547>

- Müller, O., Debortoli, S., Junglas, I., & Vom-Brocke, J. (2016). Using text analytics to derive customer service management benefits from unstructured data. *MIS Quarterly Executive*, 15(4), 243–258.
- Névéol, A., Dalianis, H., Velupillai, S., Savova, G., & Zweigenbaum, P. (2018). Clinical Natural Language Processing in languages other than English: Opportunities and challenges. *Journal of Biomedical Semantics*, 9(1), 1–13. <https://doi.org/10.1186/s13326-018-0179-8>
- Peres, R. S., Rocha, D. A., Leitao, P., & Barata, J. (2018). IDARTS Towards intelligent data analysis and real-time supervision for industry 4.0. *Computers in Industry*, 101, 138–146. <https://doi.org/10.1016/j.compind.2018.07.004>
- Picek, R., Peras, D., & Mekovec, R. (2018). Opportunities and Challenges of Applying Omnichannel Approach to Contact. In *Proceedings of 2018 4th International Conference on Information Management (ICIM)*, 231–235.
- Rahman, A., & Khan, M. N. A. (2017). An Assessment of Data Mining Based CRM Techniques for Enhancing Profitability. *International Journal of Education and Management Engineering*, 7(2), 30–40. <https://doi.org/10.5815/ijeme.2017.02.04>
- Rodrigues Chagas. B. N., Nogueira V. J. A., Reinhold, O., Lobato, F., Jacob, A. F. L., & Alt, R. (2019). Current Applications of Machine Learning Techniques in CRM: A Literature Review and Practical Implications. In *Proceedings IEEE/WIC/ACM International Conference on Web Intelligence*, 452–458. <https://doi.org/10.1109/WI.2018.00-53>
- Rodríguez Flores, F., Flores Pulido, L., & La Rosa, E. D. (2016). Inteligencia de negocios y minería de datos aplicado a la industria refresquera. *Research in Computing Science*, 126(1), 63–71. <https://doi.org/10.13053/rscs-126-1-6>
- Sang, E.T.K., & De Meulder, F. (2003). Introduction to the conll-2002 shared task: Language independent named entity recognition. In *Proceedings of CoNLL-2002*, 155–158.
- Soltani, Z., & Navimipour, N. J. (2016). Customer relationship management mechanisms: A systematic review of the state of the art literature and recommendations for future research. *Computers in Human Behavior*, 61, 667–688. <https://doi.org/10.1016/j.chb.2016.03.008>
- Thoben, K. D., Wiesner, S. A., & Wuest, T. (2017). “Industrie 4.0” and smart manufacturing—a review of research issues and application examples. *International Journal of Automation Technology*, 11(1), 4–16. <https://doi.org/10.20965/ijat.2017.p0004>
- Zhou, S., Qiao, Z., Du, Q., Wang, G. A., Fan, W., & Yan, X. (2018). Measuring Customer Agility from Online Reviews Using Big Data Text Analytics. *Journal of Management Information Systems*, 35(2), 510–539.

Mejoramiento del algoritmo ADR en una red de Internet de las Cosas LoRaWAN usando Aprendizaje de Máquina

Mauricio González-Palacio¹, Lina María Sepúlveda-Cano²,
Jhon Quiza-Montealegre¹, Juan D'Amato³

magonzalez@udem.edu.co; lmsepulveda@udem.edu.co;
jhquiza@udem.edu.co; juan.damato@gmail.com

¹ Ingeniería de Telecomunicaciones, Universidad de Medellín, 050082, Medellín, Colombia.

² Ingeniería de Sistemas, Universidad de Medellín, 050082, Medellín, Colombia.

³ Instituto Pladema, Universidad Nacional del Centro de la Provincia de Buenos Aires, B7001BBO, Tandil, Argentina.

DOI: 10.17013/risti.39.67-83

Resumen: El Internet de las Cosas es un paradigma habilitador de la Industria 4.0, donde sensores y actuadores se conectan a Internet. El protocolo LoRaWAN (Long Range Area Network) es uno de los más empleados, y es usado para transmitir información a largas distancias con mínimo consumo energético. Este protocolo implementa el esquema *Adaptive Data Rate* para mejorar la energía consumida por nodo, que al ser evaluado a través de simulaciones exhaustivas en Omnet++, ha exhibido posibilidades de mejora en el tiempo de convergencia. El presente trabajo muestra una propuesta para el mejoramiento del algoritmo ADR de tal forma que se optimice el consumo energético en redes LoRaWAN. Dentro de la propuesta se comparan diferentes modelos paramétricos y no paramétricos. Los resultados indican que los métodos basados en Máquinas de Vectores de Soporte y en Redes Neuronales Artificiales presentan la mayor exactitud, con un porcentaje por encima del 90% en las estimaciones.

Palabras-clave: Internet de las Cosas; Industria 4.0; consumo energético; LoRaWAN; Aprendizaje de Máquina

Improvement of the algorithm ADR in an Internet of Things network LoRaWAN by using Machine Learning

Abstract: The Internet of Things (IoT) is an enabling paradigm for Industry 4.0, where sensors and actuators connect to the Internet. The protocol LoRaWAN (Long Range Area Network) is one of the most used in the IoT, and its primary objective is to transmit sensor information over long distances with minimal energy consumption. This protocol implements Adaptive Data Rate scheme to optimize the energy consumed per node, which, when evaluated through exhaustive simulations in Omnet ++, has exhibited opportunities for improvement in convergence time.

The present work shows machine learning models based on parametric and non-parametric methods based on Support Vector Machines (SVM) and Artificial Neural Networks (ANN). The results indicate that the SVM and ANN methods have a success rate greater than 90% in the estimated parameters.

Keywords: Internet of Things; Industry 4.0; energy consumption; LoRaWAN; Machine Learning

1. Introducción

Cuatro revoluciones industriales han impactado la economía históricamente: la primera, cuyo impacto fue la invención de la máquina de vapor; la segunda, cuyo impacto fue el uso de la energía eléctrica en un sinnúmero de procesos productivos; y la tercera, cuyo impacto fue la digitalización de datos dentro de la industria, a través de sistemas de automatización, control y supervisión (Lasi, Fettke, Kemper, Feld, & Hoffmann, 2014). La cuarta revolución industrial (o Industria 4.0) permite realizar un análisis inteligente de los datos, para transformar la forma en la cual se ejecutan diversos procesos (Schwab, 2017). Dicha revolución se apropia, entre otros, de técnicas de algoritmia que permiten hacer clasificación y/o regresión de fenómenos. El desarrollo de software se configura como una herramienta imprescindible para habilitar la cuarta revolución industrial (Petrasch & Hentschke, 2016), permitiendo, a través de técnicas de inteligencia artificial, realizar análisis sobre grandes volúmenes de información. El Internet de las Cosas (IoT), a su vez, potencia la cuarta revolución industrial como una fuente de datos sin precedentes para su posterior análisis.

Paradigmas como el *Big Data*, la robótica, el aprendizaje automático, y el Internet de las Cosas son potenciadores de la Industria 4.0 (Schwab, 2017). El *Big Data* se define como el tratamiento de grandes volúmenes de información que no pueden ser procesados por esquemas tradicionales de computación en un tiempo tolerable (Chen, Mao, & Liu, 2014). La robótica potencia la maquinaria y su inclusión en la transformación digital de las plantas productivas. La Inteligencia Artificial (IA) predice y clasifica comportamientos con intervención mínima del ser humano (Lee, Davari, Singh, & Pandhare, 2018). El IoT permite, a través de la dotación de objetos de la cotidianidad con sensores y actuadores, conocer el comportamiento de procesos e incidir sobre ellos (Wollschlaeger, Sauter, & Jasperneite, 2017). El análisis de las características de conectividad, en particular en el consumo energético en redes de Internet de las Cosas, son el interés principal en el presente artículo.

En sitios donde no se puede depender de un operador de red, las redes de baja potencia y área amplia (LPWAN) se ajustan para transferir pequeños paquetes de información. Las tecnologías líderes en el mercado actualmente son: LoRaWAN (Sornin, Luis, Eirich, Kramp, & Hersent, 2015), Sigfox (Lavric, Petrariu, & Popa, 2019) y NB-IoT (Ratasuk, Vejlgard, Mangalvedhe, & Ghosh, 2016). El consumo energético es uno de los factores más críticos en el diseño de una red de IoT, ya que está pensada, en una gran cantidad de casos, para operar con baterías y durante tiempos prolongados. El presente artículo muestra una mejora del algoritmo *ADR* (*Adaptive Data Rate*) (San Cheong, Bergs, Hawinkel, & Famaey, 2017) del protocolo LoRaWAN, que se logra por medio del uso de Máquinas de Vectores de Soporte (SVM) y Redes Neuronales Artificiales (ANN). Tal modelación se logra a través de simulaciones exhaustivas en Omnet ++.

El presente trabajo se organiza como sigue: en la sección 2 se condensa el marco teórico requerido para la lectura expedita del trabajo, enfocado al protocolo LoRaWAN; en la sección 3 se muestra el trabajo previo que se ha adelantado para mejorar el consumo energético en redes LoRaWAN; en la sección 4 se muestra el diseño metodológico y experimental; en la sección 5 se muestra la modelación propuesta; en la sección 6 se muestra la propuesta del algoritmo para el mejoramiento del esquema ADR y su evaluación; finalmente, se presentan el trabajo futuro, las conclusiones y las referencias.

2. Marco teórico

Con respecto al componente arquitectónico, las redes LoRaWAN operan usando los siguientes componentes (de Carvalho Silva, Rodrigues, Alberti, Solic, & Aquino, 2017): (i) nodos, que incluyen sensores, sistemas embebidos y radios para transportar los datos recolectados; (ii) *gateways*, que se encargan de recibir información de los nodos y reenviarla a un servidor de red; (iii) servidor de red, el cual provee un servicio para recolectar los mensajes de diversos *gateways* y reenviarlos a demanda a algún servidor de aplicación, y de implementar esquemas como el ADR; y (iv) servidor de aplicación, donde se realizan las labores de despliegue y análisis de datos.

En cuanto a la utilización del espectro, el protocolo LoRaWAN opera en la banda ISM (*Industrial – Scientific – Medical*) que no requiere licencia¹. En América, la banda usada es 902-928MHz, y en Europa, 862 a 870 MHz (Sornin et al., 2015). Cada región define los canales permitidos para subida y bajada. En cuanto al uso apropiado del espectro, entidades como la FCC (*Federal Communications Commission*) y la ETSI (*European Telecommunications Standards Institute*) se encargan de dar directrices para permitir su uso colectivo.

LoRaWAN usa la modulación *Chirp Spread Spectrum* (CSS) (Reynders & Pollin, 2016). Los canales pueden tener anchos de banda (BW) de 125 kHz, 250 kHz y 500 kHz, dependiendo del plan de frecuencia. Si la frecuencia central es f_o , una señal conocida como *chirp* se mueve en el intervalo $(f_o - BW/2, f_o + BW/2)$, durante un tiempo de símbolo T_s . El número de bits por símbolo se conoce como *Spreading Factor* (SF) y puede variar entre 7 y 12. El espectro se divide en 2^{SF} posibles frecuencias de inicio (*chips*), de tal manera, en un mismo símbolo puede incluir 2^{SF} diferentes números. Para agregar bits de redundancia, es posible usar el *Coding Rate* (CR) que puede tomar valores entre 1 y 4. La rata de *chips* (R_c) es homóloga al ancho de banda y se define en la ecuación (1). La rata de símbolos se define en función del ancho de banda y el SF, como se indica en la ecuación (2), así como el tiempo de símbolo T_s . La rata de bits, *data rate*, se define en la ecuación (3).

$$BW \text{ (Hz)} = R_c \text{ (chips/sec)} \quad (1)$$

$$R_s \text{ (símbolos/sec)} = \frac{BW}{2^{SF}} ; T_s \text{ (sec)} = \frac{2^{SF}}{BW} \quad (2)$$

¹ <https://www.thethingsnetwork.org/docs/lorawan/frequencies-by-country.html>

$$R_b(\text{bits/sec}) = SF \frac{BW}{2^{SF}} \frac{4}{4 + CR} \quad (3)$$

De las ecuaciones (2) y (3) se puede deducir que para valores mayores de SF, el tiempo de símbolo también será mayor, permitiendo mayores distancias y mayor consumo. Con respecto a los parámetros de diseño, se definen el *duty cycle* (dc) y el *Time on Air* (ToA). Los *dc* son períodos en los cuales un nodo no transmite ninguna información, después de haber hecho una transmisión (Adelantado et al., 2017). El *dc* puede tomar valores de 0.1%, 1% o 10%, e indica la proporción de tiempo usada para realizar la transmisión. El *ToA* indica el tiempo que tarda un paquete de información en ser enviado. Así, el tiempo que debe esperarse para realizar una nueva transmisión desde un nodo se define en función del *dc* y el *ToA* como se muestra en la ecuación (4).

$$T_{nt} = \frac{(100 - dc)ToA}{dc} \quad (4)$$

Donde T_{nt} indica el tiempo para efectuar una nueva transmisión. A su vez el *ToA* se define en la ecuación (5).

$$ToA = (4.25 + n_s)T_s \left(8 + \max \left(\text{ceil} \left(\frac{(8PL - 4SF + 28 + 16CRC - 20H)}{4(SF - 2DE)} \right) (CR + 4), 0 \right) \right) \quad (5)$$

Donde n_s es el número de preámbulos (típicamente 8), *PL* es el tamaño en bytes del *payload*, *CRC* es una variable binaria que indica si se incluye comprobación por redundancia cíclica, *H* indica si el tipo de trama es implícito o explícito y *DE* si se activa la función *low data rate optimization*, el cual se activa para valores de *ToA* mayores a 100 milisegundos (obligatorio para valores de *SF* de 11 y 12). Para determinar los valores óptimos de *SF* y la potencia de transmisión *TP* de un nodo es posible habilitar el algoritmo *ADR*, ejecutado en el servidor de red (Sornin et al., 2015). Dicho servidor se encarga de recolectar los últimos veinte valores de relación de señal a ruido (*SNR*) y los *data rate* usados (por ejemplo, $SNR = 5$ dB, $DR = SF12BW125$), y procede a tomar el valor máximo. El servidor de red aplica la ecuación objetivo descrita en (6).

$$\text{Min } \text{margin}_{corrected} = SNR_{measured} - SNR_{limit} - \text{linkbudget}_{default} \quad (6)$$

Donde $\text{linkbudget}_{default}$ indica el nivel de seguridad, por encima de la sensibilidad del radio, SNR_{limit} indica la relación de señal a ruido límite que varía con el *SF* (a mayor *SF*, mejor sensibilidad de recepción del radio), $SNR_{measured}$ es el valor máximo de la *SNR* medida y $\text{margin}_{corrected}$ es la variable objetivo de optimización, cuyo valor debe ser mayor a cero. Nótese que, si las distancias son grandes, puede requerirse tener valores de *SF* y *TP* altos. No obstante, después de realizar las mediciones, el valor de $\text{margin}_{corrected}$ puede estar lejos de cero, indicando un desperdicio en el *TP* y un *ToA* elevado. Así pues, el servidor de red ajusta el *SF* y recalcula el margen. Si incluso el $\text{margin}_{corrected}$ no es cero, se ajusta *TP* para bajar la potencia de transmisión del radio. Las simulaciones implementadas muestran que el tiempo de convergencia de este algoritmo es susceptible de mejora (Li, Raza, & Khan, 2018).

3. Trabajo previo

Los trabajos de la literatura relacionados más representativos se muestran en la Tabla 1. Las aproximaciones de manejo energético en la literatura abordan diferentes perspectivas, la mayoría proponiendo valores para SF y TP , pero optimizando otras métricas diferentes a la energía, tales como la capacidad, la confiabilidad y el rendimiento en general y no se enfocan a la minimización de la energía. También se prevén métodos usando la recolección de algunos parámetros como la SNR y la $RSSI$ durante un tiempo suficiente que permita estabilizarlos en un ambiente real, por lo cual el tiempo de convergencia puede ser incierto. Finalmente, diferentes propuestas plantean problemas de optimización que consideran la energía como una restricción, esto es, se limitan a fijar un valor máximo, no a encontrar su valor mínimo. En virtud de los hallazgos, se observa la oportunidad de proponer un método que mejore el consumo energético a partir de la aplicación de modelos que lo optimicen.

Autores	Título	Contribución	Oportunidades de mejora
(Slabicki, Preamsankar, & Di Francesco, 2018)	<i>Adaptive Configuration of LoRa Networks for Dense IoT Deployments</i>	Mejora el consumo energético en canales altamente ruidosos, basándose, entre otros, en el conocimiento previo de toda la red (<i>network-aware approach</i>).	Modifica el máximo de las muestras SNR recolectadas en el algoritmo ADR por el promedio. Para que esta aproximación opere adecuadamente en un entorno real se requeriría alta concentración de los SNR alrededor del promedio.
(Sandoval, Rodenas-Herraiz, Garcia-Sanchez, & Garcia-Haro, 2020)	<i>Deriving and updating optimal transmission configurations for LoRa Networks</i>	Propone la sintonización de los parámetros de la transmisión sin requerir un presupuesto de enlace por nodo ni modelos de canal de manera previa. También clama mejorar la resistencia a ataques de denegación de servicio.	Utiliza, para proponer el esquema el modelo de consumo energético, un microcontrolador LoPy4, por lo cual puede carecer de generalidad cuando otras plataformas sean usadas. Además, el número de mediciones realizadas (12), a diferentes potencias, luce insuficiente para otorgar validez estadística a los experimentos
(Reynders, Meert, & Pollin, 2017)	<i>Power and spreading factor control in low power wide area networks</i>	Propone la disminución del efecto de colisiones destructivas entre nodos cerca y lejos del <i>gateway</i> , por medio de la variación del SF y el TP , mejorando la tasa de error de paquetes.	El resultado de la disminución (o aumento) de consumo energético es una consecuencia de la minimización de las colisiones. El algoritmo, si bien entrega valores para SF y TP , no está pensado para minimizar el consumo energético.
(Abdelfadeel, Cionca, & Pesch, 2018)	<i>Fair Adaptive Data Rate Allocation and Power Control in LoRaWAN</i>	Propone un método que permite resolver el problema de interferencia entre nodos cercanos y nodos lejanos, a través del control del SF y el TP , por medio del algoritmo FADR	Basa el cálculo de la potencia de transmisión y el SF en el promedio de muestras recolectadas de los $RSSI$, lo cual implicaría un tiempo de convergencia incierto (no se mide). No se aporta un método para calcular los TP iniciales, lo cual puede redundar en problemas iniciales de conexión entre los nodos y el GW .

Autores	Título	Contribución	Oportunidades de mejora
(Gao, Du, Zhao, Min, & Singhal, 2019)	<i>Towards energy-fairness in LoRa networks</i>	Modela el comportamiento de energía, confiabilidad en la transmisión e interferencia como un problema de optimización. Propone un algoritmo para realizar la asignación óptima de recursos en un ambiente de múltiples gateways.	Asume que la asignación de los SF y TP se puede optimizar usando más de un <i>gateway</i> , lo cual es cierto desde el punto de vista teórico. Sin embargo, en escenarios reales, los <i>gateway</i> , que tienen alimentación por red eléctrica, no pueden ser ubicados de manera arbitraria, por las condiciones topográficas de una región en particular.
(Bouguera, Diouris, Chaillout, Jaouadi, & Andrieux, 2018)	<i>Energy consumption model for sensor nodes based on LoRa and LoRaWAN</i>	Propone un modelo de cuantificación de energía para nodos en LoRa, que permite evaluar el comportamiento de potencia en diferentes configuraciones, cuantificándolas en varios escenarios.	No proporciona un algoritmo para mejorar la eficiencia en consumo energético de los nodos finales en una red LoRa. Esta actividad se propone como trabajo futuro.
(Sandoval, Garcia-Sanchez, & Garcia-Haro, 2019)	<i>Performance optimization of LoRa nodes for the future smart city/industry</i>	Presenta un modelo matemático, basado en cadenas de Markov, que permite estimar numéricamente el rendimiento de los nodos en una red LoRaWAN, por medio de <i>policy-derivation theory</i> .	El problema de optimización formula como función objetivo la maximización del rendimiento de la red, y la energía consumida se incluye como una restricción. De tal manera, la energía no se minimiza, sino que se limita a un valor máximo dentro de una región factible determinada.

Tabla 1 – Resumen de trabajos previos

4. Metodología y diseño experimental

Primero, se propone la simulación de una red LoRaWAN con 1000 nodos, 1 *gateway* y 1 servidor de red. Se escogen 1000 nodos para garantizar significancia en los datos de la simulación. A su vez, se usa 1 *gateway* y un servidor de red dado que ésta es una configuración típica en este tipo de redes. Los valores de *TP* (2-14 dBm) y *SF* (7-12) se inicializan de manera aleatoria, así como la posición de los nodos, en una geometría de 1 km² de área. El *gateway* se ubica en el centro de la región. Se considera que cada nodo transmitirá un mensaje de manera aleatoria (distribución exponencial) en un intervalo de 0 a 100 segundos. El tiempo de simulación se fija a un día, ya que se encuentra un balance adecuado entre el tamaño de la base de datos generada y la convergencia del algoritmo ADR. El ancho de banda se fija a 125 kHz, el más común para canales de subida en los diferentes planes de frecuencia por región. Con respecto a la herramienta de simulación, se usa Omnet ++ V5.2.1 y el complemento *FLoRa* (Slabicki et al., 2018). Se habilita el algoritmo tradicional ADR para escogencia de los parámetros *SF* y *TP*.

Como segundo paso, para la modelación de la base de datos, se tienen en cuenta los argumentos de entrada distancia (D), $TP_{inicial}$, $SF_{inicial}$. Se toma la razón entre la

energía consumida por un nodo y la cantidad de paquetes enviados como salida del modelo (EPP), así como el TP_{final} , el SF_{final} y el tiempo T_c que le toma al algoritmo *ADR* converger a dichos valores. Se construye la base de datos a través de las corridas de simulación respectivas.

En el tercer paso, con los valores obtenidos de energía/paquete y TP para los diferentes casos de D, $TP_{inicial}$ y $SF_{inicial}$, se evalúa la modelación a través de regresión lineal múltiple, modelos lineales generalizados (GLM) y SVM.

En cuarto lugar, se evalúan modelos de aprendizaje de máquina supervisados (regresión logística, SVM y ANN) con los mismos argumentos de entrada.

En quinto lugar, se observa el rendimiento de los métodos empleados y se escoge el que exhibe mejores resultados. La aproximación de evaluación divide la base de datos, de manera aleatoria, en 80% de muestras para entrenamiento y 20% para validación. Finalmente, se valida la mejora de energía contrastando el tiempo T_c para hallar los parámetros TP y SF a través del algoritmo *ADR* y por medio de la aplicación de los modelos propuestos.

5. Modelación

Para el análisis de datos se usa el software R Studio v1.2. Se muestran los análisis para las tres métricas de salida: energía/paquete (EPP), TP_{final} y SF_{final} . El resumen estadístico de las variables continuas y categóricas se muestra en la Tabla 2.

Variable continua	Mínimo	Cuartil 1	Mediana	Media	Cuartil 3	Máximo
EPP (J)	0.08062	0.12261	0.15706	0.15914	0.19028	0.28664
D (m)	10.63	137.64	194.19	186.2	234.95	336.7
TP_{final} (dBm)	2	5	8	8.24	14	14
$TP_{inicial}$ (dBm)	2	5	8	7.94	11	14
Variable categórica	SF = 7	SF = 8	SF = 9	SF = 10	SF = 11	SF = 12
SF_{final}	37	45	47	44	26	801
$SF_{inicial}$	171	152	156	175	175	171

Tabla 2 – Resumen estadístico de variables continuas y categóricas

5.1. Energía por paquete transmitido (EPP)

Regresión lineal múltiple

Se evalúa la regresión lineal múltiple siguiendo el modelo en la ecuación (7).

$$EPP = \hat{\beta}_0 + \hat{\beta}_1 D + \hat{\beta}_2 TP_{inicial} + \sum_{j=7}^{12} \hat{\beta}_{3j} SF_{inicial_j} + \epsilon \quad (7)$$

Donde es $\hat{\beta}_0$ es el intercepto, $\hat{\beta}_1$ es el estimador para la distancia D, $\hat{\beta}_2$ es el estimador para TP_{inicial}, $\hat{\beta}_{3,j}$ ($j=7,...,12$) son los estimadores para las variables binarias $SF_{inicial,j}$, y ε es el error del modelo. El método de estimación de los coeficientes usado es LSE (*Least Square Estimator*) con un valor de significación de 5%. Los estimadores se muestran en la Tabla 3.

Coef.	Estimado	p-valor	Coef.	Estimado	p-valor
$\hat{\beta}_0$	6.413×10^{-2}	$< 2 \times 10^{-16}$	$\hat{\beta}_{3,9}$	2.084×10^{-2}	$< 2 \times 10^{-16}$
$\hat{\beta}_1$	1.504×10^{-4}	$< 2 \times 10^{-16}$	$\hat{\beta}_{3,10}$	5.611×10^{-2}	$< 2 \times 10^{-16}$
$\hat{\beta}_2$	2.359×10^{-3}	$< 2 \times 10^{-16}$	$\hat{\beta}_{3,11}$	8.466×10^{-2}	$< 2 \times 10^{-16}$
$\hat{\beta}_{3,8}$	1.056×10^{-2}	0.000177	$\hat{\beta}_{3,12}$	1.054×10^{-1}	$< 2 \times 10^{-16}$

Tabla 3 – Predictores estimados en el modelo de regresión lineal múltiple

Los p-valores indican que todos los predictores son significativos. El modelo entrega un R^2 múltiple de 0.7686 y un R^2 ajustado de 0.7667, que indica una correlación lineal fuerte entre las variables predictoras y la variable dependiente EPP. En la Figura 1 se muestra el comportamiento de algunos de los puntos reales vs los predichos por el modelo. Se omite el conjunto completo para garantizar la legibilidad en la lectura. En orden de asegurar la validez estadística del modelo, se implementa el análisis de varianza, encontrando p-valores del orden de 2×10^{-16} , mostrando que, en efecto, todas las variables son significativas. Adicionalmente, deben garantizarse los supuestos de normalidad, independencia y homocedasticidad de los residuales. En el primer caso se aplica la prueba de Kolmogorov Smirnov fijando un nivel de significación de 5%, y entregando un valor p de 5.793%. En el caso de independencia, se aplica la prueba de Durbin Watson, entregando un $DW=1.9907$, indicando independencia entre las variables predictoras. Finalmente, para garantizar homocedasticidad, se aplica la prueba de Breusch Pagan, obteniendo un valor $p < 2 \times 10^{-16}$, indicando que los residuales son heterocedásticos. En tal virtud, el modelo pierde eficiencia en sus estimadores de mínimos cuadrados, aunque sigue siendo insesgado ($E(X)=X$).

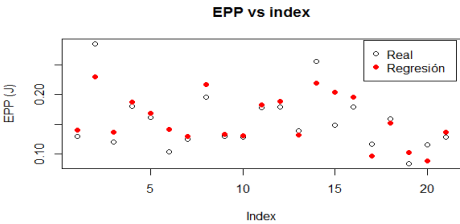


Figura 1 – Comparación entre EPP real y estimado usando el modelo RLM

Modelos lineales generalizados

Dado que se tiene heterocedasticidad, se recomienda utilizar un modelo lineal generalizado con diferentes funciones de enlace: Gamma y gaussiana inversa.

Las funciones de enlace binomial y Poisson no se consideran por el tipo de proceso que se está modelando. Los coeficientes de los modelos se estimarán utilizando el criterio de máxima verosimilitud (MLE). La calidad de los modelos se evaluará utilizando el índice AIC y la calidad de las predicciones usando el R^2 . Con la función Gamma se logra un AIC de -3918, un R^2 de 0.71 y un RMS de 0.69; y con la gaussiana inversa se logra un AIC de 3828, un R^2 de 0.74 y un RMS de 6.7. Si bien se observa una relación a través del R^2 , usando ambos modelos, también se observa que los errores cuadráticos medios están en órdenes de magnitud mucho mayores que los de los datos. Esta lectura indica que ambos modelos fallan en las regresiones. Si se grafica el EPP del modelo vs EPP teórico (ambos del subconjunto de pruebas), se esperaría una tendencia lineal positiva; no obstante, se obtiene lo graficado en la Figura 2. Por tal motivo, no se publican los valores del modelo, ya que no hay un ajuste satisfactorio.

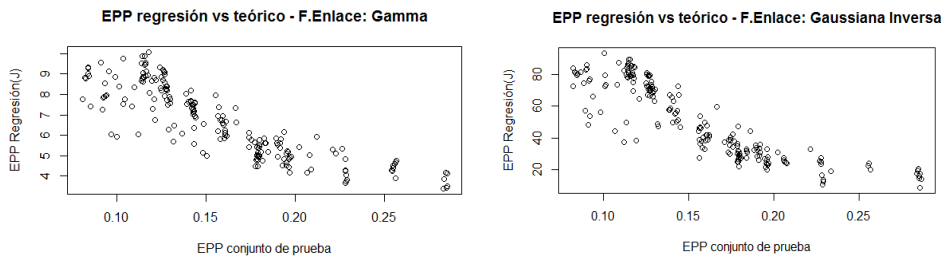


Figura 2 – EPP calculado vs. teórico usando funciones de enlace Gamma y gaussiana inversa

Máquinas de vectores de soporte (SVM)

Se procede a aplicar regresión a través de SVM (Noble, 2006). En el caso de clasificación con SVM, se busca el hiperplano que separe dos clases con el máximo margen entre ambas, por lo cual, encontrar el modelo se reduce a un problema de optimización, donde deben hallarse los coeficientes w que definen dicho hiperplano, como se muestra en la ecuación (8).

$$\text{Min } Z = \frac{1}{2} \|w\|^2 \quad (8)$$

$$\text{Sujeto a: } |y_i - w_i x_i| \leq \epsilon,$$

donde y_i corresponde a las características objetivo, w_i a los pesos y x_i a las características de entradas. En el caso de regresión, se agrega a la ecuación (8) un conjunto de variables de holgura, y así, para cada valor que cae por fuera de ϵ , se registra su desviación ξ , de tal manera que se desea que ésta sea mínima. Se introduce el hiper parámetro costo (C) para penalizar dichas desviaciones, el cual es sintonizado a través de validación cruzada. De tal manera, la ecuación (8) se reescribe como se muestra en la ecuación (9).

$$\text{Min } Z = \frac{1}{2} \|w\|^2 + c \sum_{i=0}^n |\xi_i| \quad (9)$$

$$\text{Sujeto a: } |y_i - w_i x_i| \leq \epsilon + |\xi_i|$$

Para el caso de EPP, se toman como características de entrada las mismas usadas en los métodos paramétricos: D , TP_{inicial} y SF_{inicial} y las características objetivo son los valores de EPP. Para realizar las evaluaciones, se usan cuatro tipos diferentes de *kernels*: lineal (RMS=0.025, $R^2=0.75$), polinomial (no converge), sigmoideal (RMS=0.029, $R^2=0.714$) y de base radial (RMS=0.02, $R^2=0.837$). De acuerdo con las métricas presentadas, y a una inspección visual (mostrada en la Figura 3a), el *kernel* de base radial (RBF) es el que se ajusta mejor a los datos. El RBF se considera una medida de similitud modificada (calculada a partir de la distancia euclidiana), y extrapola el espacio de características a infinitas dimensiones. El RBF entre dos muestras se calcula como: $K(x, x') = \exp(-\gamma \|x - x'\|^2)$. El hiper parámetro γ se debe sintonizar a través de validación cruzada. Para escoger los hiper parámetros C y γ se usa el software *R Studio*, librería *e1071*, método *tune*. La Figura 3b muestra la comparación de los errores de clasificación usando diferentes configuraciones de C y γ , de donde se deduce que, para diferentes conjuntos de entrenamiento, los mejores valores son $C=10$ y $\gamma=10$. Se omite la escritura de los vectores de soporte encontrados, dada su cantidad: 439.

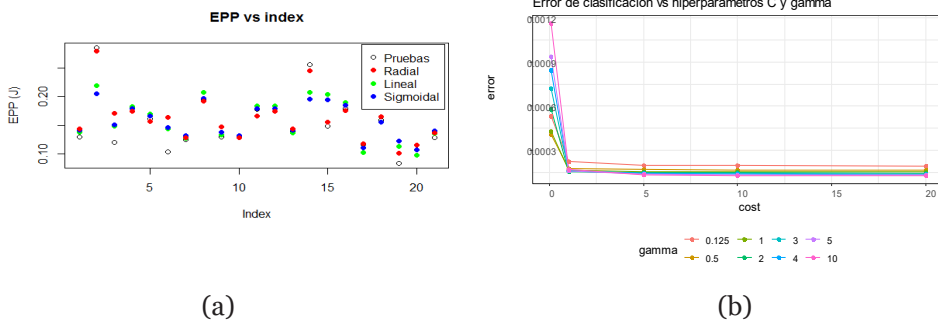


Figura 3 – (a) EPP del conjunto de pruebas vs. regresiones de *kernels* radial, lineal y sigmoideal
(b) validación cruzada de parámetros gamma y costo.

5.2. Potencia de transmisión final (TP_{final})

Los análisis de RLM, GLM y SVM realizados en este aparte se analizan bajo la misma óptica de la sección 5.1., por lo cual se omitirán los detalles y se procederá a ilustrar los resultados. En todos los casos TP_{final} será una función de D , TP_{inicial} y SF_{inicial} .

Regresión lineal múltiple

Al implementar el RLM, se encuentra que el predictor $\hat{\beta}_1$ correspondiente a la distancia no es significativo para el modelo. Tampoco lo es la variable *dummy* SF_{final} . El análisis de varianza arroja un p-valor 0.2173 para la distancia, lo cual corrobora la no significancia de dicha variable predictora. El modelo ajustado presenta un R^2 múltiple de 0.9144 y un R^2 ajustado de 0.9136. La distribución de los residuales no es normal (p-valor < 2.2 x

10^{-16} en la prueba de Kolmogorov Smirnov), no hay autocorrelación (prueba de Durbin Watson, $DW=2.0856$, $p\text{-valor}=0.2246$) y no hay homocedasticidad (prueba de Breusch Pagan, $p\text{-valor} = 1.426 \times 10^{-15}$). La violación del supuesto de normalidad en los residuales invalida la aplicación del modelo, por lo cual no se publican los coeficientes.

Modelos lineales generalizados

Al aplicar la función de enlace gamma se obtiene un AIC de 3755, un R^2 de 0.91 y un RMS de 9.24. Al aplicar la gaussiana inversa se obtiene un AIC de 4303, un R^2 de 0.91 y un RMS de 9.48. El comportamiento exhibido es similar al observado en la Figura 2. Dado que la tendencia debería ser lineal y no se obtiene este comportamiento, se concluye que GLM no ajusta adecuadamente los datos, por lo cual no se publican los resultados de los modelos.

Máquinas de vectores de soporte (SVM)

Se realizan las evaluaciones con cuatro tipos diferentes de *kernels*: lineal (RMS=1.2686, $R^2=0.91307$), polinomial (no converge), sigmoidal (RMS=1.27548, $R^2=0.91159$) y RBF (RMS=0.9196, $R^2=0.955$). Dado que RBF es el que logra los mejores resultados con el conjunto de pruebas (Figura 4a), se procede a sintonizar los hiper parámetros C y γ , cuyos resultados se muestran en la Figura 4b. Los mejores valores son $C=20$ y $\gamma=0.125$. Se omite la publicación de los vectores de soporte, debido a su número: 244.

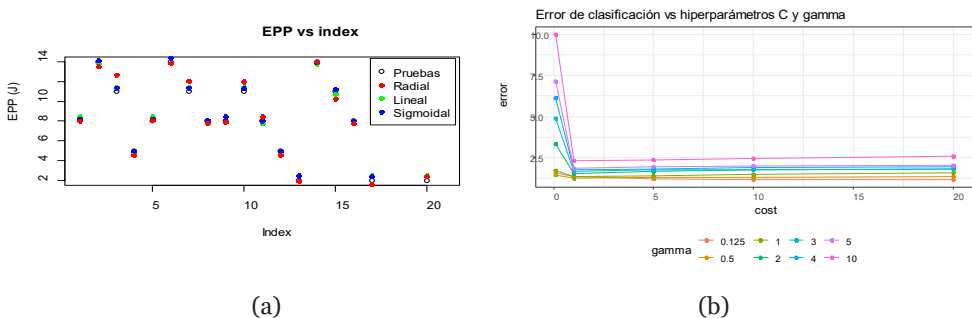


Figura 4 – (a) EPP del conjunto de pruebas vs. regresiones de *kernels* radial, lineal y sigmoidal, (b) Sintonización de hiper parámetros del SVM de base radial

5.3. Número de bits por símbolo (*Spreading Factor*) final (SF_{final})

Dado que el parámetro SF_{final} es de tipo categórico, el análisis paramétrico se realizará a través de regresión logística múltiple y el no paramétrico con la estrategia de clasificación de SVM y usando ANN. En todos los casos TP_{final} será una función de D , $TP_{inicial}$ y $SF_{inicial}$.

Regresión logística multinomial

Sea $Y=\{1,2, 3,...,n\}$ una variable politómica cuyas probabilidades de ocurrencia son $p_1, p_2,...,p_n$ correspondientemente, las cuales se desean estimar a partir de un conjunto de características de entrada $x_1,..., x_m$. Se define cada p_j como se muestra en la ecuación (10).

$$p_j = E(Y_j) = \frac{e^{\sum_{i=0}^m \beta_{ij} x_i}}{1 + \sum_{l=1}^n e^{\sum_{i=0}^m \beta_{il} x_i}} \quad (10)$$

Donde los β_{ij} son los regresores que desean ser estimados, a través del criterio de máxima verosimilitud. Para el caso particular de la estimación del SF_{final} , la probabilidad de que una nueva observación pertenezca a p_j ($j = 7, 8, \dots, 12$) se puede reescribir a partir de (9), como se muestra en la ecuación (11).

$$p_j = E(Y_j) = \frac{e^{\beta_{j0} + \beta_{j1} D + \beta_{j2} TP_{inicial} + \sum_{k=8}^{12} \beta_{j3,k} SF_{inicial_k}}}{1 + \sum_{l=1}^n e^{\beta_{l0} + \beta_{l1} D + \beta_{l2} TP_{inicial} + \sum_{k=8}^{12} \beta_{l3,k} SF_{inicial_k}}} \quad (11)$$

De tal manera, se tendrán 8 regresores por cada clase, y debido a que se tienen valores de $SF_{final} = 7, 8, \dots, 12$, se tendrán 5 probabilidades ($SF_{final} = 8, \dots, 12$), debido a que $SF = 7$ se considera como una variable de referencia. Así, el número de parámetros a ajustar será de 40, y se muestran en la Tabla 4. Se usa el software *R Studio*, librería *nnet*, método *multinom*.

SF_{final}	Intercepto	D	TP _{inicial}	$SF_{inicial8}$	$SF_{inicial9}$	$SF_{inicial10}$	$SF_{inicial11}$	$SF_{inicial12}$
8	-7.918	0.270	-1.727	8.573	18.424	10.334	12.803	20.624
9	-9.036	0.339	-2.429	8.382	22.165	13.190	14.900	26.045
10	-11.290	0.372	-2.736	8.954	22.117	15.987	8.779	19.409
11	-12.799	0.380	-2.729	6.750	21.418	15.141	9.141	19.869
12	-11.873	0.402	-3.195	9.902	24.128	19.913	23.458	35.370

Tabla 4 – Parámetros estimados para el modelo de regresión logística multinomial

La evaluación se implementa generando una matriz de confusión (librería *caret*, método *confusionMatrix*), la cual contrasta los valores estimados vs. los valores esperados y realiza el conteo de aciertos y fracasos, y se muestra en la Tabla 5a (conjunto de entrenamiento) y Tabla 5b (conjunto de pruebas). Puede observarse que, si bien hay casos en los cuales hay éxito en las predicciones, i.e. $SF_{final}=7, 8$ y 12, hay otros casos en los cuales no se hacen predicciones correctas, i.e. $SF_{final}=10, 11$. La exactitud del modelo es 88.3%, y el intervalo de confianza (IC) [85.81%, 90.4%].

Máquinas de vectores de soporte (SVM)

Se realizan las evaluaciones con cuatro tipos diferentes de *kernels* (que se evalúan por medio de una matriz de confusión): lineal (exactitud: 86% IC 5%: (80.41%, 90.49%)), polinomial (no converge), sigmoidal (exactitud: 86.5% IC 5%: (83.93%, 88.79%)), y de base radial (exactitud: 85% IC 5%: (79.28%, 89.65%)). Dado que el *kernel* sigmoidal es el que entrega mejores resultados, se procede a sintonizar los hiper parámetros C y γ , cuyos valores son 5.1 y 0.125 respectivamente. No obstante, se presenta el mismo problema ilustrado en la regresión logística: no hay predicciones correctas para SF_{final} de 10 y 11, presentándose cero coincidencias, como se muestra en la Tabla 5. Se concluye que la SVM óptima hallada no es un clasificador válido para el problema en cuestión.

Pred. Ref (Entren.)	7	8	9	10	11	12
7	21	4	0	0	0	0
8	2	26	3	0	0	4
9	1	4	20	1	3	11
10	0	1	3	2	4	23
11	0	0	2	0	5	15
12	0	4	7	1	1	632

(a)

Pred. Ref (Entren.)	7	8	9	10	11	12
7	9	2	0	0	0	1
8	0	7	2	0	0	1
9	0	0	1	1	1	4
10	0	0	5	1	0	5
11	0	0	0	0	0	4
12	0	0	0	0	0	156

(b)

Tabla 5 – Matriz de confusión para valores del conjunto de entrenamiento/pruebas vs. valores reales

Redes neuronales artificiales (ANN)

Un método alternativo para lidiar con el problema exhibido por la SVM es usar ANN. Dicho método exhibe algunas desventajas con respecto a las SVM, relacionado con el *overfitting* (sobre entrenamiento), el cual puede redundar en regresiones o clasificaciones erróneas. A su vez, el riesgo se calcula a partir de la minimización de una función de pérdida (normalmente gradiente descendente) de los datos de entrenamiento (conocido como riesgo empírico), lo cual puede redundar en generalización errónea, a diferencia de las SVM. Por tal motivo, es muy importante el entrenamiento exhaustivo hasta encontrar el modelo que reduzca el *overfitting* y generalice adecuadamente, lo cual requiere muchos recursos de hardware en conjuntos de datos de gran tamaño. Se presenta una ANN para solucionar el problema de predicción errónea de los SF finales. Se busca el mejor modelo a partir del paquete CARET de R, método *train*, configurando una grilla que varía los parámetros intrínsecos de la red (número de capas, número de neuronas por capa, *decay*, entre otros) y calcula un nuevo modelo por cada configuración de parámetros. La métrica de desempeño es el acierto (*accuracy*). Para asegurar el mejor modelo, se configuran 100 épocas. Se encuentra una ANN perceptrón con 7 neuronas de entrada (D, SF_{inicial} *i*-ésimo (*i*= 8,...,12) y TP_{inicial}), 8 neuronas en una única capa oculta, y 6 neuronas de salida (SF_{final} *i*-ésimo (*i*=7,...,12)). La evaluación de la clasificación se hace usando una matriz de confusión, como se muestra en la Tabla 6. Puede observarse que los errores de clasificación mejoran con respecto a los dos métodos anteriores, exhibiendo una exactitud de 95% en el conjunto de entrenamiento y en el de prueba de 96%.

6. Método propuesto y evaluación

Una vez inferidos los algoritmos con mejor desempeño para inferir los parámetros TP_{final} y SF_{final}, se propone el Algoritmo 1. Dicho algoritmo se piensa para ser ejecutado en el servidor de red, de tal manera, los nodos enviarán la información necesaria al mismo: TP_{inicial}, SF_{inicial} y la distancia D desde el nodo hasta el *gateway* más cercano. Por seguridad, se recomienda inicializar los parámetros TP y SF a sus valores máximos,

y así se garantizará la entrega del primer paquete de información al servidor de red. Una vez el servidor recibe los parámetros, procede a calcular los valores mejorados, a través de SVM (en el caso de TP) y ANN (en el caso de SF), como se discutió en las secciones previas.

Pred. Ref (Entren.)	7	8	9	10	11	12
7	20	4	0	0	0	1
8	2	29	2	0	0	2
9	0	3	36	0	0	1
10	0	0	1	27	0	5
11	0	0	1	2	15	4
12	2	2	1	0	0	640

(a)

Pred. Ref (Entren.)	7	8	9	10	11	12
7	11	0	0	0	0	1
8	2	6	1	0	0	1
9	0	0	7	0	0	0
10	0	0	1	9	0	1
11	0	0	0	0	4	0
12	0	0	0	0	0	156

(b)

Tabla 6 – Matriz de confusión para valores del conjunto de entrenamiento/prueba

Algoritmo 1: Determinación de TP y SF usando SVM y ANN (en el servidor de red)	
1:	entrada: distancia D, $TP_{inicial}$, $SF_{inicial}$
2:	parámetros: SV, C, γ , w, b
3:	para cada nodo hacer ➤ Inferir TP_{final}
4:	recibir distancia D del nodo
5:	$C \leftarrow 10$
6:	$\gamma \leftarrow 0.125$
7:	fijar SVR a SVM de base radial
8:	$TP_{final} \leftarrow SVR(D, TP_{inicial}, SF_{inicial}, C, \gamma, SV)$
9:	fin para
10:	para cada nodo hacer ➤ Inferir SF_{final}
11:	inicializar ANN Perceptrón multicapa: Capa entradas (7 neuronas), Una capa oculta (8 neuronas), Capa salida (6 neuronas), Pesos w, Bias b.
12:	$SF_{final} \leftarrow \max(SF_{final} \text{ i-ésimo}) \leftarrow ANN(D, TP_{inicial}, SF_{inicial}, w, b)$
13:	fin para
14:	salida: TP_{final} , SF_{final}

Para comparar la efectividad de los métodos propuestos para la estimación de los parámetros TP y SF, se analiza el tiempo de convergencia del algoritmo tradicional ADR vs. el tiempo que tarda el método propuesto en este trabajo. Al realizar la evaluación con respecto al parámetro TP, se encontró a través de las simulaciones que, de los 1000 nodos, 120 requirieron el ajuste del TP a través de ADR. El mínimo número de veces que el nodo debió solicitar ajustar el TP fue 20, y el máximo fue 485. Teniendo en cuenta el TP inicial, es posible hallar la potencia consumida en la transmisión en *watts*, si se tiene en cuenta que $W = 10^{TP/10}/1000$. Al realizar los cálculos, se encontraron potencias mínimas iniciales por paquete de 1.5×10^{-3} w y máximas iniciales de 2.5×10^{-2} w, las cuales fueron inicializadas de manera aleatoria.

En tal virtud, es posible calcular la energía consumida en exceso por las retransmisiones, si se conoce el ToA (ecuación 5). Utilizando los parámetros de la simulación ($BW = 125$ kHz, $\min SF = 7$, $\max SF = 12$, $Coding\ rate = 1$, longitud del *payload* = 10 bytes), es posible inferir que el mínimo ToA es igual a 0.041 segundos y el máximo a 0.991 s. Así, la energía consumida en exceso (para $SF = 7$) puede oscilar entre 0.002 J y 0.318 J y la energía consumida en exceso (para $SF = 12$) puede oscilar entre 0.062 J y 7.67 J. Ahora bien, dado que el método que se propone en este trabajo sólo requiere enviar de manera satisfactoria desde el nodo hasta el *gateway* los parámetros iniciales TP, SF y distancia, el costo energético para el nodo para ajustar el algoritmo de control de energía será equivalente a una única transmisión/ recepción, la cual, en función de los valores de retransmisión enunciados previamente, pueden estar en proporciones desde 1/485 hasta 1/20, es decir, el método logra reducir, porcentualmente, la energía consumida por el algoritmo ADR tradicional al 5%, en el peor de los casos, lo cual exhibe una mejora sustancial en consumo energético.

En cuanto al tiempo de convergencia, se encontró que el tiempo mínimo para hallar el valor óptimo fue de 1871 s, y el máximo fue de 86.314 s; no obstante, dicho tiempo está en función de las transmisiones que haga cada nodo, las cuales fueron aleatorizadas a través de una distribución exponencial.

7. Conclusiones y trabajo futuro

En el presente trabajo se presentó un método basado en técnicas no paramétricas de aprendizaje de máquina que permiten estimar la potencia consumida en un nodo de Internet de las Cosas que transmita información a través del protocolo LoRaWAN. Para recolectar los datos, se usó el simulador Omnet++ usando el *framework FLoRa*, en el cual se desplegaron 1000 nodos. La energía consumida en un nodo, como se mostró, depende de la distancia, la potencia de transmisión y el número de bits por símbolo, valores que son sintonizados a través del algoritmo ADR provisto por el protocolo LoRaWAN. Se mostró que la cantidad de paquetes transmitidos para sintonizar dichos valores oscila entre 20 y 485, exhibiendo una posible mejora para el tiempo de convergencia, y de esta manera, una reducción en el consumo energético. Del conjunto de métodos evaluados, se encontró que las máquinas de vectores de soporte (SVM) son efectivas para la regresión de energía por paquete (EPP) y para ajustar la potencia de transmisión (TP_{final}), en tanto las redes neuronales son efectivas para clasificar el valor del número de bits por símbolo (SF_{final}). Se propone un algoritmo para modificar el ADR tradicional,

el cual corre en el servidor y asegura la convergencia de los parámetros SF_{final} y TP_{final} en una única transmisión/recepción por nodo. Se planea, como trabajo futuro, realizar la implementación física en un ambiente de radio definido por software, para asegurar la validez de los modelos y métodos propuestos en un ambiente de operación real.

Agradecimientos

A la Universidad de Medellín por financiar el trabajo de investigación “Esquema algorítmico para la transmisión de información que minimice el consumo energético y mejore el uso del espectro radioeléctrico en dispositivos de internet de las cosas”, código de proyecto 1039, y por la beca de estudios doctorales otorgada a uno de los autores.

Referencias

- Abdelfadeel, K. Q., Cionca, V., & Pesch, D. (2018). *Fair adaptive data rate allocation and power control in lorawan*. Paper presented at the 2018 IEEE 19th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM).
- Adelantado, F., Vilajosana, X., Tuset-Peiro, P., Martinez, B., Melia-Segui, J., & Watteyne, T. (2017). Understanding the limits of LoRaWAN. *IEEE Communications magazine*, 55(9), 34-40.
- Bouguera, T., Diouris, J.-F., Chaillout, J.-J., Jaouadi, R., & Andrieux, G. (2018). Energy consumption model for sensor nodes based on LoRa and LoRaWAN. *Sensors*, 18(7), 2104.
- Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. *Mobile networks and applications*, 19(2), 171-209.
- de Carvalho-Silva, J., Rodrigues, J. J., Alberti, A. M., Solic, P., & Aquino, A. L. (2017). LoRaWAN—A low power WAN protocol for Internet of Things: A review and opportunities. In *Proceedings of the 2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*.
- Gao, W., Du, W., Zhao, Z., Min, G., & Singhal, M. (2019). Towards Energy-Fairness in LoRa Networks. In *Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*.
- Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & information systems engineering*, 6(4), 239-242.
- Lavric, A., Petrariu, A. I., & Popa, V. (2019). Long range sigfox communication protocol scalability analysis under large-scale, high-density conditions. *IEEE Access*, 7, 35816-35825.
- Lee, J., Davari, H., Singh, J., & Pandhare, V. (2018). Industrial Artificial Intelligence for industry 4.0-based manufacturing systems. *Manufacturing letters*, 18, 20-23.
- Li, S., Raza, U., & Khan, A. (2018). How Agile is the Adaptive Data Rate Mechanism of LoRaWAN? In *Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM)*.

- Noble, W. S. (2006). What is a support vector machine?. *Nature biotechnology*, 24(12), 1565-1567.
- Petrasch, R., & Hentschke, R. (2016). Process modeling for Industry 4.0 applications: Towards an Industry 4.0 process modeling language and method. In *Proceedings of the 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*.
- Ratasuk, R., Vejlggaard, B., Mangalvedhe, N., & Ghosh, A. (2016). NB-IoT system for M2M communication. In *Proceedings of the the 2016 IEEE wireless communications and networking conference*.
- Reynders, B., Meert, W., & Pollin, S. (2017). Power and spreading factor control in low power wide area networks. In *Proceedings of the 2017 IEEE International Conference on Communications (ICC)*.
- Reynders, B., & Pollin, S. (2016). *Chirp spread spectrum as a modulation technique for long range communication*. Paper presented at the 2016 Symposium on Communications and Vehicular Technologies (SCVT).
- San Cheong, P., Bergs, J., Hawinkel, C., & Famaey, J. (2017). *Comparison of LoRaWAN classes and their power consumption*. Paper presented at the 2017 IEEE symposium on communications and vehicular technology (SCVT).
- Sandoval, R. M., Garcia-Sanchez, A.-J., & Garcia-Haro, J. (2019). Performance optimization of LoRa nodes for the future smart city/industry. *EURASIP Journal on Wireless Communications and Networking*, 2019(1), 1-13.
- Sandoval, R. M., Rodenas-Herraiz, D., Garcia-Sanchez, A.-J., & Garcia-Haro, J. (2020). Deriving and Updating Optimal Transmission Configurations for Lora Networks. *IEEE Access*, 8, 38586-38595.
- Schwab, K. (2017). *The fourth industrial revolution*. Penguin Random House Grupo Editorial España, 2016.
- Slabicki, M., Premasankar, G., & Di Francesco, M. (2018). *Adaptive configuration of LoRa networks for dense IoT deployments*. Paper presented at the NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium.
- Sornin, N., Luis, M., Eirich, T., Kramp, T., & Hersent, O. (2015). *Lorawan specification*. LoRa alliance. <https://www.lora-alliance.org>
- Wollschlaeger, M., Sauter, T., & Jasperneite, J. (2017). The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0. *IEEE industrial electronics magazine*, 11(1), 17-27.

Visualización de conjuntos de datos de múltiples instancias

Jorge Eliecer Valencia-Duque¹, Carlos Mera², Lina Maria Sepúlveda¹

jeliecery@gmail.com; carlosmera@itm.edu.co; lmsepulveda@itm.edu.co

¹ Universidad de Medellín, Medellín, 050031, Antioquia, Colombia.

² Instituto Tecnológico Metropolitano (ITM), Medellín, 050013, Antioquia, Colombia

DOI: 10.17013/risti.39.84-99

Resumen: En el reconocimiento de patrones, los algoritmos de aprendizaje de múltiples instancias han ganado importancia puesto que evitan que el usuario tenga que delimitar, las imágenes de forma individual, para el reconocimiento de objetos. Esto supone una ventaja frente a los algoritmos de aprendizaje tradicional, puesto que disminuyen considerablemente el tiempo requerido para preparar el conjunto de datos. No obstante, una desventaja es que los conjuntos de datos resultantes suelen ser complejos, lo que dificulta su representación visual usando las técnicas tradicionales de visualización de información. Así, en este trabajo se propone una herramienta para la visualización y análisis de conjuntos de datos del paradigma de aprendizaje de múltiples instancias. La propuesta de visualización fue evaluada utilizando el criterio de expertos. Además, se realizaron diferentes pruebas que muestran que una correcta visualización puede ayudar a tomar decisiones sobre el conjunto de datos para mejorar la precisión de la clasificación.

Palabras-clave: Aprendizaje de múltiples instancias; Visualización de información; Análisis visual.

Visualization multi-instance data sets

Abstract: In pattern recognition, multiple-instance learning algorithms have gained importance since they avoid that the user must delimit, the images individually in order to recognize the objects. This is an advantage over traditional learning algorithms since these considerably reduce the time required to prepare the data set. However, a disadvantage is that the resulting data sets are often complex, making it difficult to visualize them using traditional information visualization techniques. Thus, this work proposes a tool for the visualization and analysis of data sets of the multi-instance learning paradigm. The visualization proposal was evaluated using the expert criteria. In addition, different tests were carried out that show that a correct visualization can help to make decisions about the data set to improve the classification precision.

Keywords: Multi-instance learning; Information visualization; Representation; Visual Analysis.

1. Introducción

El aprendizaje de múltiples instancias es un paradigma de clasificación supervisado en el que, en el contexto del reconocimiento de patrones, los objetos complejos pueden ser representados usando múltiples vectores de características (Herrera et al., 2016). Esta peculiaridad lo diferencia del aprendizaje supervisado tradicional, en el que los objetos sólo se pueden representar por un único vector de características. Así, el paradigma de aprendizaje de múltiples instancias llamado MIL por sus siglas en inglés (*Multiple Instance Learning*), proporciona un marco de trabajo que permite preservar mayor cantidad de información de los objetos que se desean reconocer.

Los conjuntos de datos de múltiples instancias poseen una estructura compleja donde los objetos son representados por *bolsas* que contienen múltiples vectores de características (llamados *instancias*), de ahí que se les denomine conjuntos de datos de múltiples instancias (MI). Este tipo de estructura incrementa la dificultad en la visualización del conjunto de datos, debido que además de visualizar las relaciones que existen entre las instancias individuales, se deben visualizar las relaciones que existen entre las bolsas, en otras palabras, las relaciones entre los conjuntos de instancias.

Como área de investigación la visualización de información busca el desarrollo de herramientas que ayuden a descubrir patrones ocultos en los datos, siendo un instrumento esencial para la toma de decisiones en lo que respecta a la Industria 4.0 (Arancegui & Laskurain, 2016; Janvrin, Raschke, & Dilla, 2014). En general, son diversos los campos de aplicación de la visualización de información, por ejemplo, en el análisis de datos de mercados para detectar anomalías o tendencias de mercados bursátiles (Yang, Gao, & Cao, 2013) o en el análisis de datos deportivos a fin de estimar el rendimiento de los jugadores durante un encuentro determinado (Cheplygina & Tax, 2015).

Si bien las técnicas de visualización han sido adaptadas para explorar y analizar conjuntos de datos multidimensionales tradicionales, el uso de estas técnicas es poco explorado en la representación visual de conjuntos de datos de MI. Esto se debe, como se dijo anteriormente, a la complejidad que representa su visualización. Por otra parte, son conjuntos de datos relativamente recientes y poco explorados, además que su uso aún no está generalizado entre los investigadores.

Con base en lo anterior, en este trabajo se propone una estrategia de visualización para la exploración y reducción de dimensionalidad de conjuntos de datos de MI. El resto del documento se encuentra distribuido de la siguiente forma: en la Sección 2 se presenta la fundamentación teórica del paradigma MIL y se hacen algunas consideraciones acerca de la visualización de información. En la Sección 3 se describe el desarrollo de la propuesta de visualización de conjuntos de datos de MI. En la Sección 4 se presentan los resultados internos y externos obtenidos con el método de visualización propuesto. Por último, en la Sección 5, se presentan las conclusiones y el trabajo futuro.

2. Fundamentación teórica

2.1. Aprendizaje de múltiples instancias (MIL)

En el paradigma MIL, como fue propuesto por Dietterich, Lathrop, & Lozano-Pérez (1997), un objeto es representado por una bolsa y cada instancia de la misma contiene

la información extraída de una parte del objeto (Dietterich, Lathrop, & Lozano-Pérez, 1997). Formalmente, en un problema de dos clases, un conjunto de datos MI tiene la forma: $T = \{(B_1, y_1), \dots, (B_N, y_N)\}$, donde $y_i \in \Omega = \{-1, +1\}$ es la etiqueta de clase de la bolsa $B_i = \{x_{i1}, \dots, x_{in_i}\}$, la cual contiene n_i instancias y cada instancia x_{ij} es un vector de características en un espacio d -dimensional ($\mathbb{R}^d$) (Mera, Orozco-Alzate, Branch, & Mery, 2016). Basado en esto, el propósito de los algoritmos MIL es aprender una función de transformación $F(B): B \rightarrow \Omega$ la cual determine la etiqueta de clase del objeto que representa una bolsa B .

Existe un supuesto, llamado el *supuesto estándar*, que indica que una bolsa es positiva si al menos una de sus instancias es positiva; al contrario, una bolsa es negativa si todas sus instancias son negativas (Dietterich et al., 1997). De acuerdo con lo anterior, un clasificador MIL toma la forma de la ecuación (1).

$$F(B_i) = \begin{cases} +1, & \text{si } \exists x_{ij} \in B_i | f(x_{ij}) = +1 \\ -1, & \text{de otro modo} \end{cases} \quad (1)$$

Nótese que F depende de un clasificador de instancia $f: \mathbb{R}^d \rightarrow \Omega$. Esto significa que aunque los datos de entrenamiento son pasados como un conjunto de bolsas, el clasificador f debe aprender en el ámbito de las instancias usando las etiquetas de las bolsas porque las etiquetas de las instancias individuales no se conocen (Weidmann, Frank, & Pfahringer, 2003). La Figura 1 ilustra la diferencia entre el aprendizaje supervisado tradicional y el aprendizaje de múltiples instancias. En el primero, un objeto se presenta por una sola instancia y en el segundo se representa por una bolsa que contiene múltiples instancias.

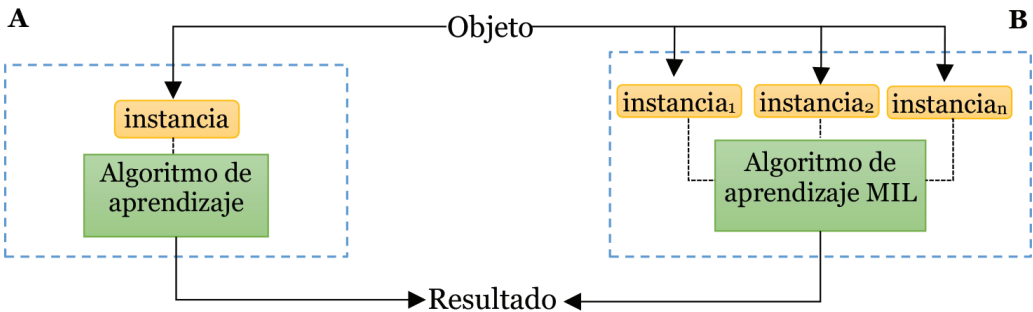


Figura 1 – A. Escenario tradicional del aprendizaje supervisado. B. Escenario de un aprendizaje de múltiples instancias. Imagen basada en la Figura 1 de (Foulds & Frank, 2010)

Con base en lo anterior, el proceso de aprendizaje de un algoritmo de clasificación en el paradigma MIL consiste en construir un modelo que a partir de un conjunto de bolsas de entrenamiento aprenda a predecir las etiquetas de clase de nuevas bolsas (Foulds & Frank, 2010). De acuerdo con Amores (Amores, 2013), los algoritmos MIL pueden ser agrupados en tres familias: basados en instancias (IS), basados en bolsas (BS) o basados en espacios embebidos o transformados (ES) (Amores, 2013; Mera et al., 2016).

La familia de algoritmos IS busca predecir las etiquetas de clase de una bolsa a partir de las etiquetas de sus instancias. Para ello, estos algoritmos consideran diferentes supuestos, siendo el supuesto estándar uno de los más utilizados (Foulds & Frank, 2010). Algunos algoritmos MIL que hacen uso de este paradigma son *Multiple-Instance Support Vector Machines* (MI-SVM) (Andrews, Tsochantaridis, & Hofmann, 2003) y *Sparse MIL* (SMIL) (Bunescu & Mooney, 2007).

Los algoritmos que pertenecen a la familia BS, toman en cuenta las bolsas como un todo y definen medidas de similitud o de distancia entre bolsas, lo que les permite definir relaciones espaciales entre bolsas y clases (Herrera et al., 2016). Además, no hay suposiciones sobre las instancias de las bolsas como sucede en IS. Usualmente, son clasificadores basados en distancias como el algoritmo de k vecinos más cercanos (k -NN), que en MIL se designa *Citation k-NN* (Amores, 2013; Foulds & Frank, 2010).

La familia de algoritmos ES usa funciones de transformación donde a partir de las instancias de una bolsa generan un nuevo vector de características, proyectando la bolsa a un nuevo espacio dimensional. De esta manera, un problema MIL se convierte en un problema de aprendizaje supervisado estándar (Amores, 2013). Algunos de los algoritmos MIL que emplean este paradigma son *Simple-MIL* que usa estadísticas de primer orden para transformar las bolsas (Mera et al., 2016); y *MILES* (Ma, Xu, Wu, Wang, & Chen, 2017).

2.2. Visualización de información

La visualización de información es un área de investigación que tiene como objetivo el desarrollo de herramientas que proporcionan una manera accesible de ver y comprender tendencias, valores atípicos y patrones en los datos. Lo anterior ayuda a los usuarios a explorar, entender y analizar datos a través de representaciones visuales interactivas (Liu, Cui, Wu, & Liu, 2014).

Debido al auge del análisis de grandes cantidades de datos, las herramientas que permiten gestionarlos para comprenderlos de manera profunda han ganado relevancia. Estas herramientas resultan valiosas para las empresas, debido a que permiten mejorar procesos impulsando la innovación e incrementando la productividad y los beneficios.

La visualización de información es una respuesta a una serie de retos acerca de cómo representar, interpretar y comparar datos de forma rápida y efectiva. La visualización de datos se puede presentar de diferentes maneras: estática o interactiva, dependiendo de las necesidades que se tengan, reduciendo los tiempos de respuesta a problemas puntuales (Janvrin et al., 2014).

Para que los datos puedan ser visualizados, es necesario contar con un proceso que permita transformarlos en información visual para el usuario. Usualmente, este proceso se constituye en cinco etapas: transformación de datos y análisis, filtrado, proyección, representación y controles de interfaz de usuario (Liu et al., 2014), como se ilustra en la Figura 2. En este proceso, la interacción del usuario es parte fundamental en todas las etapas de una visualización, puesto que esta permite al usuario hacer una

retroalimentación continua a la herramienta para lograr una visualización final de la que el usuario pueda extraer información relevante.

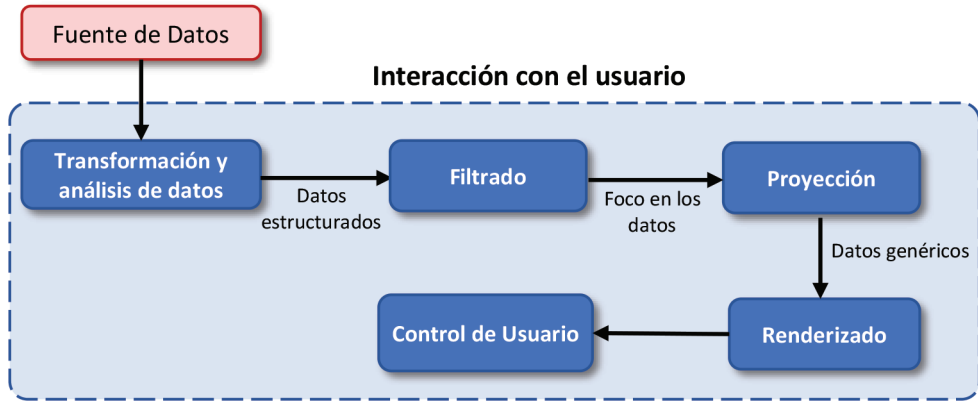


Figura 2 – Flujo de una visualización. Imagen construida con base en la Figura 1 de (Liu et al., 2014)

3. Visualización de conjuntos de datos de múltiples instancias

Las herramientas de visualización son usadas para extraer información de un conjunto de datos; sin embargo, la construcción de esas herramientas es un reto en lo que respecta a cómo representar visualmente, en un plano 2D o 3D, un conjunto de datos d -dimensional (Chan, 2006), más cuando ese conjunto de datos tiene una estructura compleja, como ocurre con los conjuntos de datos del paradigma MIL.

Por lo anterior, la visualización de un conjunto de datos de MI debe seguir un flujo que permita transformar los datos para que los métodos de visualización puedan representar la mayor cantidad de información posible. Para lograr este objetivo se utilizó como base el flujo de visualización que muestra la Figura 2. A continuación, se detalla el proceso seguido para lograr la propuesta de visualización de conjuntos de datos de MI y del desarrollo de las actividades que permitieron su construcción.

3.1. Selección de técnicas de visualización que mejor se adaptan a los conjuntos de datos de MI

Aunque la lista de métodos de visualización es bastante amplia, estos se pueden agrupar de acuerdo con su característica predominante, así:

- Proyecciones geométricas (Chan, 2006): Diagramas de dispersión (*Scatterplot*) (Cleveland, McGill, & Cleveland, 2011), Coordenadas Paralelas (*Parallel Coordinates*) (Inselberg, 1985)
- Técnicas orientadas a los píxeles (*Pixel-Oriented Techniques*) (Chan, 2006): Segmento Circular (*Circle Segment*) (Ankerst, Keim, & Kriegel, 1996), Gráfico de barras de píxeles (*Pixel Bar Chart*) (Keim & Kriegel, 1996)

- Visualización Jerárquica (Chan, 2006): *Treemap* (Wang, Wang, Dai, & Wang, 2006)
- Iconografía (Chan, 2006): *Star Glyph* (Cleveland & McGill, 1984)

Para el caso de los conjuntos de datos de MI, uno de los principales criterios para escoger un método de visualización es que este permita representar las relaciones que existen, no solo entre las instancias, sino también entre las bolsas del conjunto de datos. Otra característica deseable es que permita analizar más de una característica a la vez para poder comparar sus distribuciones y determinar cuáles pueden tener una mayor incidencia en el proceso de clasificación de las bolsas.

Basado en lo anterior, los métodos de visualización que pueden ser útiles para representar conjuntos de datos de MI son: diagramas de dispersión, gráficos de radar, gráficos de barras, mapas de calor, diagramas de red, gráficos de coordenadas paralelas y gráficos de columna radial.

3.2. Propuesta de visualización de conjuntos de datos de MI

Para iniciar con el desarrollo de la propuesta se seleccionó un grupo de conjuntos de datos de MI de prueba. Las áreas de aplicación de estos conjuntos de datos son diversas, siendo las más comunes la predicción de la actividad molecular, el etiquetado de imágenes, la categorización de texto, el reconocimiento de imágenes médicas y la clasificación de grabaciones de audio y video. Por lo anterior, los conjuntos que se usaron en la construcción de la herramienta son: *Musk1* (Dietterich et al., 1997) y *Fox* (Andrews et al., 2003). Estos son algunos de los conjuntos de datos más empleados en el paradigma MIL.

En la literatura también se encuentran conjuntos de datos de MI artificiales contruidos a partir de distribuciones Gaussianas (Carbonneau, Cheplygina, Granger, & Gagnon, 2018). Estos conjuntos artificiales ofrecen un mayor control sobre las características de los datos y permiten analizar más fácilmente el comportamiento de los algoritmos MIL y su visualización. Es por esto que se utilizó un conjunto de datos artificiales basados en Gaussianas de tres dimensiones.

Por la complejidad de los conjuntos de datos de MI se hace necesario tratar de simplificarlos sin perder información en el proceso. El uso de estrategias de reducción de la dimensionalidad permite remover las características menos relevantes en un conjunto de datos. A la vez, estas estrategias ayudan a mejorar el rendimiento de los algoritmos de clasificación y pueden incrementar la comprensión de los datos (Huang, Wu, & Ye, 2019).

Las técnicas de reducción de dimensionalidad se pueden aplicar de dos formas distintas a los conjuntos de datos de MI: sea en el espacio de las instancias o en el espacio de las bolsas. Cuando se aplica la reducción en el espacio de las instancias se omite la información de las bolsas, es decir, todo el conjunto de datos se trata como un conjunto de datos de aprendizaje supervisado tradicional en el que cada instancia hereda la etiqueta de clase de la bolsa a la que pertenece. Por otro lado, cuando se aplica la reducción de dimensiones al nivel de las bolsas, se toma cada bolsa como un conjunto de datos tradicional individual, el método considera las instancias de cada bolsa por separado y, al final, se forma un nuevo conjunto de datos con los resultados de cada subconjunto de forma independiente. De los experimentos iniciales se encontró que la reducción de dimensionalidad al nivel de

las bolsas no tiene un funcionamiento adecuado debido a que se forman subconjuntos de información que, intuitivamente, parecen no guardar relación entre sí.

Las técnicas de reducción de dimensionalidad usadas en la etapa de transformación de los datos fueron: *Kernel Principal Component Analysis* (KPCA) (Schölkopf, Smola, & Müller, 1997), *Fast Independent Component Analysis* (FastICA) (Hyvärinen & Oja, 2000), *Isometric Feature Mapping* (Isomap) (Tenenbaum, De Silva, & Langford, 2000), *Locally Linear Embedding* (LLE) (Roweis & Saul, 2000), *Multidimensional Scaling* (MDS) (Kruskal, 1964) y *T-Distributed Stochastic Neighbor Embedding* (TSNE) (Maaten & Hinton, 2008).

También, se implementó una estrategia de reducción basada en *Kernel Density Estimator* (KDE), como se propuso en (Mera, Orozco-Alzate, & Branch, 2014). KDE es un estimador no paramétrico de densidades univariadas o multivariadas que puede ser usado para reducir el número de instancias en las bolsas, conservando las instancias más positivas y negativas (Mera et al., 2014). Otras estrategias usadas para reducir el número de instancias en las bolsas fueron los métodos estadísticos de primer orden para extraer los máximos, mínimos, promedios y extremos de cada bolsa. Estas estrategias permiten reducir el número de instancias que contiene cada bolsa y se consideró como una simplificación de esta.

En la Figura 3 se muestra la estrategia desarrollada para transformar los datos haciendo uso de las técnicas de reducción y simplificación de bolsas. El proceso parte de un conjunto de datos de MI a partir del cual se pueden usar dos flujos, bien sea de manera independiente o combinándolos. En el caso de la reducción de datos se cuenta con la opción de elegir la cantidad de componentes (características resultantes) que se quiere obtener y se habilita la reducción en el espacio de bolsas o de instancias. Por otro lado, las técnicas de simplificación de bolsas permiten reducir el número de instancias en la bolsa. Estas se pueden aplicar posterior a una reducción de datos o de manea independiente sobre el conjunto de datos original. Al finalizar el proceso de transformación se obtiene el conjunto resultante.

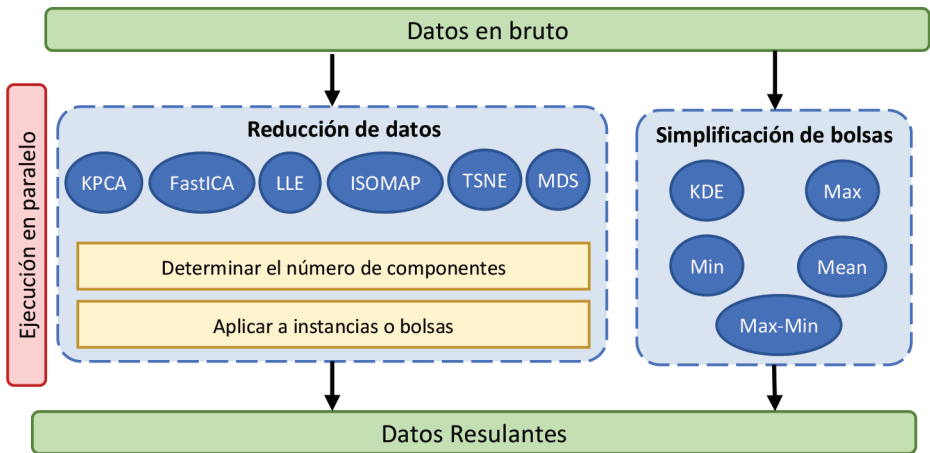


Figura 3 – Estrategia de reducción y simplificación de datos

El flujo anterior se integró a una aplicación web. Como se muestra en la Figura 4, los controles implementados tienen los métodos de reducción y simplificación mencionados. Además, se agregaron controles extra para la manipulación de los datos originales y los datos resultantes después de la reducción.

Para permitir al usuario analizar los conjuntos de datos de MI, se agregó un gráfico que muestra la distribución de las instancias negativas y positivas de cada una de las dimensiones del conjunto de datos resultante.

Para efectos de un análisis visual del conjunto de datos se propone el uso de dos gráficos. El primero permite visualizar las relaciones entre las bolsas y es un grafo dirigido por fuerzas con dos tipos de nodos: los círculos rojos, que representa las bolsas positivas y los triángulos azules que representan las bolsas negativas (Figura 5). El tamaño de cada nodo está definido con base en la cantidad de instancias, así entre más instancias tiene una bolsa, mayor es el tamaño de la figura que la representa. En el lateral de la visualización se dispusieron dos barras que indican el número de conexiones de las bolsas positivas y negativas en una escala de intensidad por color. Como elemento de interacción se implementaron diferentes algoritmos que cambian la disposición de los nodos del grafo.

Por defecto, en la visualización se usa el algoritmo *Fruchterman-Reingold force-directed (spring)*, el cual busca minimizar la energía del sistema moviendo los nodos y cambiando las fuerzas entre dos nodos cualquiera. En este sentido, la suma de los vectores de fuerza determina en qué dirección debe moverse un nodo para posicionarlo automáticamente de manera que se minimice la energía total del sistema. El grafo se estabiliza cuando el sistema alcanza su estado de equilibrio, es decir cuando se minimiza por completo la energía del sistema (Kobourov, 2012).

Como el peso de los vértices está definido con base en la distancia entre las bolsas, y dicha distancia se calcula con base en las instancias entre las bolsas (Amores, 2013), entonces se puede escoger si el peso de un vértice se asigna con base en la distancia mínima, máxima o la media entre las distancias de las instancias en dos bolsas. Por otro lado, la métrica de distancia usada es la distancia Euclidiana, aunque se tienen implementados diferentes métricas en caso de que el usuario considere más conveniente usar otra. Algunas de estas son: *braycurtis*, *chebyshev*, *correlation*, *euclidean*, *haming* y *mahalanobis*.

El segundo gráfico de la propuesta de visualización corresponde a dos diagramas de radar. Uno para visualizar las instancias de las bolsas positivas y otro para visualizar las instancias de las bolsas negativas. Cada uno de estos diagramas muestra la distribución de las instancias respecto a las características seleccionadas. En la Figura 6 se puede observar la representación de un conjunto Gaussiano, en el cual se observa que las características de las instancias de las bolsas negativas (derecha) tienen valores cercanos a uno; cabe aclarar que esos valores están normalizados para una representación uniforme entre los diferentes conjuntos de datos. Respecto a las instancias de las bolsas positivas (izquierda) el gráfico permite identificar que algunas de las instancias tienen valores cercanos a uno en sus características (al igual que las instancias de las bolsas negativas); esto sugiere que dichas instancias pueden ser negativa y se consideran instancias ruidosas en las bolsas positivas.

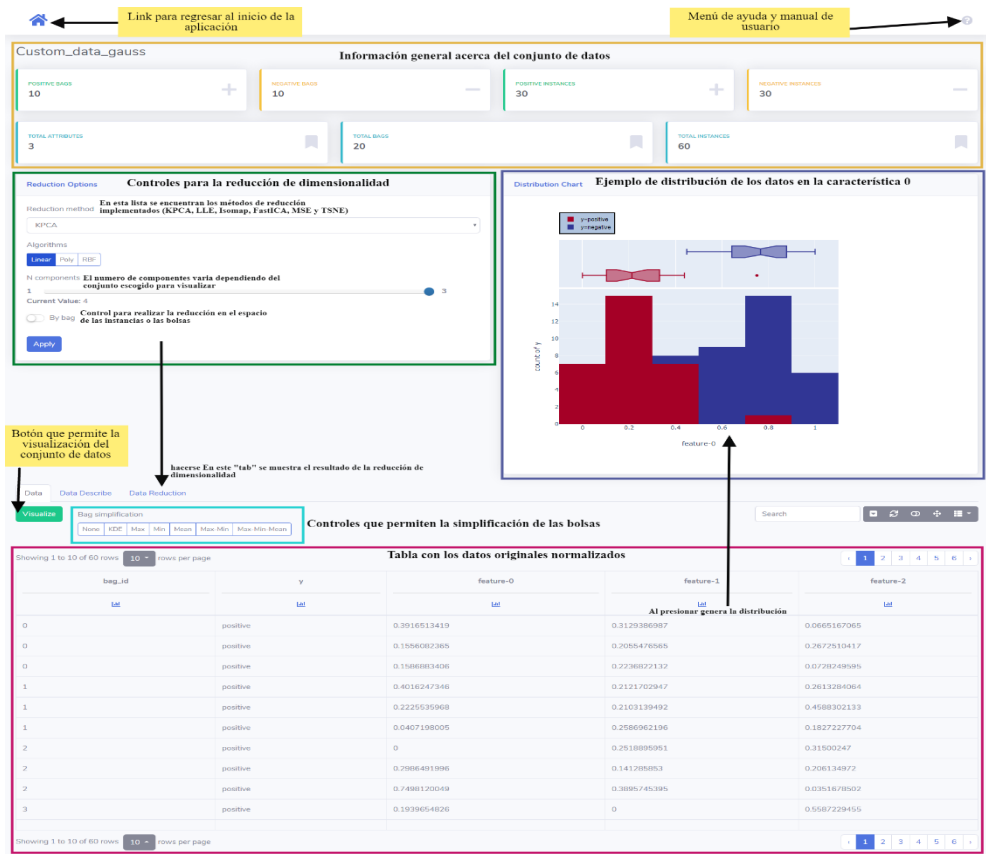


Figura 4 – Herramienta para preparación de conjuntos de datos MI.

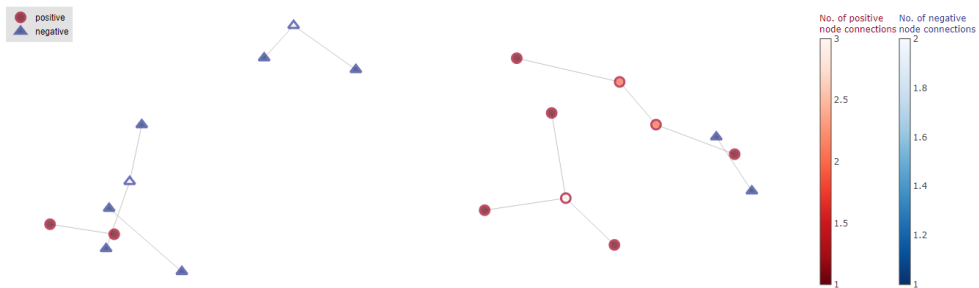


Figura 5 – Visualización usando un grafo dirigido por fuerzas de un conjunto de datos MI aussiano.

Los gráficos propuestos permiten, rápidamente, determinar qué características son predominantes entre las instancias, lo que es útil para determinar qué atributos pueden tener mayor influencia en la definición y la distinción entre las bolsas positivas y las negativas. No obstante, una de las debilidades de esta representación se presenta cuando se seleccionan bolsas con demasiadas instancias, puesto que por la naturaleza del gráfico este se torna más difícil de leer conforme aumenta el número de instancias.

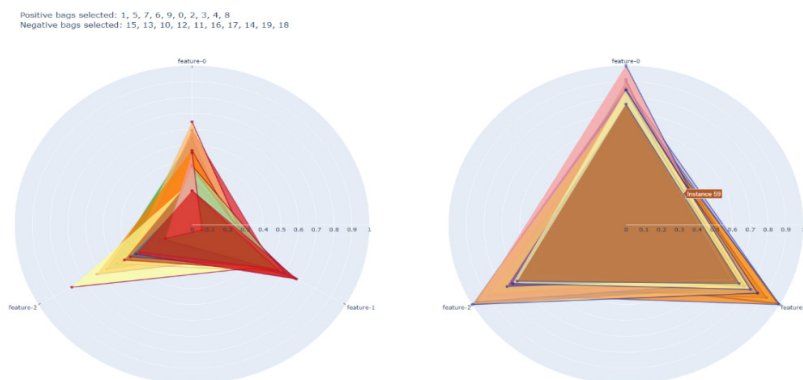


Figura 6 – Gráfico de radar que muestra la distribución de las instancias de un subconjunto de bolsas seleccionado.

3.3. Evaluación de la técnica de visualización propuesta

Para evaluar la propuesta de visualización se realizaron dos tipos de validaciones, una externa y otra interna. La externa fue realizada por expertos en el paradigma MIL que evaluaron desde dos puntos de vista la herramienta. El primero se centró en la usabilidad y el segundo se enfocó en la utilidad de la herramienta. En la evaluación interna se valoró si las transformaciones realizadas a los conjuntos de datos mejoraban la precisión de un clasificador MIL. Los resultados de estas evaluaciones se presentan en la Sección 4.

4. Resultados

4.1. Resultados de la evaluación interna

Esta evaluación se enfocó en valorar si existe alguna mejora en la comprensión del conjunto de datos midiendo el impacto de las transformaciones que realiza el usuario sobre el desempeño de un clasificador MIL. Específicamente, se utilizó el algoritmo *SimpleMIL* el cual pertenece a la categoría de espacios embebidos para transformar el problema de clasificación MIL a un problema de clasificación tradicional (Foulds & Frank, 2010). Para ello, las instancias de cada bolsa se embebieron a un nuevo espacio de características aplicando las funciones mínimo, máximo, promedio y extremos, entre las instancias de cada bolsa.

También se utilizaron tres métodos de reducción de dimensionalidad: KPCA (con un *kernel* lineal), Isomap y LLE. Igualmente, se aplicó un método para la reducción de

instancias, considerando la instancia más positiva y más negativa calculadas a partir de la estimación de densidad de la clase negativa con KDE. Las pruebas se hicieron sobre los conjuntos de datos *Musk1*, *Fox* y el conjunto de datos artificiales basado en Gaussianas. La Tabla 1 resume los resultados de clasificación sobre estos conjuntos de datos en términos de precisión usando una partición 80/20 para entrenamiento y validación respectivamente.

Método de reducción	SimpleMIL Max (%)	SimpleMIL Min (%)	SimpleMIL Promedio (%)	SimpleMIL Extremo (%)	SimpleMIL comparación (%)
Gaussiano					
Base	100,00	100,00	100,00	100,00	100,00
Isomap	40,00	80,00	70,00	80,00	100,00
KPCA	72,73	72,73	63,64	68,18	100,00
LLE	94,44	100,00	83,33	88,89	100,00
Musk 1					
Base	81,67	75,00	80,00	76,67	73,70
Isomap	60,00	65,00	53,33	55,00	73,70
KPCA	73,33	80,00	80,00	75,00	73,70
LLE	68,89	70,00	54,44	62,22	73,70
Fox					
Base	50,83	53,33	56,67	55,83	63,80
Isomap	40,83	55,00	57,50	58,83	63,80
KPCA	50,00	53,33	55,00	50,83	63,80
LLE	56,67	52,78	56,11	58,33	63,80

Tabla 1 – Comparación de la precisión promedio del clasificador *Simple MIL* usando los diferentes métodos de reducción de dimensionalidad en los conjuntos de MI, frente a la línea base de clasificación

Los resultados arrojados permitieron ver que en algunos casos se puede mejorar la clasificación si los atributos excluidos son los adecuados; en caso contrario la precisión del algoritmo disminuye. Esto se evidencia para los tres conjuntos usados, sin embargo, es más notorio en *Musk 1* y *Fox* que en el Gaussiano.

La exclusión de ciertas características en el conjunto de datos se hizo con base en la visualización de cada conjunto de datos. Para ello, se utilizó la visualización basada en grafos, la cual permitió detectar los grupos de bolsas que estaban relacionadas, de acuerdo con una métrica de distancia. A partir del grafo, se seleccionaron aquellas bolsas positivas y negativas que se encuentran conectadas y cercanas unas a otras.

Las bolsas seleccionadas fueron proyectadas usando el gráfico de radar el cual permitió identificar las características con valores predominantemente altos entre las instancias del conjunto de datos. Al retirar esas características se creó un subconjunto con las características de valores bajos con el que se entrenó el clasificador *SimpleMIL*.

Al realizar el procedimiento descrito, se encontró que al retirar las características que tienen valores altos es posible obtener una mejora leve en la precisión del algoritmo. Por el contrario, si se retiran las características con valores bajos, la precisión del algoritmo tiende a bajar. La mejora y la disminución en la precisión en cualquier caso está en un rango de 0 a 10 puntos, no obstante, esto depende del conjunto de datos usado. En las pruebas se observó que la precisión tiende a mejorar más en los conjuntos de datos de MI de problemas reales, mientras que para el conjunto de los datos Gaussianos no suele haber mejoras significativas.

En la Tabla 1 se muestra cómo varía la precisión promedio del clasificador Simple MIL cuando se aplican los métodos de reducción de dimensionalidad comparado con la línea base de referencia¹. Para el conjunto de datos Gaussianos, no es muy eficiente aplicar métodos de reducción, en especial Isomap y KPCA, que arrojan una precisión baja en comparación con los valores de referencia de la última columna y la primera fila. En cuanto a los conjuntos de datos Musk1 y Fox los métodos de reducción son un poco más efectivos, aunque no siempre es así con las diferentes variaciones del algoritmo de clasificación. En Musk1 KPCA es el método con mejores resultados, mientras en Fox LLE e Isomap muestran resultados satisfactorios.

4.2. Resultados de la evaluación externa

Para esta evaluación se solicitó a los expertos en MIL que realizaran una valoración sobre la usabilidad y el funcionamiento de la herramienta. Para ello se realizó una encuesta con 7 preguntas las cuales buscaban comprender mejor lo que los expertos opinaban acerca de la aplicación. En la evaluación de usabilidad se trataron de valorar de manera objetiva 10 apartados: características y funcionalidad, página de inicio, navegación, búsquedas, controles y retroalimentación, formularios, errores, contenido y texto, ayudas y rendimiento. Esta evaluación se basó en los 45 principios de usabilidad de mejores prácticas descritos en (Turner, 2011).

En la Tabla 2 se muestran las puntuaciones resultantes de cada experto sobre la usabilidad de la herramienta, dejando claro que esta aún debe mejorarse, bien sea creando instructivos más fáciles de entender o proporcionando herramientas de ayuda que guíen al usuario durante el flujo de la visualización.

Experto #	Puntaje general de usabilidad	Rango del puntaje
<i>Experto 1</i>	80	Bueno
<i>Experto 2</i>	62	Moderado
<i>Experto 3</i>	69	Moderado

Tabla 2 – Resultados generales para la revisión de usabilidad.

A continuación, se presentan algunas de las respuestas más relevantes que se obtuvieron en la encuesta realizada. No se muestran todas debido a que con estas opiniones es suficiente para hacerse una idea acerca de la propuesta de visualización.

Encuesta

¿Considera herramientas como esta útiles para el proceso de toma de decisiones en el paradigma de aprendizaje de múltiples instancias (MIL)?

“Claro, siempre será importante la exploración inicial y visualización de los datos. Herramientas como esta ayudan a tener una idea general de nuestros datos y a realizar transformaciones que son muy útiles en el aprendizaje de MIL.”

¿Los datos mostrados son precisos y útiles para el entendimiento en general del conjunto de datos de MI?

“Sí, pero falta mayor información descriptiva y visual acerca de los datos...”

¿La visualización responde claramente las preguntas planteadas inicialmente o plantea nuevas preguntas sobre los datos?

“Las visualizaciones no me parecieron claras. Es necesario acompañar las gráficas en la interfaz con texto adicional que facilite su interpretación...”

En resumen y con base en las respuestas dadas por los expertos a la encuesta realizada, se puede deducir que aún hay un camino de mejoras que se deben hacer en la visualización de conjuntos de datos de MI, esto muestra también que esta primera aproximación es una buena idea y un campo de investigación que podría ser de ayuda significativa en la exploración de conjuntos de datos complejos y ayudaría en un entendimiento mejor de los conjuntos de MI y los algoritmos MIL para su clasificación.

5. Conclusiones y trabajo futuro

La comparación de los diferentes métodos de visualización y la caracterización de los conjuntos de datos de MI permitió encontrar formas de representar estos datos y extraer información relevante para mejorar la precisión en el entrenamiento de un algoritmo MIL. Además, posibilitó el desarrollo de una herramienta de visualización que, pese a sus debilidades, puede ser de utilidad a los investigadores del área.

También se ha podido concluir que la reducción en conjuntos de datos de MI de pocas dimensiones puede resultar contraproducente en la clasificación, también que no todos los métodos de reducción de datos son efectivos o funcionan de igual manera en los diferentes tipos de conjuntos de datos de MI. Esto plantea algunas incógnitas, en especial las relacionadas con qué métodos de reducción podrían resultar más efectivos en conjuntos de datos de múltiples instancias.

Como trabajo futuro, se tiene planeado hacer esfuerzos en encontrar otros métodos de visualización que representen la estructura y las relaciones de los datos de mejor manera, además de profundizar más en las particularidades de la estructura de los conjuntos de datos de MI. Esto si se quiere tener un aporte más significativo a la industria 4.0 y ayudar de forma más profunda a la toma de decisiones. Por otro lado, en siguientes versiones de la herramienta se deben las interacciones necesarias para modificar los parámetros de los métodos de reducción. También se deben agregar y evaluar otros métodos de reducción como *Uniform Manifold Approximation and Projection* (UMAP).

Para finalizar, se comprobó que dependiendo del conjunto de datos usado la representación puede verse afectada, por lo que sería conveniente crear métodos que se adapten al tipo o estructura del conjunto de datos de MI y de esta forma obtener mejores resultados.

Referencias

- Amores, J. (2013). Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*, 201, 81–105.
- Andrews, S., Tsochantaridis, I., & Hofmann, T. (2003). Support vector machines for multiple-instance learning. In *Advances in neural information processing systems* (pp. 577–584).
- Ankerst, M., Keim, D., & Kriegel, H. (1996). “Circle Segments”: A Technique for Visually Exploring Large Multidimensional Data Sets. In *Proc. IEEE Visualization '96, Hot Topic Session* (pp. 5–8). <http://nbn-resolving.de/urn:nbn:de:bsz:352-opus-70761>
- Arancegui, M. N., & Laskurain, X. S. (2016). Reflexiones sobre la Industria 4.0 desde el caso vasco. *EKONOMIAZ. Revista Vasca de Economía*, 89(01), 142–173. <https://ideas.repec.org/a/ekz/ekonoz/2016106.html>
- Bunescu, R. C., & Mooney, R. J. (2007). Multiple instance learning for sparse positive bags. In *Proceedings of the 24th international conference on Machine learning* (pp. 105–112).
- Carbonneau, M. A., Cheplygina, V., Granger, E., & Gagnon, G. (2018). Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition*, 77, 329–353. <https://doi.org/10.1016/j.patcog.2017.10.009>
- Chan, W. W.-Y. (2006). A survey on multivariate data visualization. *Department of Computer Science and Engineering. Hong Kong University of Science and Technology*, 8(6), 1–29.
- Cheplygina, V., & Tax, D. M. J. (2015). Characterizing multiple instance datasets. In *International Workshop on Similarity-Based Pattern Recognition* (pp. 15–27).
- Cleveland, W. S., & McGill, R. (1984). Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*, 79(387), 531–554. <https://doi.org/10.1080/01621459.1984.10478080>
- Cleveland, W. S., McGill, R., & Cleveland, S. (2011). The Many Faces of a Scafferplot. *Faces*, 79(388), 807–822.
- Dietterich, T. G., Lathrop, R. H., & Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2), 31–71. [https://doi.org/10.1016/S0004-3702\(96\)00034-3](https://doi.org/10.1016/S0004-3702(96)00034-3)
- Foulds, J. R., & Frank, E. (2010). A review of multi-instance learning assumptions. *Knowledge Engineering Review*, 25(1), 1–25.

- Herrera, F., Ventura, S., Bello, R., Cornelis, C., Zafra, A., Sánchez-Tarragó, D., & Vluymans, S. (2016). *Multiple instance learning: Foundations and algorithms*. In *Multiple Instance Learning: Foundations and Algorithms*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-47759-6>
- Huang, X., Wu, L., & Ye, Y. (2019). A Review on Dimensionality Reduction Techniques. *International Journal of Pattern Recognition and Artificial Intelligence*, 33(10), 975–8887. <https://doi.org/10.1142/S0218001419500174>
- Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks: The Official Journal of the International Neural Network Society*, 13(4–5), 411–430. [https://doi.org/10.1016/S0893-6080\(00\)00026-5](https://doi.org/10.1016/S0893-6080(00)00026-5)
- Inselberg, A. (1985). The plane with parallel coordinates. *The Visual Computer*, 1(4), 69–91. <https://doi.org/10.1007/BF01898350>
- Janvrin, D. J., Raschke, R. L., & Dilla, W. N. (2014). Making sense of complex data using interactive data visualization. *Journal of Accounting Education*, 32(4), 31–48. <https://doi.org/10.1016/j.jaccedu.2014.09.003>
- Keim, D. A., & Kriegel, H. P. (1996). Visualization techniques for mining large databases: A comparison. *IEEE Transactions on Knowledge and Data Engineering*, 8(6), 923–938. <https://doi.org/10.1109/69.553159>
- Kobourov, S. G. (2012). Spring Embedders and Force Directed Graph Drawing Algorithms. <http://arxiv.org/abs/1201.3011>
- Kruskal, J. B. (1964). Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29(2), 115–129. <https://doi.org/10.1007/BF02289694>
- Liu, S., Cui, W., Wu, Y., & Liu, M. (2014). A survey on information visualization: recent advances and challenges. *The Visual Computer*, 30(12), 1373–1393. <https://doi.org/10.1007/s00371-013-0892-3>
- Ma, Y., Xu, J., Wu, X., Wang, F., & Chen, W. (2017). A visual analytical approach for transfer learning in classification. *Information Sciences*, 390, 54–69.
- Maaten, L. van der, & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov), 2579–2605.
- Mera, C., Orozco-Alzate, M., & Branch, J. (2014). Improving Representation of the Positive Class in Imbalanced Multiple-Instance Learning. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8814, pp. 266–273). Springer Verlag. https://doi.org/10.1007/978-3-319-11758-4_29
- Mera, C., Orozco-Alzate, M., Branch, J., & Mery, D. (2016). Automatic visual inspection: An approach with multi-instance learning. *Computers in Industry*, 83, 46–54. <https://doi.org/10.1016/j.compind.2016.09.002>

- Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500), 2323–2326. <https://doi.org/10.1126/science.290.5500.2323>
- Schölkopf, B., Smola, A., & Müller, K.-R. (1997). Kernel principal component analysis. In *International conference on artificial neural networks* (pp. 583–588).
- Tenenbaum, J. B., De Silva, V., & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500), 2319–2323. <https://doi.org/10.1126/science.290.5500.2319>
- Turner, N. (2011). A guide to carrying out usability reviews . Retrieved July 14, 2020, from <http://www.uxforthemasses.com/usability-reviews/>
- Wang, W., Wang, H., Dai, G., & Wang, H. (2006). Visualization of large hierarchical data by circle packing. In *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06* (Vol. 1, p. 517). ACM Press. <https://doi.org/10.1145/1124772.1124851>
- Weidmann, N., Frank, E., & Pfahringer, B. (2003). A two-level learning method for generalized multi-instance problems. In *European Conference on Machine Learning* (pp. 468–479).
- Yang, W., Gao, Y., & Cao, L. (2013). TRASMIL: A local anomaly detection framework based on trajectory segmentation and multi-instance learning. *Computer Vision and Image Understanding*, 117(10), 1273–1286. <https://doi.org/10.1016/j.cviu.2012.08.010>

Sistema de recomendación para contenidos musicales basado en el análisis afectivo del contexto social

Mauricio Sánchez Barragán¹, Gabriel Elías Chanchí G², Wilmar Yesid Campo M³

mausanbar93@gmail.com; gchanchig@unicartagena.edu.co; wycampo@uniquindio.edu.co

¹ Institución Universitaria Colegio Mayor del Cauca, Carrera 7 # 2-41, 19003, Popayán, Colombia.

² Universidad de Cartagena, Avenida del Consulado, calle 30 No. 39 B - 192, 130001, Cartagena de Indias, Colombia.

³ Universidad del Quindío, Cra. 15 Cl 12 norte, 630004, Armenia, Colombia.

DOI: 10.17013/risti.39.100-113

Resumen: En la actualidad gracias a la difusión de las redes sociales, se hace necesario aprovechar el contexto social de un usuario, con el fin de enriquecer la toma de decisiones en los sistemas inteligentes. Así, este artículo se centra en el estudio afectivo del contexto social de un usuario, para enriquecer la recomendación de contenidos multimedia musicales más relevantes. De este modo, se propone como aporte un sistema de recomendación de contenidos musicales, el cual relaciona el análisis sentimental del contexto social de un usuario a través de la red social twitter con el análisis sentimental de las letras de las canciones. Así, este artículo presenta los diferentes componentes asociados al sistema de recomendación, tales como: dataset de contenidos musicales, método computacional basado en un clasificador bayesiano encargado de la predecir contenidos musicales a partir del análisis del contexto social del usuario y servicio de música en línea.

Palabras-clave: Análisis de sentimientos; contenidos musicales; contexto; sistema de recomendación.

Recommender system for musical contents based on the affective analysis of the social context

Abstract: Nowadays, thanks to the diffusion of social networks, it is necessary to take advantage of the social context of a user, in order to enrich decision-making in intelligent systems. Thus, this paper focuses on the affective study of the social context of a user, to enrich the recommendation of more relevant musical multimedia content. In this way, we propose as a contribution a system of recommendation of musical contents, which relates the sentimental analysis of the social context of a user through the social network twitter with the sentimental analysis of the lyrics of the songs. Thus, this paper presents the different components associated to the recommendation system, such as: musical content dataset, computational method

based on a Bayesian classifier in charge of predicting musical contents from the analysis of the user's social context and online music service.

Keywords: context; musical contents; recommendation system; sentiment analysis.

1. Introducción

El concepto de las emociones en la música se relaciona al tipo de género o clase de contenidos que una persona escucha (Lundqvist, Carlsson, Hilmersson, & Juslin, 2009), donde incluso se considera la música como un lenguaje de emociones (Ahlberg, 1994) y la importancia de reconocer estas a través de la música y los efectos que genera en el ser humano (Patterson, Uppenkamp, Johnsrude, & Griffiths, 2002). Es por ello que se han abordado una serie de trabajos que permiten ver la importancia de las emociones con la música o contenidos de video. En este trabajo se consideran dos temáticas principales: sistemas multimedia basados en emociones y sistemas de análisis de las emociones.

En cuanto a la primera temática, existen trabajos relacionados con las emociones y contenidos multimedia, tales como (Rho, Han, & Hwang, 2009), donde los autores proponen la relación del estado de ánimo inferido a partir de variables de contexto de usuario, con la recomendación de contenidos musicales. De igual manera este trabajo asocia la emoción de usuario con el contenido a través de las características afectivas de energía y valencia, tradicionales en este tipo de estudios. En (Cunningham, Caulder, & Grout, 2008) definen 9 diferentes tipos de estados emocionales de usuario a partir de variables del contexto como: temperatura, cantidad de luz, entre otros; y los cuales generan uno u otro tipo de contenido multimedia musical previamente clasificado. Trabajos como (Perik, Ruyter, Markopoulos, & Eggen, 2004; Park, Yoo, & Bae, 2006; Chanchí, Campo & Arciniegas, 2018) proponen la recomendación de contenidos multimedia a partir del contexto, emoción o actividad física del usuario; donde unifican que las emociones ayudan en la clasificación y recomendación de contenidos multimedia. Adicionalmente, se tienen plataformas inteligentes de música, tales como: Spotify, rockolaFM y AllMusic las cuales proporcionan contenidos musicales al usuario, de acuerdo a una emoción o categoría; todas ellas se basan en las características musicales tales como: tempo, tono, energía, entre otros; para la asociación del contenido a una emoción. Las anteriores propuestas no consideran el uso del contexto social de usuario para la recomendación de contenidos musicales.

En lo referente a la segunda temática, en (Westerink, van den Broek, Schut, van Herk, & Tuinenbreijer, 2008) se presenta la forma de caracterizar las emociones (negativas, positivas, mixtas y neutras) a partir de señales biomédicas. Por otra parte, en (Sharma & Kapoor, 2008) se propone un sistema portátil para la medición de bioseñales que permiten el reconocimiento de emociones o estrés mental en usuarios, por medio de un algoritmo de autoaprendizaje. En (Patil, Singh, Singh, & Sharma, 2015) se presenta el estudio de diferentes técnicas para la evaluación y reconocimiento del estrés mental, permitiendo obtener datos que posibiliten la predicción del estado de ánimo de una persona.

De igual modo, trabajos como (Laurier, Grivolla & Herrera, 2008; Yang & Lee, 2009; Hu, Downie & Ehmann, 2010) muestran el estudio para la clasificación de contenidos musicales por emociones, de acuerdo al análisis del audio y componentes de la pista con la

letra; éste último presenta diferentes puntos de vista en las soluciones propuestas, ya que para algunos estados de ánimo difieren los resultados esperados en comparación con los obtenidos. Lo anterior, a causa de los diferentes tipos de géneros de las pistas musicales y los idiomas asociados a las letras de las canciones. Finalmente, en (Chanchí & Córdoba, 2019) se realiza un análisis afectivo del discurso de firma del acuerdo de paz en Colombia, mediante el análisis de sentimientos del texto del discurso y el análisis de emociones sobre el audio del mismo. El trabajo mostró concordancia a lo largo del tiempo entre la propiedad valencia de la pista de audio y la polaridad del sentimiento identificado en diferentes partes del discurso. Así, en este artículo se propone una combinación del estudio tanto auditivo como lingüístico de las pistas musicales con el contexto social del usuario, para la recomendación de contenidos musicales. Cabe resaltar que tanto a nivel del análisis del contexto social, como en el análisis de las letras de las canciones se toma en consideración los tres estados asociados al análisis de sentimientos: positivo, negativo y neutro.

De este modo, a partir de lo anterior, se plantea como aporte investigativo un sistema de recomendación (SR) de contenidos musicales, basado en el uso del análisis de sentimientos del contexto social de usuario y el estudio lingüístico de las pistas musicales, el cual busca enriquecer la recomendación de contenidos de los enfoques tradicionales.

2. Metodología

Para la conformación del SR de contenidos musicales construido, se proponen cuatro fases a saber: generación del dataset, método computacional para el análisis de sentimientos a partir del contexto social, diseño e implementación del SR de contenidos musicales basado en el contexto social y evaluación del SR a través de un servicio de música en línea (ver Figura 1). En la fase uno (I) se conformó el catálogo de contenidos musicales latinos a ser recomendados al usuario a través del SR; para ello se realizó la clasificación sentimental de los contenidos a partir del análisis emocional-lingüístico. En la fase dos (II) se hizo un análisis del contexto social del usuario por medio de información recolectada a través de twitter, proponiendo a partir de ello un método computacional para el análisis y clasificación del sentimiento asociado al contexto social. En la fase tres (III) se realizó el diseño e implementación del SR de contenidos musicales a partir del método computacional y los datos de entrada del usuario suministrados a través de la red social. Finalmente, en la fase cuatro (IV) el SR es evaluado mediante la implementación de un servicio de música en línea.



Figura 1 – Metodología considerada

3. Generación del catálogo musical

El dataset fue constituido a partir de la exploración de las fuentes de datos que permitieron obtener un listado clasificado de canciones latinas populares. Así, se

encontró que la revista Billboard es una fuente especializada en evaluar semanalmente las mejores canciones sugeridas por los usuarios de distintas plataformas musicales. Definida la fuente para la recolección, se estableció un mecanismo de clasificación de contenidos, por lo que se optó por el uso de herramientas para el análisis de sentimientos basado en la letra de las canciones. Lo anterior considerando que muchos usuarios se identifican con una canción no solo por la música, sino también por su letra. En la Tabla 1 se presentan un conjunto de APIs para el análisis y clasificación del contenido.

Librería	Descripción
<i>Text-Processing</i>	Permite el análisis sentimental de un texto, haciendo uso de un clasificador bayesiano que ayuda a determinar el grado de neutralidad del texto, y a partir de este último puede establecer la polaridad sentimental (positivo o negativo), brindando como resultado un valor de probabilidad para cada sentimiento (positivo, neutral y negativo). Para usar esta API ilimitadamente (1000 peticiones gratuitas) es necesario adquirir una llave registrándose en el sitio de Mashape.
<i>Sentiment140</i>	Tiene un enfoque orientado a la interpretación del sentimiento de pequeños textos informales (publicaciones en twitter, mensajes de productos, estados de personas, entre otros), los cuales permiten la clasificación de dichos textos en tres sentimientos: positivo, neutral y negativo, a través de un algoritmo de aprendizaje automático. Permite un acceso ilimitado a su algoritmo de clasificación, siempre y cuando no se utilice con fines comerciales (adquirir licencia de uso). El mensaje de respuesta que proporciona la API a las peticiones HTTP del usuario es formato JSON, y brinda soporte a dos idiomas: español e inglés.
<i>Sentiment</i>	API para el análisis sentimental de textos en idioma inglés, mediante la clasificación e interpretación con un clasificador bayesiano y otros métodos. Brinda en su sitio oficial una interfaz de pruebas, donde se puede agregar un texto con un número limitado de caracteres y posteriormente obtiene el sentimiento detectado (positivo, negativo y neutro) y un nivel de confianza en la clasificación.
<i>Repustate Sentiment</i>	Permite el análisis de sentimientos y extracción de información semántica de medios sociales, noticias, encuestas, blogs, foros, etc. Esta API permite el análisis sentimental en diferentes idiomas tal como: español, inglés, ruso, italiano, francés, holandés, entre otros. Para usarla ilimitadamente y sin fines comerciales es necesario adquirir una llave "key" registrándose en el sitio oficial de la API.

Tabla 1 – APIs para el análisis de texto

A partir de lo anterior, se pudo analizar cómo cada una de éstas brinda como aspecto común el análisis ilimitado y de textos en idioma español, lo cual es adecuado para la clasificación de contenidos musicales latinos. Del mismo modo cada una de estas API contempla tres niveles de polaridad para el análisis de sentimientos: positivo (1), neutro (0) y negativo (-1).

En la Figura 2, se presenta un diagrama de flujo del proceso de generación del dataset, donde se incluyen seis etapas: generación de listado de contenidos populares, obtención del contenido lírico, clasificación de contenidos por tipo de sentimiento, obtención URL YouTube, consolidación del catálogo y descarga de contenidos. Para generar el dataset musical se usó la API de Billboard para Python, en la cual se consultaron las canciones latinas más populares del último año.

En el paso “1” a través de la API de Billboard, se procede a generar un listado con las canciones latinas más populares del catálogo de la revista Billboard. Para lo anterior, un script se encarga de realizar la consulta de las 100 mejores canciones cada semana en el último año, seleccionando las de mayor ranking.

En el paso “2” se obtienen las letras de las canciones seleccionadas, para ello se llevó a cabo la correlación de los resultados obtenidos por distintas APIs para la consulta del contenido lírico, tales como: lyricwikia, PyLyrics y Musixmatch. Adicionalmente, se complementa el dataset con el contenido lírico de cada canción, para posteriormente realizar el análisis sentimental de cada contenido.

En el paso “3” se genera la clasificación de los contenidos por tipo de sentimiento (positivo, neutral o negativo), lo cual es realizado mediante los métodos matemáticos provistos por las APIs de la Tabla 1. Estas APIs brindan información sobre la polaridad de las letras de las canciones, por lo cual se correlacionaron los resultados y se seleccionó aquel con mayor posibilidad de éxito.

En el paso “4” se procede a obtener para cada contenido musical, la URL disponible en YouTube del video oficial asociado a este, para lo cual se hizo uso de la librería youtube-dl y la API de YouTube. La librería de youtube-dl permite filtrar, seleccionar, descargar el contenido de YouTube por medio de las características como la calidad, título, formato, categorías, fechas, entre otros. Con ayuda de dicha librería se realiza la búsqueda de los títulos para todos los videos en la categoría de música de la API de YouTube y se hace una comparación con los proporcionados por la API de Billboard. Para ello, son filtradas las canciones con una correlación más alta a nivel del texto, haciendo uso del método SequenceMatcher de la librería difflib de Python. De este modo, se seleccionan las URL que sobrepasen el valor de 0.8 de correlación y que correspondan al de mayor número de visitas.

En el paso “5”, una vez obtenidas las direcciones URL, se consolida el dataset en un archivo JSON, administrado por la base de datos libre de Python “tinydb”. Finalmente, en el paso “6” se descargan los contenidos musicales en formato .mp3 haciendo uso de la librería youtube-dl. Así, fueron descargadas 120 canciones para la conformación del catálogo de contenidos musicales latinos, agrupando alrededor de 40 canciones por sentimiento (positivo, neutral y negativo).

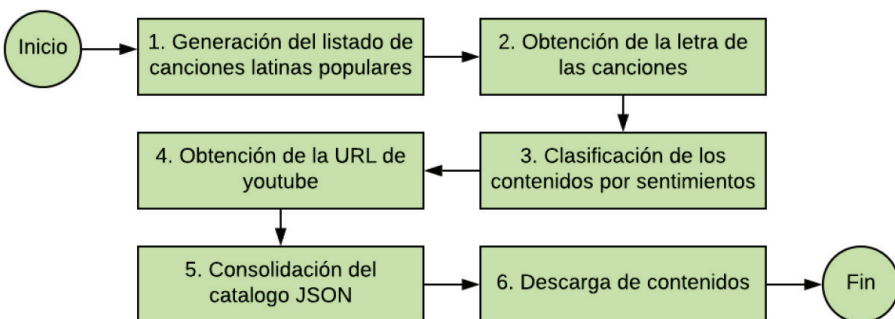


Figura 2 – Proceso de generación del dataset

En la Figura 3 se muestran los campos asociados a cada uno de los contenidos musicales que constituyen el dataset de contenidos musicales latinos. Cada campo de la pista es representado mediante un color, el cual se presenta en la Tabla 2.

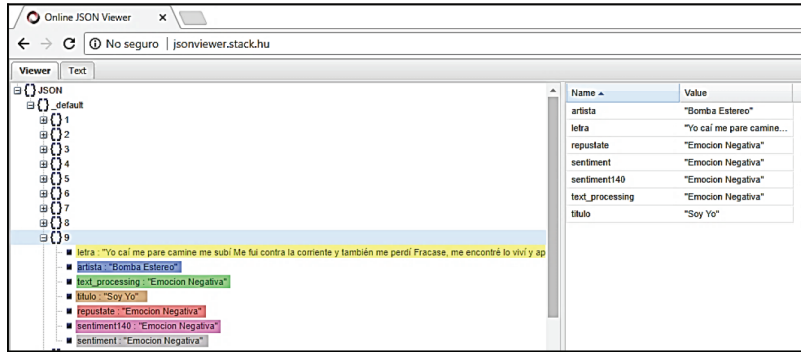


Figura 3 – Dataset generado

Cada contenido del dataset incluye los campos de la Tabla 2, dentro de los cuales se destaca la letra y los resultados de las APIs de la Tabla 1, las cuales se usaron para hallar la polaridad de los contenidos. Finalmente, se implementó un algoritmo basado en correlación para integrar los resultados de las cuatro (4) tecnologías.

Característica	Valor	Color
<i>Id</i>	9	
<i>Letra</i>	Yo caí me pare camine me subí..	
<i>Artista</i>	Bomba Estéreo	
<i>Título</i>	Soy yo	
<i>Text_processing</i>	Polaridad: negative	
<i>Sentiment</i>	Polaridad: negative	
<i>Sentiment140</i>	Polaridad: negative	
<i>Repustate</i>	Polaridad: negative	

Tabla 2 – Características musicales por contenidos

4. Análisis de sentimientos a partir del contexto social

El contexto de un usuario puede entenderse como cualquier información que puede ser utilizada para caracterizar el estado de una entidad, siendo la entidad, una persona, lugar u objeto relevante para las interacciones entre el usuario y la aplicación (Moreno, Segrera, López, Muñoz, & Sánchez, 2015). Dentro del contexto de usuario, se puede determinar tres aspectos importantes: dónde está el usuario, con quién está el usuario y qué recursos

hay cerca (Yang, 2009). El uso compuesto de los tres aspectos para caracterización del usuario permite obtener información precisa acerca de las interacciones del individuo. Es por ello que el estudio del contexto abre interrogantes sobre cómo utilizar el contexto de usuario para creación de sistemas inteligentes, los cuales a partir de las preferencias de un usuario recomienden información relevante (Furht & Agarwal, 2013).

4.1. Modelado del contexto de usuario

Para modelar el contexto del usuario, es necesario tener en cuenta los datos del contexto personal, social, de aplicación e histórico. Los datos del contexto personal incluyen elementos como: el perfil de usuario, ubicación del usuario, hora del día o de la semana (día de la semana, fin de semana, vacaciones, entre otros); actividades del usuario (caminar, dormir, hablar, correr, conducir, entre otros); información fisiológica del usuario (ritmo cardíaco, variabilidad del ritmo, sensibilidad de la piel, entre otros); dispositivos personales (móvil, reloj inteligente, tv, entre otros). Dentro de los datos del contexto social se incluyen listas de contactos, vínculos en redes sociales e interacciones y tipos de información compartida. En cuanto a los datos del contexto de aplicación se encuentran: tipos de servicios web utilizados, requisitos de ancho de banda y confiabilidad de cada servicio, tipos de protocolos y mecanismos de acceso necesarios para cada servicio. Dentro de los datos del contexto histórico están: información histórica de las redes sociales, páginas web que navega un usuario, videos y música preferida, compras que ha realizado, entre otros. Así, el modelado del contexto de usuario se puede llevar a cabo tomando como base los tipos de contexto y el uso de técnicas para el análisis de este.

4.2. Tecnologías para el análisis del contexto social

En la Tabla 3 se presenta un conjunto de APIs explorados para la obtención de las publicaciones en twitter de los usuarios del sistema musical. Mediante la exploración de estas APIs, se escogió la librería REST API, considerando que es la que presenta menos restricciones en el acceso a las publicaciones. La obtención de las publicaciones se hizo con el fin de inferir un sentimiento a partir del análisis de las publicaciones del usuario en twitter, para posteriormente recomendarle contenidos musicales previamente clasificados y dispuestos en el dataset.

API	Descripción
<i>Search API</i>	API utilizada para la búsqueda de tweets, información y eventos dentro de la red social durante los últimos 7 días.
<i>Streaming API</i>	Versión en tiempo real del conjunto de APIs de Twitter.
<i>Twitter REST API</i>	API que obtiene el histórico de tweets publicados por el usuario.

Tabla 3 – Tecnologías para el análisis del contexto social

5. Construcción del SR basado en contexto

El SR de contenidos musicales se diseñó a partir de un SR clásico tipo booleano basado en el algoritmo de Naive Bayes. El clasificador ordena el contenido relevante al estado afectivo del usuario e infiere sus posibles gustos musicales mientras interactúa con el

sistema. La Tabla 4 presenta un ejemplo de clasificador bayesiano, el cual considera la siguiente información relevante del dataset: título, ranking(R), semanas de ranking(S), número de visualizaciones(V) y calificación(C).

Nombre del video	R	S	V	C
<i>Robarte un beso – Carlos Vives & Sebastián Yatra</i>	1	1	1	0
<i>Es tarde – Juanes</i>	1	1	1	1
<i>Una lady como tú – Manuel Turizo</i>	0	1	0	1
<i>Imitadora – Romeo Santos</i>	0	0	1	?

Tabla 4 –Ejemplo del clasificador bayesiano propuesto

Para obtener las características del contenido multimedia, se consultan los parámetros musicales al dataset. Dado que los rangos de valores de las variables utilizadas: ranking Billboard, semanas de ranking en Billboard y número de visualizaciones en YouTube, corresponden a valores no booleanos, es necesario discretizar sus valores (“0” y “1”), para ser usados en el clasificador bayesiano. Para ello se utiliza la ecuación 1, la cual permite discretizar como 1 un valor cuya resta con la media es mayor a 0 y como 0 si la resta con la media es menor a 0.

$$Valor_{Discret} \begin{cases} valor - media > 0 \Rightarrow 1 \\ valor - media \leq 0 \Rightarrow 0 \end{cases} \quad (1)$$

En el caso de la calificación, también es necesario discretizar ya que sus valores oscilan entre 1 y 5, para ello se considera como un “1” las calificaciones mayores a 3 y con un “0” las menores o iguales a este valor. En la Ecuación 2 se presenta la ecuación para discretizar en un valor booleano la calificación.

$$Calificación_{Discret} \begin{cases} calificación > 3 \Rightarrow 1 \\ calificación \leq 3 \Rightarrow 0 \end{cases} \quad (2)$$

Los valores de ranking, semanas, visualizaciones y calificación discretizados conforman los atributos de entrada del clasificador booleano de Naive Bayes, el cual fue implementado en Java usando la API de la herramienta de análisis de datos weka. Un ejemplo para la obtención de los valores booleanos se muestra en la Tabla 5, donde se tiene para una canción con los siguientes valores de ranking, semanas, visualizaciones y calificación como se aprecia a continuación. Luego del cálculo de la lista de valores booleanos, el clasificador predice la posible valoración que el usuario daría al contenido “Imitadora – Romeo Santos” (columna C de la Tabla 5) teniendo como base calificaciones anteriormente realizadas, generando un listado de videos ordenados según la hipótesis de clasificación más probable.

Nombre del video	R	S	V	C
Robarte un beso – Carlos Vives & Sebastián Yatra	1	1	1	0

Tabla 5 – Ejemplo del clasificador bayesiano propuesto

A continuación, se presenta un esquema de las herramientas utilizadas en la construcción (ver Figura 4) del prototipo del servicio. Este servicio está compuesto por varios componentes tal como: una interfaz web, servidor lógico, sistema de recomendación, base de datos y servidor de contenidos.

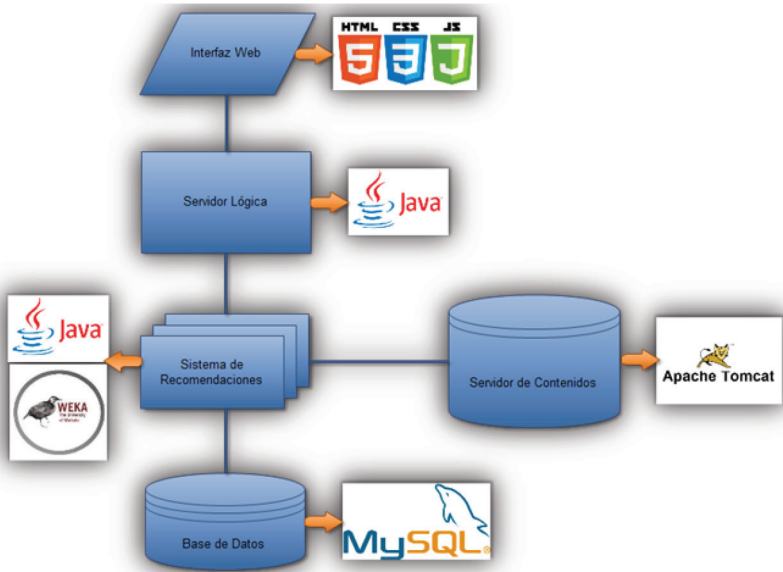


Figura 4 – Esquema funcional del sistema de recomendación

Dentro de los componentes del SR que permitieron la caracterización y recomendación de contenidos al usuario, se encuentra la plataforma y red social de twitter y su widget de chats. Para ello es necesario tener vinculada la cuenta de SRMusical con una cuenta de usuario en Twitter. Como medio de prueba se utilizó una cuenta no oficial “@UsuarioSR1”. La cuenta de Twitter tiene tanto interfaz web como móvil donde el usuario interactúa con los eventos, notificaciones y las publicaciones de otros y propias. Cabe mencionar que el fin de tomar como base la red social de Twitter es poder extraer a partir del análisis computacional afectivo de cada tweet que el usuario realice o replique. Una vez registrada la cuenta de prueba en la red social se procedió con la cuenta “admin” a registrar los datos asociados a cada usuario en la plataforma SRMusical (nombres, cuenta de twitter, contraseña, correo). Cuando el usuario entra a la interfaz del sistema y se valida con los datos puede acceder a la interfaz principal del sistema la cual se presenta en la Figura 5.

En la pantalla del menú principal se puede observar tres tipos de componentes: un listado de contenidos recomendados, un listado con los tweets publicados o retweets en tiempo real del usuario y un componente de reproducción de los contenidos musicales. Es así como finalmente, se realiza la recomendación de contenidos musicales latinos al usuario a través del servicio de música en línea que tiene como base el SR

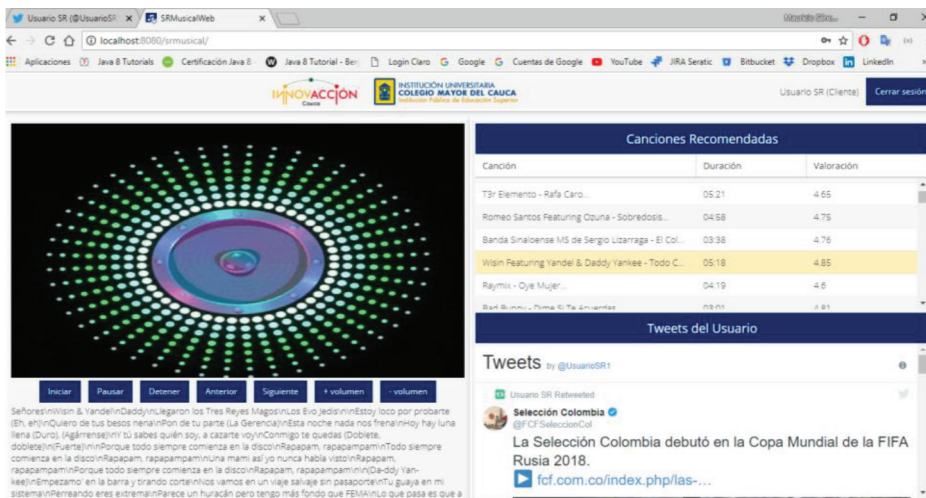


Figura 5 – Interfaz principal del sistema de música en línea

6. Evaluación del SR

La conformación del sistema de recomendación y los componentes que permiten la aplicación y uso de las recomendaciones de canciones, se pudo establecer a partir del servicio de música en línea el cual facilita la navegación y pruebas del sistema de recomendación; para ello se analizaron las recomendaciones que el sistema proporcionó al usuario a partir de un mensaje o tweet de entrada, el cual puede estar vinculado a un tipo o estado de emoción (negativa, neutral y positiva) que siente el usuario al momento de publicar el tweet. Las medidas se pudieron definir a partir de los datos recolectados de cuatro (4) diferentes usuarios registrados y los cuales estuvieron interactuando con el servicio de música en línea durante un período de tiempo mientras mantenían igual interacción en la red social de Twitter.

6.1. Promedio de predicciones exitosas vs fallidas por emoción

La primera evaluación que se realizó corresponde al número de predicciones exitosas en comparación a las fallidas por cada sentimiento de entrada asociado a los tweets de los usuarios. En la Figura 6 se pueden observar los resultados del análisis, donde se observa que para el sentimiento negativo el número de predicciones fallidas fue mayor que a los otros sentimientos, esto puede deberse a que el sistema al arrancar necesita como insumo un sentimiento de entrada al sistema y en su mayoría los usuarios no

se sintieron identificados con la polaridad negativa, por lo tanto la valoración de las canciones negativas fue variante e inferior, lo cual hace que la probabilidad de recomendar contenidos musicales con polaridad negativa sea variante para el SR. Adicionalmente, la predicción promedio para las polaridades positiva y neutral fue superior a las 70 predicciones de los 80 contenidos musicales que hacen parte de cada sentimiento en el dataset.

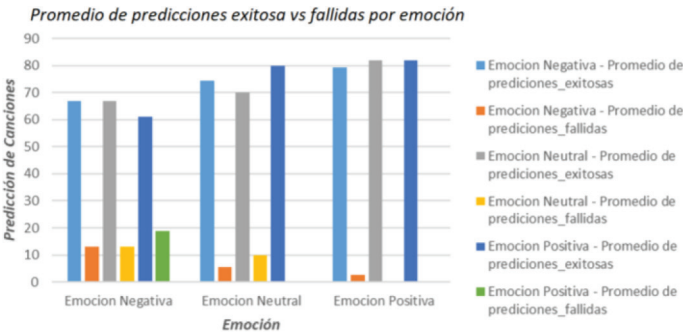


Figura 6 –Promedio de predicciones exitosas vs fallidas por emoción

6.2.Promedio de predicciones exitosas vs fallidas por usuario

Continuando la evaluación del SR se pudo analizar también el promedio de predicciones por usuario. En la Figura 7, se puede observar el análisis de los datos recolectados para cada uno de los cuatro usuarios durante 3 ciclos de ingreso diferentes y la recomendación de contenidos a partir de los tweets publicados en cada una de las sesiones de interacción.

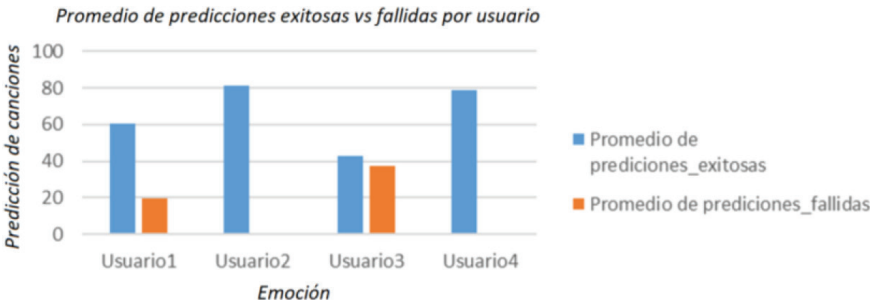


Figura 7 – Promedio de predicciones exitosas vs fallidas por usuario

En la Figura 7 se puede evidenciar como los usuarios dos (2) y cuatro (4) obtuvieron un alto promedio de predicciones exitosas o de acuerdo a la polaridad inferida de los tweets publicados, mientras que para el usuario tres (3) la probabilidad de éxito fue similar a la fallida, por lo que para este usuario la predicción del sistema no fue acertada a partir del análisis de los tweets publicados, lo cual pudo ocurrir porque el usuario no tenía un número suficiente de valoraciones de las canciones para tener como base la predicción.

6.3. Promedio de exitosas vs fallidas general

Una vez analizado la predicción de contenidos exitosos por emoción y usuario, se buscó un indicador que permitiera medir el grado o probabilidad de predicciones exitosas y fallidas totales del sistema, para lo cual se generó la gráfica de la Figura 8, donde se puede observar el porcentaje de predicciones exitosas vs fallidas totales de la plataforma durante las pruebas. Este indicador evidencia que el sistema de recomendación tuvo por cada nueve (9) recomendaciones exitosas una (1) recomendación fallida, una probabilidad de 90% de precisión frente a un 10% de margen de error en cada recomendación realizada al usuario.

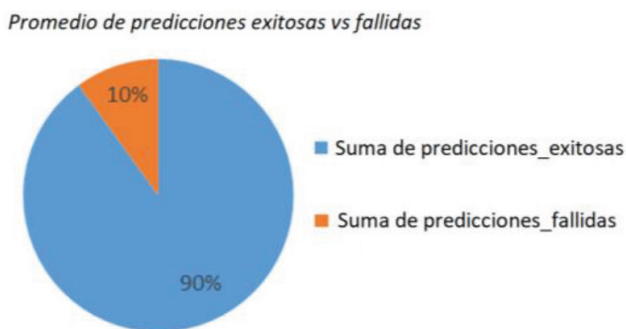


Figura 8 – Promedio de predicciones exitosas vs fallidas por usuario general

7. Conclusiones y trabajos futuros

El dataset de contenidos musicales latinos, constituye un aporte importante para el diseño e implementación de servicios de video basados en computación afectiva, integrando las ventajas del dataset musical de Billboard y la información de video provista por la API de YouTube.

El estudio permitió realizar una investigación exhaustiva sobre librerías para la clasificación y descarga de contenidos musicales latinos, los cuales van a ser el insumo principal del sistema de recomendación a proponer. Además, la vinculación del análisis lírico para la clasificación por emoción permitió acotar y mejorar la clasificación por polaridad todos los contenidos que conforman el dataset.

El estudio del contexto de usuario permitió analizar a más detalle las variables que pueden ayudar a caracterizar el perfil o preferencias del usuario, y así poder determinar los gustos o sentimientos que el usuario puede presentar y de alguna manera poder brindar un tipo de contenido musical de acuerdo a sus gustos.

La plataforma SRMusical permite al usuario escuchar un catálogo de contenidos musicales recomendados de acuerdo a su preferencia y polaridad presente en los tweets registrados en la red social de Twitter bajo la cuenta del usuario.

La evaluación del SR permite determinar la precisión de las predicciones de contenidos musicales latinos, en los resultados se pudo evaluar el porcentaje de precisión exitoso

es mucho superior a la precisión de error; lo cual permite considerar que la clasificación e inferencia de la emoción o polaridad de un tweet permite la retroalimentación para la clasificación de contenidos por parte del SR.

Como trabajo futuro, se espera vincular más variables del contexto para el análisis de emociones, tales como: la voz, la vista, ritmo cardiaco, entre otros. Adicionalmente, se espera evaluar la experiencia de usuario con el SR

Referencias

- Ahlberg, L. (1994). Susanne Langer on representation and emotion in music. *The British Journal of Aesthetics*, 34(1), 69-80.
- Chanchí, G., & Cordoba, A. (2019). Análisis de emociones y sentimientos sobre el discurso de firma del acuerdo de paz en Colombia. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, (E22), 95-107.
- Chanchí, G., Campo, W., & Arciniegas, J. (2018). Arquitectura basada en contexto para el soporte del servicio de VoD. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, (29), 55-71.
- Cunningham, S., Caulder, S., & Grout, V. (2008). Saturday night or fever? Context aware music playlists. In *Proceedings of the 3rd International Audio Mostly conference on Sound in Motion*, (pp.1-8). <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.489.7774&rep=rep1&type=pdf>
- Furht, B., & Agarwal, A. (2013). *Handbook of Medical and Healthcare Technologies*. Springer.
- Hu, X., Downie, J., & Ehmann, A. (2010). Lyric text mining in music mood classification. In *10th International Society for Music Information Retrieval Conference*, (pp. 411-416). <http://citeseerx.ist.psu.edu/viewdoc/download?jsessionid=07F1D93AD8D61977789D777FC2B27D00?doi=10.1.1.205.8487&rep=rep1&type=pdf>
- Laurier, C., Grivolla, J., & Herrera, P. (2008). Multimodal Music Mood Classification Using Audio and Lyrics. In *The Seventh International Conference on Machine Learning and Applications*, (pp. 1-6). <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.182.426&rep=rep1&type=pdf>.
- Lundqvist, L., Carlsson, F., Hilmersson, P., & Juslin, P. (2009). Emotional responses to music: experience, expression, and physiology. *Psychology of Music*, 37(1), 61-90.
- Moreno, M., Segreña, S., López, V., Muñoz, M., & Sánchez, A. (2015). Web mining based framework for solving usual problems in recommender systems. A case study for movies' recommendation. *Neurocomputing*, 176(2), 72-80.
- Park, H., Yoo, J., & Bae, S. (2006). A Context-Aware Music Recommendation System Using Fuzzy Bayesian Networks with Utility Theory. *Fuzzy Systems and Knowledge Discovery*, 4223, 970-979.

- Patil, K., Singh, M., Singh, G., & Sharma, A. (2015). Mental stress evaluation using heart rate variability analysis: a review. *International Journal Of Public Mental Health And Neurosciences*, 2, 10-16.
- Patterson, R., Uppenkamp, S., Johnsrude, I., & Griffiths, T. (2002). The processing of temporal pitch and melody information in auditory cortex. *Neuron*, 36(4), 767-776.
- Perik, E., Ruyter, B., Markopoulos, P., & Eggen, B. (2004). The Sensitivities of User Profile Information in Music Recommender Systems. In *Second Annual Conference on Privacy*, (pp. 137-141). <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=185E88759DC8C677F656A5D919FB2330?doi=10.1.1.84.9880&rep=rep1&type=pdf>
- Rho, S., Han, B., & Hwang, E. (2009). SVR-based music mood classification and context-based music recommendation. In *Proceedings of the 17th ACM International Conference on Multimedia*, (pp. 713-716). <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.661.6746&rep=rep1&type=pdf>
- Sharma, T., & Kapoor, B. (2008). Intelligent data analysis algorithms on biofeedback signals for estimating emotions. In *International Conference on Reliability, Optimization and Information Technology*, (pp. 335-340) . <https://doi.org/10.1109/ICROIT.2014.6798350>
- Westerink, J., van den Broek, E., Schut, M., van Herk, J., & Tuinenbreijer, K. (2008). Computing Emotion Awareness Through Galvanic Skin Response and Facial Electromyography. En *Computing Emotion Awareness Through Galvanic Skin Response and Facial Electromyography*, (pp. 149-162). Springer. https://doi.org/10.1007/978-1-4020-6593-4_14
- Yang, D., & Lee, W. (2009). Music Emotion Identification from Lyrics. In *Proceedings of the 11th IEEE International Symposium on Multimedia*, (pp. 624-629). IEEE. <https://doi.org/10.1109/ISM.2009.123>
- Yang, Y. (2009). *Personalized Redirection of Communication and Data* (pp. 902-915). IGI Global. <https://doi.org/10.4018/978-1-60566-046-2.ch062>

Creación colaborativa de una arquitectura de referencia para la implementación de plataformas de servicios de datos

Edisson Estelio Gutiérrez Jiménez¹, Juan Bernardo Quintero¹, Bell Manrique Losada¹

eegutierrez@udem.edu.co; juquintero@udem.edu.co; bmanrique@udem.edu.co

¹ Universidad de Medellín, Carrera 87 No. 30-65, 50026, Medellín, Colombia.

DOI: 10.17013/risti.39.114-130

Resumen: Big Data se refiere a conjuntos de datos cuyo volumen, velocidad y variedad dificultan su captura, gestión y procesamiento mediante tecnologías y herramientas convencionales. Este concepto está generando nuevas necesidades en las organizaciones para permitir la captura, almacenamiento y análisis de datos con estas características y así obtener información relevante para la toma de decisiones. Un reto para las organizaciones es la implementación de una arquitectura que permita cubrir estas necesidades, ya que deben considerar las diferentes tecnologías existentes y deben establecer las políticas para el gobierno de datos que estarán en manos de los usuarios. Una arquitectura de referencia de una plataforma de analítica de datos, que sea capaz de desvincularse de herramientas tecnológicas será una guía que le permitirá a las organizaciones trazar un camino para lograr la gestión de esos datos y así tener herramientas efectivas para la toma de decisiones empresariales.

Palabras-clave: Big data; arquitectura; analítica de datos.

Collaborative creation of a reference architecture for the implementation of data service platforms

Abstract: Big Data refers to data set whose volume, velocity, and variety make it difficult to capture, manage and process using conventional technologies and tools. This concept is generating new needs in organizations to allow the capture, storage, and analysis of data with these characteristics and thus obtain relevant information for decision-making. A challenge for organizations is the implementation of an architecture that covers these needs, since they must consider the different existing technologies and must establish the policies for data governance that will be available to users. A reference architecture of a data analytics platform that is capable of decoupling from technological tools will be a guide that will allow organizations to define a path to achieve the management of these data and thus have effective tools for make decisions in the company.

Keywords: Big data; architecture; data analytics.

1. Introducción

El descubrimiento de conocimiento y la toma de decisiones a partir de datos de gran volumen, variados, de diferente tipo y con un rápido crecimiento es un reto para las empresas en términos de almacenamiento, administración y procesamiento (Bibri, 2019). El concepto de Big Data es un conjunto de datos cuyo volumen, variedad y velocidad dificultan su captura, gestión y procesamiento mediante tecnologías convencionales (Madden, 2012). Este concepto está generando en las organizaciones la necesidad de recolectar, almacenar, analizar y exhibir datos con estas características para obtener información que ayude a la toma de decisiones y a la proyección en el mercado (Oussous, Benjelloun, Ait Lahcen, & Belfkih, 2018). Unido a esto, los avances tecnológicos de la Cuarta Revolución Industrial y su aplicación en las organizaciones, los cuales buscan la automatización y el intercambio de datos, generarán más volumen de datos y es en este punto donde el Big Data cobra relevancia, ayudando a mejorar el desarrollo de productos y a la innovación de los servicios (Wan, Cai, & Zhou, 2015).

Con el fin de gestionar datos con las características propias del Big Data se han desarrollado diferentes herramientas y arquitecturas tecnológicas, lo que ha generado un desafío para los profesionales de este campo (Mazumder, 2016), ya que ellos deben comprender y seleccionar las herramientas para abordar un problema organizacional relacionado con Big Data y así obtener el conocimiento que se puede extraer de los datos.

Dificultades se presentan en las organizaciones al definir una arquitectura para Big Data, una es que las organizaciones no identifican la forma y las herramientas tecnológicas para gestionar y procesar datos con esas características (Jovanovic, Romero, & Abello, 2017), otra es que las plataformas desarrolladas no facilitan el aumento de la capacidad de trabajo sin comprometer el funcionamiento (Oussous et al., 2018), y la última es el desconocimiento del dominio del problema que se desea atender y de los datos que se procesarán (Zhelev & Rozeva, 2017).

Los servicios de datos se refieren a plataformas para el almacenamiento y distribución de datos con características relacionadas al Big Data (Pollock & Dietrich, 2009), los cuales permiten su diagramación para facilitar el análisis de los datos. La dificultad actual se presenta con la implementación de los servicios de datos, debido a las diferentes tecnologías que existen alrededor de estos servicios sin tener pautas que indiquen cómo será su construcción. Esta misma situación la exponen Jovanovic et al. (2017) al mencionar que la variedad de motores de ejecución para la analítica de datos y la complejidad de las transformaciones son un desafío para las arquitecturas de Big Data y los sistemas de inteligencia de negocio de próxima generación.

Es necesario plantear la definición de una arquitectura de referencia de una plataforma de Big Data, que sea capaz de desvincularse de los detalles técnicos (Blazquez & Domenech, 2018), dicha arquitectura será la base para gestionar datos con características relacionadas al Big Data, no debe estar ligada a herramientas técnicas particulares y debe incluir los posibles tipos de usuario que la pueden utilizar (Zaghloul, Ali-Eldin, & Salem, 2015), los componentes tecnológicos serán determinados por las organizaciones al momento de implementar la arquitectura. En este artículo se presenta una aproximación a una arquitectura de referencia para plataformas de Big Data, la cual incluye dos

componentes: los lineamientos que permiten guiar el proceso de implementación y una fase de gobierno de datos, que como lo exponen Khatri & Brown (2010), permitirá tener control sobre los datos que se exponen, definiendo los usuarios que tienen derechos de decisión y se hacen responsables de los activos de datos de la organización.

El resto del artículo se organiza de la siguiente manera: en la primera sección se presenta el contexto inicial de la investigación sustentando el problema planteado y la fundamentación conceptual. En la segunda sección se muestra un análisis del estado del arte mostrando tendencias para solucionar el problema planteado. En la siguiente sección se presenta la arquitectura propuesta para solucionar el problema y se describe en detalle el proceso de implementación y validación. En la sección de hallazgos se muestran los resultados de la experimentación con su respectivo análisis, para finalmente dar las conclusiones y líneas de trabajo futuro.

2. Fundamentación conceptual

En esta sección se describe el marco teórico y referencial que sustenta este trabajo.

2.1. Big data

Big data es un fenómeno caracterizado por un aumento continuo en el volumen, la variedad, la velocidad y la veracidad de los datos que requieren técnicas y tecnologías avanzadas para capturar, almacenar, distribuir, administrar y analizar estos datos (Ebner, Bühnen, & Urbach, 2014). Los cambios rápidos y continuos en el volumen y variedad de los datos requieren de una infraestructura técnica avanzada y de arquitecturas que abarquen estas nuevas características.

2.2. Arquitectura de software

La arquitectura de software se define como la estructura de los componentes de un programa o sistema, sus interrelaciones, y los principios y guías que controlan su diseño y evolución en el tiempo (Kazman, Abowd, Bass, & Webb, 1993). Dentro del desarrollo de software esta área juega un papel importante debido a la continua evolución de sistemas de información y a la creciente evolución de nuevas tecnologías.

Las propiedades particulares del Big Data y el desafío en las organizaciones para gestionar datos con sus características requiere arquitecturas específicas que posean componentes que almacenen, procesen y analicen el volumen y variedad de los datos (Blazquez & Domenech, 2018).

2.3. Arquitecturas para Big data

Con el fin de atender los retos tecnológicos que demanda el Big Data se han definido arquitecturas para atender sus características particulares:

- Arquitecturas intensivas de datos, se componen generalmente de múltiples dispositivos computacionales que pueden resumirse en términos de trabajos, adquiriendo datos de una o más fuentes y produciendo datos en uno o más

sumideros. Las fuentes y sumideros de datos pueden ser de varios tipos. Los trabajos dentro de esta arquitectura pueden funcionar por lotes, secuencia o interactivo (Artac et al., 2018).

- Arquitecturas Lambda y Kappa. La arquitectura Lambda combina el procesamiento de datos por lotes y en tiempo real, se enfoca principalmente en la ingestión de datos y presenta dificultades en la implementación y el soporte, debido a que mantener estos procesamientos es complejo; la arquitectura Kappa simplifica la arquitectura Lambda combinando el procesamiento por lotes y en tiempo real en una sola capa, llamada capa de procesamiento, la segunda capa que posee es la de publicación, que se utiliza para consultar resultados (Zhelev & Rozeva, 2017).
- Arquitectura basada en modelos, fue desarrollada para agilizar los procesos de analítica desde la preparación de los datos hasta la visualización. Esta última etapa logra ser más concreta por medio de la automatización de los objetivos que el usuario desea visualizar (Golfarelli & Rizzi, 2019).
- Arquitectura de procesamiento de Big Data, consiste en cuatro etapas: recolección o recopilación, carga, transformación y extracción de datos (Krishnan, 2013).

El potencial del Big Data reside en permitir que las organizaciones aprovechen todos sus datos para comprender, supervisar y planificar eficazmente sus procesos (Bibri, 2019), una de las propuestas para lograr este objetivo es alinear el Big Data y la computación en la nube, debido a las oportunidades que brinda la combinación de estos dos componentes para realizar inteligencia de negocios y análisis de datos (Dabbéchi, Nabli, & Bouzguenda, 2016).

2.4. Inteligencia de negocios

La inteligencia de negocios es un conjunto de aplicaciones, tecnologías y procesos para recopilar, almacenar, acceder y analizar datos, para ayudar a los usuarios empresariales a tomar decisiones (Schlesinger & Rahman, 2016). Para lograrlo, es necesario que los usuarios cuenten con herramientas que les faciliten el autoservicio, para que ellos puedan manipular los datos y obtener respuestas a sus preguntas, sin hacer requerimientos a las áreas de informática de las compañías.

El autoservicio se está convirtiendo en la norma en datos y análisis, cada vez más usuarios empresariales exigen acceso a los datos para obtener sus propios conocimientos e impulsar iniciativas (Clarke, Tyrrell, & Nagle, 2016).

2.5. Datos abiertos

Un aspecto por considerar dentro de la arquitectura de plataformas de servicios de datos es el de datos abiertos. Un dato o contenido está abierto si alguien es libre de usarlo, reutilizarlo y compartirlo, sujeto solo al requisito de hacer énfasis en la fuente original. Aplicado a los datos, requiere que un conjunto de datos sea accesible sin costo y sin restricciones técnicas que eviten su uso (Gray & Darbishire, 2011).

2.6. DataOps

Con el propósito de obtener calidad y reducir los tiempos en los ciclos de análisis de datos en la inteligencia de negocios se define DataOps. Esta metodología es un conjunto

de prácticas, procesos y tecnologías que se ajustan al flujo continuo de trabajo de la inteligencia de negocios para reaccionar a requisitos imprevistos de una solución, y así permitir el despliegue e integración continua (Ereth, 2018).

3. Revisión de literatura /Antecedentes

A continuación, se presenta la síntesis de los estudios previos priorizados, los cuales están relacionados con el objeto de estudio y sus fundamentos teóricos ayudan a dar sustento a este trabajo. Se organizan en tres categorías: arquitecturas en un contexto específico, combinación de arquitecturas y nuevas arquitecturas.

3.1. Aplicación de arquitecturas en un contexto específico

En el campo de la atención médica, Wang, Kung, & Byrd (2018) desarrollaron una arquitectura de Big Data, la cual se construyó con base en las experiencias de implementación de sistemas de Big Data en la industria, y se compuso de cinco capas: (i) capa de datos, que incluye las fuentes de datos para apoyar las operaciones y la resolución de problemas; (ii), capa de agregación de datos, que se encarga de adquirir, transformar y almacenar datos; (iii) capa analítica, que se encarga de procesar y analizar datos; (iv) capa de exploración, que funciona generando resultados para el apoyo a la decisión clínica. Por último, la capa de gobierno de datos, que se encarga de administrar los datos a lo largo de todo su ciclo de vida. Esta capa es necesaria, dada la sensibilidad de los datos clínicos. Esta arquitectura solo considera los usuarios con experiencia en analítica, dejando de lado aquellos que pueden requerir un mayor procesamiento de datos a la hora de la visualización. Por otra parte, solo son tenidas en cuenta las fuentes de datos estructuradas, quedando por fuera aquellas fuentes no estructuradas que son comunes en la gestión de las organizaciones.

La arquitectura de Big Data en tiempo real, establecida e implementada por Goncalves, Portela, Santos, & Rua (2017) en la unidad de cuidados intensivos del Centro Hospitalario de Porto, en Portugal, responde a las necesidades de la organización y ayuda a tomar decisiones rápidas en servicios críticos del Centro Hospitalario. La arquitectura implementada trabaja con agentes que permiten la ejecución de tareas automáticas como la recopilación de datos y la actualización de modelos predictivos. La solución presentada esta basada en productos de código abierto y deja de lado una gran variedad de herramientas que no tienen esta característica y que pueden ser útiles en otros escenarios. La arquitectura solo puede ser implementada en unidades de cuidados intensivos que tengan especificaciones similares. Una arquitectura de referencia debe ser útil para empresas de diferentes campos y no generar dependencia por el tipo de herramientas tecnológicas que se utilizan.

La arquitectura de Big Data propuesta por Blazquez & Domenech (2018) explica las particularidades de los análisis de comportamiento económico y social en la era digital. La primera particularidad está relacionada con la variedad de fuentes que podrían proporcionar información sobre estos temas, en este punto proponen una taxonomía para clasificar las fuentes de acuerdo con el propósito del generador de datos. La segunda contribución está relacionada con los métodos para procesar los datos y permitir la gestión en una arquitectura robusta y flexible.

Una dificultad es su implementación en un entorno de computación en la nube, y este aspecto es un obstáculo para implementar una plataforma de servicio de datos, que es el objetivo principal en esta investigación.

3.2. Propuestas de combinación de arquitecturas

Zhelev & Rozeva (2017) exponen que es necesario contar con un amplio conocimiento para escoger la arquitectura de datos cuando se trata de temas de Big Data y profundiza en algunas de ellas, enfocándose en las arquitecturas Lambda y Kappa. Ambas presentan dificultades en la implementación y el soporte, debido a que es complejo mantener el procesamiento por lotes y en tiempo real. En la arquitectura de referencia, la organización será la responsable de definir el tipo de procesamiento que se debe realizar sobre los datos, según las necesidades.

Camargo, Camargo, & Joyanes (2015) revisan tres propuestas de arquitectura de Big Data y proponen una que reúne características de cada modelo. La arquitectura propuesta contiene las siguientes etapas: recolección de datos, carga de datos, transformación de datos, extracción de datos y un aspecto de seguridad. En la arquitectura planteada hace falta una etapa para el gobierno de datos, que no solo considere seguridad, sino aspectos como calidad del dato, datos maestros y metadatos, como lo sugieren Khatri & Brown (2010).

Un estudio realizado por Oussous et al. (2018), analiza tecnologías desarrolladas para Big Data, el objetivo es ayudar a seleccionar la combinación correcta de herramientas, de acuerdo con las necesidades tecnológicas y a los requisitos de la organización, proporcionando una visión detallada de la arquitectura. Un resultado del estudio es que existen deficiencias en las herramientas tecnológicas, en la mayoría de los casos relacionadas con la arquitectura y técnicas adoptadas. Atender esta problemática es un objetivo del presente trabajo, a través de una arquitectura de referencia que defina las bases para que una organización pueda establecer su arquitectura de Big Data.

3.3. Propuestas de nuevas arquitecturas

Passlick, Lebek, & Breitner (2017) describen un proceso llamado inteligencia del auto servicio para hacer inteligencia de negocio con Big Data, el objetivo es que los usuarios sean quienes hagan sus informes y los analicen. Se resaltan dos componentes dentro del diseño planteado, las salas de colaboración y una base de datos de conocimiento para el autoaprendizaje, ya que hacen que el conocimiento implícito de los usuarios sea utilizable. La base de datos de conocimiento utiliza un algoritmo de autoaprendizaje, el cual se ve afectado si no se le suministra un gran volumen de datos, en este caso se aplicará una base de datos de conocimiento simple sin el algoritmo. Una dificultad que se presenta con esta arquitectura es el traslado de los componentes a la nube. Además, dicha arquitectura no ha sido probada en un ambiente empresarial.

Pääkkönen & Pakkala (2015) describen una arquitectura de referencia para sistemas de Big Data basada en el análisis de algunos casos de implementación en la industria. En este trabajo se describen las siguientes funcionalidades: fuentes de datos, extracción de datos, carga y preprocesamiento de datos, análisis de datos, transformación de datos, interfaz y visualización. La arquitectura descrita en este trabajo no ha sido evaluada en

un caso real con grandes volúmenes de datos y no considera una capa de gobierno de datos para gestionar la seguridad y los metadatos.

Assunção, Calheiros, Bianchi, Netto, & Buyya (2015) identificaron los componentes que deben estar presentes en cualquier arquitectura de Big Data: fuentes de datos, procesamiento de datos, modelado y análisis de resultados y visualización. Esta arquitectura se propuso para ser implementada en la nube, debido a los beneficios que ofrece este componente para almacenar grandes cantidades de datos. En el trabajo futuro se deja descrito el establecimiento de lineamientos para definir la arquitectura en una organización, los cuales serán definidos en esta investigación.

3.4. Síntesis de estudios primarios

A continuación, se presenta una tabla con la síntesis de los estudios primarios y las fases de una arquitectura de Big data que cada investigación ha contemplado.

Investigación \ Capas	Fuente de datos	Carga de datos	Transformación y almacenamiento	Analítica	Visualización	Gobierno de datos
(Wang et al., 2018)	X		X	X	X	X
(Goncalves et al., 2017)	X		X	X	X	
(Blazquez & Domenech, 2018)	X	X	X	X	X	X
(Zhelev & Rozeva, 2017)	X	X	X		X	
(Camargo et al., 2015)	X	X	X	X	X	
(Oussous et al., 2018)	X		X	X	X	
(Passlick et al., 2017)	X	X	X	X	X	X
(Pääkkönen & Pakkala, 2015)	X	X	X	X	X	
(Assunção et al., 2015)	X		X	X	X	

Tabla 1 – Síntesis de trabajos analizados

4. Arquitectura de referencia

La co-creación de la arquitectura de referencia tendrá en cuenta diferentes usuarios del área de Big Data, con los cuales se tendrán sesiones de trabajo para conocer las arquitecturas que actualmente están manejando en sus organizaciones y luego plasmar una arquitectura ideal de Big Data que tenga en cuenta las capas presentadas en el numeral anterior. La elección de una técnica para abordar las sesiones de trabajo, que permita obtener nuevos requisitos y validar las características de la arquitectura de referencia

es el primer paso para lograr un acercamiento con los potenciales usuarios, que a futuro podrán hacer uso de la arquitectura para una plataforma de servicio de datos.

El estudio determinó que la combinación de grupos focales con la creación de prototipos facilita la interacción y participación de los usuarios, de forma que ellos puedan expresar abiertamente sus opiniones y proponer nuevas ideas que ayuden a la creación de la arquitectura de referencia. Un grupo focal es un medio para obtener ideas sobre un producto o servicio en un entorno grupal interactivo. La creación de prototipos se utiliza para validar las necesidades de las partes interesadas a través de un proceso que crea un modelo o diseño de requisitos.

Con el propósito de lograr el desarrollo de la arquitectura de referencia se ha tomado la metodología *Design Science in Information Systems Research* (Hevner, March, Park, & Ram, 2008), la cual tiene como principio que el conocimiento y comprensión de un problema y su solución, se adquieren en la aplicación y construcción de un artefacto. Este principio será aplicado en el diseño de la arquitectura de referencia y el diseño inicial se va refinando en cada avance del proceso de investigación. Las fases de esta metodología son: fase I – Análisis del entorno, fase II – Análisis de la base del conocimiento y fase III – Diseño y validación del modelo.

La fase I – Análisis del entorno, ha sido cubierta con la revisión bibliográfica presentada en el numeral anterior; la fase II – Análisis de la base del conocimiento, contempla las actividades: desarrollo de grupos focales y definición de las capas de la arquitectura, estas actividades se abarcan en los numerales 4.1 y 4.2. Por último, la fase III – Diseño y validación del modelo, que contempla el diseño y evaluación de la arquitectura se desarrolla en el numeral 4.3 y en la evaluación y conclusiones del artículo.

4.1. Bases y fundamentos de la arquitectura

Basados en las investigaciones analizadas se abstraen las siguientes capas para una arquitectura de referencia para plataformas de servicios de datos: fuentes de datos, carga de datos, transformación y almacenamiento, analítica, visualización y gobierno de datos. En la capa de visualización se utilizará el portal o plataforma de servicios de datos para facilitar el auto servicio y el análisis de los usuarios. La capa de gobierno de datos será transversal a todas las demás y permitirá definir quién se hace responsable de la toma de decisiones de la organización y quién debe responder por los datos, esta capa tendrá en cuenta la seguridad, los datos maestros y la metadata.

4.2. Aproximación a la arquitectura de referencia

El diseño de la arquitectura de referencia para plataformas de Big Data fue guiado a partir de un proceso colaborativo con 3 grupos focales. Se realizó la identificación previa de los participantes para garantizar su conocimiento en Big Data y analítica y se dividieron en grupos de trabajo con características similares, según el tipo de industria de la que hacen parte.

Los participantes de cada uno de los grupos focales fueron los siguientes:

- Cuatro empleados de empresas de tecnología con conocimiento en Big Data y analítica.

- 11 personas con conocimiento y experiencia en entornos de Big Data y analítica de Instituciones de Educación Superior de la ciudad de Medellín.
- Dos arquitectos de Big Data de empresas de servicios.

En el desarrollo de cada uno de los grupos focales se ejecutaron las siguientes actividades:

- Presentación de alternativas para implementar arquitecturas de Big Data.
- Los asistentes presentan la arquitectura y herramientas tecnológicas de las plataformas de Big Data que tienen en las empresas.
- Diseño de modelo de arquitectura para plataformas de servicios de datos, teniendo en cuenta 4 capas: repositorios, preparación de datos, modelado y evaluación y visualización.
- Análisis de los modelos de arquitectura construidos e identificación de características comunes.
- Calificación de las características comunes encontradas, para priorizar necesidades.

La identificación de características comunes y la calificación asignada por cada uno de los asistentes a los grupos focales arrojó el resultado que se plasma en la tabla 2.

Criterio \ Calificación	Alto (5)	Medio – Alto (4)	Medio (3)	Medio – Bajo (2)	Bajo (1)	Ponderación
Reportes Self-Services	10	1	0	0	0	54
Servicios de “Datos Abiertos”	10	1	0	0	0	54
Necesidad de gobierno de datos	10	1	0	0	0	54
Factibilidad de software libre	5	3	1	0	0	40
Necesidad de un cluster (Data Lake)	0	5	6	0	0	38
Preparación Self-Services	2	3	3	3	0	37
Necesidad de data no estructurada	0	2	6	2	0	30
Necesidad de replicación	0	1	2	1	7	19

Tabla 2 – Calificación de características para una plataforma de servicio de datos

Como se puede observar en la tabla anterior, el auto servicio, los datos abiertos y el gobierno de datos obtuvieron la mayor ponderación, evidenciando que son características con las cuales debe contar una arquitectura de Big Data.

Con el grupo de arquitectos se hizo énfasis en las capas que debe tener una arquitectura de Big Data y sus características principales, como resultado se construyó la arquitectura que se muestra en la figura 1.

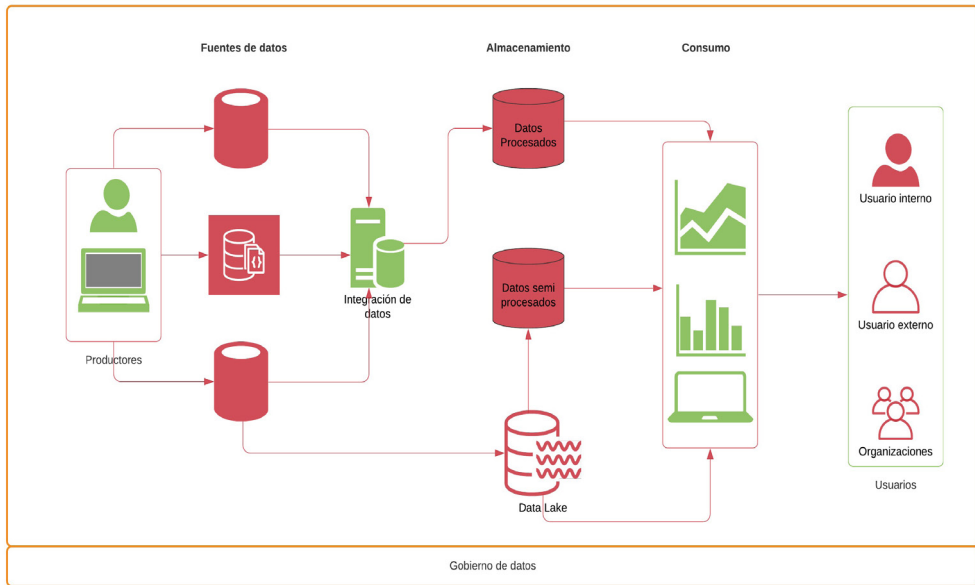


Figura 1 – Arquitectura propuesta por arquitectos de Big data

Las principales conclusiones de las sesiones de trabajo que se deben tener en cuenta en una arquitectura de referencia de una plataforma de servicios de datos son:

- Es indispensable tener gobierno de datos antes de definir la arquitectura para la plataforma de servicios de datos, con el fin de iniciar el control sobre la información que puede gestionar cada usuario.
- La arquitectura de referencia debe ser escalable y flexible para permitir la auto gestión de los usuarios finales.
- Se deben definir los metadatos, como parte del gobierno, para identificar y estandarizar los nombres de cada uno de los campos que pueden tratar y visualizar los usuarios.

4.3. Representación gráfica de la arquitectura

Con la revisión de literatura y los hallazgos de los grupos focales se propone la arquitectura de referencia para plataformas de servicios de datos, la cual se muestra en la figura 2.

La arquitectura de referencia que se propone para plataformas de servicios de datos tiene en cuenta las siguientes capas: fuentes de datos, integración o procesamiento, almacenamiento, analítica, visualización y gobierno de datos.

Fuentes de datos

El objetivo de esta capa es leer los datos proporcionados por diversos canales, con diversos tamaños y formatos; en esta capa se presenta el primer obstáculo en la implementación

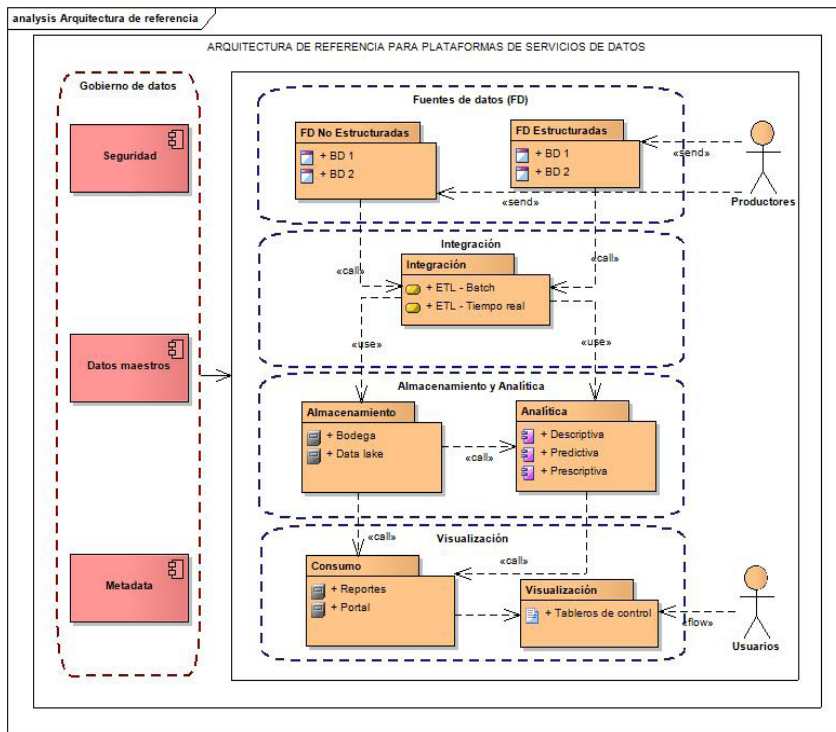


Figura 2 – Arquitectura de referencia para plataformas de servicios de datos

de una arquitectura de Big Data, ya que las características de los datos pueden variar considerablemente y la adquisición de componentes para almacenarlos puede exceder el presupuesto (Wang et al., 2018). Ante esta situación cada compañía debe decidir la cantidad y el tipo de datos a los que les dará tratamiento.

Integración o procesamiento

En esta capa se ejecuta la tarea de preparación de los datos, normalmente esta actividad es la más laboriosa y la que regularmente consume más tiempo (Assunção et al., 2015). En esta capa se fusionan los datos en una estructura homogénea, lo que permite que los datos sean rastreables, más fáciles de acceder y manipular (Blazquez & Domenech, 2018).

Las arquitecturas Lambda y Kappa pueden ser tenidas en cuenta en esta capa, dado que ambas permiten el procesamiento por lotes y en tiempo real. Elegir la arquitectura correcta depende de los requisitos de la solución a implementar, si los algoritmos que procesan datos en tiempo real e históricos son los mismos y la solución no necesita cálculos pesados, entonces se podría optar por la arquitectura Kappa. Existen casos en los que no se puede simplificar el procesamiento de datos y sería necesario usar la arquitectura Lambda.

Esta es una de capas en las cuales el gobierno de datos debe incorporar restricciones, ya que los datos integrados pueden facilitar el descubrimiento de información privada y personal que de otro modo sería anónima.

Almacenamiento

En esta capa se almacenan los datos para luego permitir su recuperación y se deben producir los metadatos asociados (Blazquez & Domenech, 2018). Actualmente, existen opciones en la nube que permiten realizar esta tarea, pero deben ser analizadas con detalle, ya que pueden no satisfacer las necesidades de concurrencia y rendimiento de las herramientas utilizadas en el análisis (Assunção et al., 2015).

Una de las actividades a analizar es el costo computacional de transportar los datos del sitio de procesamiento al lugar de almacenamiento, dichos costos deben ayudar a determinar si el lugar de procesamiento debe ser más cercano al de almacenamiento. El almacenamiento de los datos se puede realizar en un lago o en una bodega de datos, esto está determinado por los requisitos de la solución.

Analítica

Esta capa permite el análisis de datos y la obtención e interpretación de resultados. Los métodos estadísticos y de aprendizaje automático deben ser utilizados para extraer conocimiento y hacer predicciones a partir de los datos procesados y almacenados en las capas anteriores. Para lograr este objetivo, se aplican técnicas descriptivas, predictivas y prescriptivas. El análisis descriptivo ayuda a proporcionar ideas sobre las características y la evolución de las diferentes variables de una organización, estos resultados se utilizarán en la capa de visualización para crear tablas y gráficos que representen la relación entre las variables (Blazquez & Domenech, 2018).

El análisis predictivo se basa en modelos que ayudan a explicar, clasificar, y pronosticar la actividad futura de la organización, el comportamiento y las tendencias. Para hacerlo, los modelos se entrenan usando métodos de aprendizaje automático para seleccionar las variables más significativas e ir mejorando las predicciones (Blazquez & Domenech, 2018).

El análisis prescriptivo ayuda a la toma de decisiones determinando acciones y evaluando su impacto con respecto a los objetivos, requisitos y limitaciones del negocio.

Existen diversas formas de realizar este análisis: procesamiento por lotes o en paralelo y procesamiento en tiempo real. La elección de cual técnica escoger estará basada en el tipo de análisis que se debe realizar y esto determinará la herramienta tecnológica que se debe usar (Wang et al., 2018).

En esta capa se deben definir los tipos de usuarios que realizarán el análisis de los datos para determinar qué tipos de herramientas tecnológicas se disponen para dicho fin y el grado de procesamiento que se debe realizar en los datos, de esta manera se podrán facilitar las tareas de autogestión de los diferentes grupos de usuarios (Passlick et al., 2017).

Visualización

En esta capa se generan los informes y se permite el monitoreo de los datos procesados. La presentación de informes es una característica crítica del análisis de Big Data que

permite visualizar los datos de una manera útil para respaldar las operaciones diarias y ayudar a la toma de decisiones en las compañías (Wang et al., 2018).

La presentación de los datos se debe dividir según el tipo de usuario que los visualizará. La separación se realiza de acuerdo con la habilidad y la necesidad del usuario (Passlick et al., 2017). En los tableros de control, los usuarios son consumidores de informes predefinidos y allí dan seguimiento a las métricas de desempeño de la organización. Los tableros de control son utilizados, generalmente, por usuarios ocasionales. Otro grupo de usuarios son los científicos de datos, ellos tienen un alto grado de libertad, así como los derechos de acceso y las herramientas para construir sus propios informes. Entre las dos opciones mencionadas anteriormente se encuentra el portal de análisis, allí los informes están predefinidos, pero los usuarios pueden realizar ajustes con restricciones. Las definiciones de grupos de usuarios deben adaptarse a las necesidades de cada organización.

En esta capa aparece la plataforma de servicios de datos para permitir la distribución y diagramación y así facilitar el análisis y la toma de decisiones.

Durante esta etapa el gobierno de datos debe establecer los derechos de autor de los datos e informes visualizados, citando fuentes y dando controles de acceso (Blazquez & Domenech, 2018).

Gobierno de datos

Esta capa es transversal a las demás y allí se deben aplicar las políticas y regulaciones de la organización a todo el ciclo de vida de los datos: desde la ingesta de datos hasta la eliminación (Blazquez & Domenech, 2018). Otros componentes de esta capa son la gestión de datos maestros, la gestión de la seguridad y los metadatos. Esta capa enfatiza en el “cómo” aprovechar los datos en la organización. La gestión de datos maestros está definida como los procesos y estándares para administrar los datos. El componente de seguridad es la plataforma que proporciona actividades para facilitar la evaluación de la configuración, el monitoreo, auditoría y protección de los datos (Wang et al., 2018). Los metadatos permiten identificar y estandarizar los nombres de cada uno de los campos que pueden tratar y visualizar los usuarios.

La capa de gobierno debe tener en cuenta los siguientes módulos: módulo de ingestión, se ocupa de la gestión de las fuentes de datos, los permisos de usuario y la integridad de los metadatos; módulo de procesamiento, realiza un seguimiento de las transformaciones y de los permisos para acceder a los datos; módulo de resultados, se ocupa de los permisos para acceder a los informes y los resultados; módulo de auditoría, inspecciona que la implementación de la arquitectura sea coherente con las políticas de seguridad (Blazquez & Domenech, 2018).

5. Evaluación de la arquitectura

La evaluación de la arquitectura se realiza con base en aspectos planteados en la introducción. Primero, las organizaciones no identifican la forma y las herramientas tecnológicas para gestionar y procesar grandes volúmenes de datos (Jovanovic et al., 2017); frente a este punto, la arquitectura de referencia brinda una guía para que cada

organización adopte el camino a seguir para establecer su arquitectura, según las fases que desee aplicar y las herramientas tecnológicas con la que cuente o desee adquirir.

Segundo, las plataformas desarrolladas no facilitan el aumento de la capacidad de trabajo sin comprometer el funcionamiento (Oussous et al., 2018) y se tiene desconocimiento del dominio del problema que se desea atender y de los datos que se procesarán (Zhelev & Rozeva, 2017); ante estas situaciones, identificar las fuentes de datos que serán tratadas es el primer paso para definir todo el proceso de extracción, transformación y análisis de datos y así, no comprometer a futuro el funcionamiento de la arquitectura. La definición de las fuentes de datos debe incluir la cantidad y los tipos de datos a los que se les dará tratamiento, como primer paso para tener claridad del problema que se desea atender.

Con relación al desconocimiento del dominio del problema, es necesario tener entendimiento de la organización, claridad en sus objetivos y comprensión de los datos que se procesarán. El propósito es convertir esos objetivos en objetivos técnicos dentro de la arquitectura y así establecer las capas y procesos que se deben cubrir, así se podrán recolectar los datos correctos y después interpretar los resultados.

Por último, una arquitectura de referencia de una plataforma de Big Data no debe estar ligada a herramientas técnicas particulares y debe incluir los posibles tipos de usuario que la pueden utilizar (Zaghloul et al., 2015); frente a este aspecto, la arquitectura de referencia pretende ser lo suficientemente general para implementarse con diferentes tecnologías, paradigmas informáticos y software analítico, dependiendo de los requisitos y propósitos de cada caso en particular.

6. Conclusiones

El gobierno de datos y la autogestión son dos características que las organizaciones deben incluir en una arquitectura de servicios de datos, con el fin de facilitar el uso de estas plataformas y establecer políticas que regulen el ciclo de vida de los datos.

Es necesario identificar los tipos de usuario que tendrá la plataforma de servicios de datos, ya que ellos serán una de las guías para la construcción de la arquitectura.

La arquitectura de referencia para plataformas de servicios de datos pretende ser lo suficientemente general como para implementarse con diferentes tecnologías, paradigmas informáticos y software analítico, dependiendo de los requisitos y propósitos de cada organización.

Frente a los hallazgos de la revisión del estado del arte, la propuesta de arquitectura de referencia tiene las siguientes diferencias: considera diferentes tipos de usuario y de fuentes de datos, no genera dependencia por el tipo de herramientas tecnológicas que se utilizan, las capas pueden ser implementadas en la nube y se establece una capa para el gobierno de datos.

Dentro de los componentes de la arquitectura establecida, uno de los retos pendientes es la gestión de los flujos intensivos de datos, y requiere una atención especial por parte del mundo académico y de la industria.

La arquitectura de referencia diseñada es la principal contribución de este trabajo, quedando como trabajo futuro la implementación y respectivas pruebas en algún tipo de industria y dentro de un contexto escalable.

Aunque la arquitectura propuesta es lo suficientemente general como para implementarse con cualquier tecnología, su adaptación no está exenta de obstáculos. Para mencionar alguno de ellos, la integración de la arquitectura con los sistemas de información de la organización se convierte en un proceso crítico, el cual debe asegurar la generación de pronósticos y predicciones sin problemas.

Referencias

- Artac, M., Borovsak, T., Di Nitto, E., Guerriero, M., Perez-Palacin, D., & Tamburri, D. A. (2018). Infrastructure-as-Code for Data-Intensive Architectures: A Model-Driven Development Approach. In *Proceedings of the 2018 IEEE 15th International Conference of Software Architecture ICSA* (pp. 156–165). <https://doi.ieeecomputersociety.org/10.1109/ICSA.2018.00025>
- Assunção, M. D., Calheiros, R. N., Bianchi, S., Netto, M. A., & Buyya, R. (2015). Big Data computing and clouds: Trends and future directions. *Journal of Parallel and Distributed Computing*, 79, 3–15. <https://doi.org/10.1016/j.jpdc.2014.08.003>
- Bibri, S.E. (2019). The anatomy of the data-driven smart sustainable city: instrumentation, datafication, computerization and related applications. *Journal of Big Data*, 6(59), 1–43. <https://doi.org/10.1186/s40537-019-0221-4>
- Blazquez, D., & Domenech, J. (2018). Big Data sources and methods for social and economic analyses. *Technological Forecasting and Social Change*, 130, 99–113.
- Camargo Vega, J. J., Camargo Ortega, J. F., & Joyanes Aguilar, L. (2015). Arquitectura Tecnológica Para Big Data. *Revista Científica*, 21(1), 1–7. <https://doi.org/10.14483/udistrital.jour.RC.2015.21.a1>
- Clarke, P., Tyrrell, G., & Nagle, T. (2016). Governing self service analytics. *Journal of Decision Systems*, 25, 145–159.
- Dabbéchi, H., Nabli, A., & Bouzguenda, L. (2016). Towards cloud-based data warehouse as a service for big data analytics. In Nguyen N., Iliadis L., Manolopoulos Y., Trawiński B. (eds). *Computational Collective Intelligence. ICCCI 2016. Lecture Notes in Computer Science* (vol 9876, pp. 180 – 189). Springer. https://doi.org/10.1007/978-3-319-45246-3_17
- Ebner, K., Bühnen, T., & Urbach, N. (2014). Think big with big data: Identifying suitable big data strategies in corporate environments. In *Proceedings of the 47th Hawaii International Conference on System Sciences* (pp. 3748–3757). <https://doi.ieeecomputersociety.org/10.1109/HICSS.2014.466>
- Ereth, J. (2018). DataOps - Towards a Definition. *Published in LWDA*.
- Golfarelli, M., & Rizzi, S. (2019). A model-driven approach to automate data visualization in big data analytics. *Information Visualization*, 19, 24–47. <https://doi.org/10.1177/1473871619858933>

- Goncalves, A., Portela, F., Santos, M. F., & Rua, F. (2017). Towards of a Real-time Big Data Architecture to Intensive Care. *Procedia Computer Science*, 113, 585–590. <https://doi.org/10.1016/j.procs.2017.08.294>
- Gray, J. & Darbishire, H. (2011). Beyond Access: Open Government Data & the Right to (Re)use Public Information. In *Access Info Europe and Open Knowledge* (pp. 1 - 89). https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2586400
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2008). Design science in information systems research. *Management Information Systems Quarterly*, 28, 75 - 105.
- Jovanovic, P., Romero, O., & Abelló, A. (2017). A unified view of data-intensive flows in business intelligence systems: A survey. In: A. Hameurlain, J. Küng, R. Wagner (Eds.), *Transactions on Large-Scale Data- and Knowledge-Centered Systems XXIX. Lecture Notes in Computer Science* (vol 10120, pp. 66-107). http://dx.doi.org/10.1007/978-3-662-54037-4_3
- Kazman, R., Abowd, G., Bass, L., & Webb M. (1993). Analyzing the Properties of User Interface Software Architectures. *Computer Science Technical Report Carnegie-Mellon University*
- Khatri, V., & Brown, C.V. (2010). Designing data governance. *Commun. ACM*, 53, 148-152.
- Krishnan, K. (2013). *Data Warehousing in the Age of Big Data*. Newnes.
- Madden, S. (2012). From databases to big data. *IEEE Internet Computing*, 16, 4-6. <https://doi: 10.1109/MIC.2012.50>
- Mazumder S. (2016). Big Data Tools and Platforms. In S. Yu & S. Guo (Eds.). *Big Data Concepts, Theories and Applications*, (pp. 29 - 128). Springer. https://doi.org/10.1007/978-3-319-27763-9_2
- Oussous, A., Benjelloun, F.Z., Ait Lahcen, A., & Belfkih, S. (2018). Big Data technologies: A survey. *Journal of King Saud University – Computer and Information Sciences*, 30, 431-448.
- Pääkkönen, P., & Pakkala, D. (2015). Reference Architecture and Classification of Technologies , Products and Services for Big Data Systems. *Big Data Research*, 2(4), 166–186. <https://doi.org/10.1016/j.bdr.2015.01.001>
- Passlick, J., Lebek, B., & Breitner, M. H. (2017). A Self-Service Supporting Business Intelligence and Big Data Analytics Architecture. In *Proceedings der 13 Internationalen Tagung Wirtschaftsinformatik* (pp. 1126–1140).
- Pollock, R., & Dietrich, D. (2009). CKAN: apt-get for the Debian of Data. In *Proceedings 26th Chaos Communication Congress*. <https://fahrplan.events.ccc.de/congress/2009/Fahrplan/events/3647.en.html>
- Schlesinger, P., & Rahman, N. (2016). Self-Service Business Intelligence Resulting in Disruptive Technology. *Journal of Computer Information Systems*, 56(1), 11-21. <https://doi: 10.1080/08874417.2015.11645796>

- Wan, J., Cai, H., & Zhou, K. (2015). Industrie 4.0: enabling technologies. *In Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things* (pp.135–140). IEEE. <https://doi.org/10.1109/ICAIOT.2015.7111555>
- Wang, Y., Kung, L., & Byrd, T. A. (2018). Big data analytics: Understanding its capabilities and potential benefits for healthcare organizations. *Technological Forecasting and Social Change*, 126, 3–13.
- Zaghloul, M.M., Ali-Eldin, A., & Salem, M. (2015). A process-centric data analytics architecture. *In Proceedings of the 9th International Conference on INFormatics and Systems* (pp. 34-39). IEEE. <https://doi.org/10.1109/INFOS.2014.7036705>
- Zhelev, S., & Rozeva, A. (2017). Big data processing in the cloud - Challenges and platforms. *In AIP Conference Proceedings*. <https://doi.org/10.1063/1.5014007>

Influencia de la Teoría de Roles en actividades de Gestión: Un experimento controlado con estudiantes de Ingeniería de Software

Raúl Aguilar Vera¹, Omar Gómez Gómez², Julio Díaz Mendoza¹, Juan Ucan Pech¹

avera@correo.uady.mx; ogomez@epoch.edu.ec; julio.diaz@correo.uady.mx;
juan.ucan@correo.uady.mx

¹ Facultad de Matemáticas, Universidad Autónoma de Yucatán, C.P. 97000, Mérida, Yucatán, México.

² Facultad de Informática y Electrónica, Escuela Superior Politécnica de Chimborazo, Riobamba 060155, Ecuador.

DOI: 10.17013/risti.39.131-147

Resumen: El estudio explora la influencia de la Teoría de Belbin en actividades de gestión de proyectos. Para crear el contexto experimental con tareas de planificación y control, se diseñó un escenario ficticio en el que los sujetos experimentales planificaron y controlaron el desarrollo de un proyecto —construcción de fachadas. Los resultados del experimento mostraron que, en cuanto a la estimación de costos y obtención de ganancias, los equipos integrados con base en la Teoría de Belbin (EC) no muestran diferencias con los equipos integrados aleatoriamente (ET); sin embargo, en lo que respecta a la variable “penalización”, los EC difieren significativamente de los ET, al haber entregado productos de mejor calidad.

Palabras-clave: Experimentación en Ingeniería de Software; Gestión de Proyectos Software; Roles de Belbin.

Influence of Role Theory on Management Activities: A Controlled Experiment with Software Engineering Students.

Abstract: The study explores the influence of Belbin’s Theory on project management activities. To create the experimental context with planning and control tasks, a fictitious scenario was designed in which the experimental subjects planned and controlled the development of a project - building a set of different facades. The results of the experiment showed that, in terms of estimating costs and obtaining profits, the integrated teams based on the Belbin Theory (EC) do not show differences with the randomly integrated teams (ET); however, regarding the penalties obtained, however, as the variable “penalty”, the EC presents statistically significant differences with respect to the ET, having obtained better quality products.

Keywords: Belbin Roles; Software Engineering Experimentation; Software Engineering Experimentation; Software Management.

1. Introducción

La Ingeniería de Software (IS), como disciplina ingenieril dispone hoy día de un cuerpo de conocimientos reconocido por profesionistas y académicos afines de la aún joven disciplina, dicho cuerpo de conocimientos integra un conjunto de áreas relacionadas con los procesos de desarrollo y de gestión del software (Bourque & Fairley, 2014); sin embargo, las necesidades de una sociedad en constante cambio, requiere de nuevos métodos, técnicas, herramientas, así como buenas prácticas, que permitan al proceso software, su adaptación.

La Ingeniería de software puede ser definida como: (1) La aplicación de un enfoque sistemático, disciplinado y cuantificable, al desarrollo, operación y mantenimiento de software; esto es, la aplicación de la ingeniería al software; y (2) El estudio de enfoques como los citados en (1). (IEEE-CS, 1990, p. 67).

El estudio de los procesos de desarrollo y gestión en la IS ha sido desarrollado considerando diversas variables, sin embargo, el aspecto social intrínseco de la disciplina (Juristo & Moreno, 2001) ha llevado a considerar al factor humano, de singular interés para su investigación. En un artículo reciente sobre la mejora del proceso software, Morales & Vega (2018) propusieron un catálogo de factores humanos, en el cual se considera como factor crítico para la organización, al compromiso de cada persona de cumplir con las tareas asignadas a su rol, o dicho de otra manera, a la responsabilidad de los roles. En el ámbito del proceso software, es bien sabido que muchas tareas y/o actividades se realizan en el contexto de equipos de desarrollo, en este sentido, Humphrey (2000) comenta que el proceso de formación y creación de un equipo de desarrollo de software no se produce por accidente, es un proceso complejo en el cual los integrantes del equipo deben establecer relaciones de trabajo, ponerse de acuerdo sobre los objetivos del proyecto y determinar las funciones de los miembros del grupo.

El propósito del estudio que se describe en este artículo, consiste en explorar si el uso de la teoría de roles propuesta por Belbin (1993) para la formación de equipos de trabajo, tiene alguna influencia en tareas vinculadas con los procesos de gestión del software, y para ello, se diseñó un escenario ficticio en el cual, alumnos de un programa de Ingeniería de Software, deben ponerse de acuerdo al estimar, planificar y desarrollar una tarea, con base en un conjunto de recursos humanos y materiales identificados para un período de tiempo limitado.

2. Teoría de Roles de Belbin

Entre las investigaciones sobre los roles que desempeñan los integrantes de un equipo, existe hoy día diferenciación entre roles orientados a la actividad individual, es decir, que se encuentran descritos en función de la tarea que pueden desempeñar —dependen de su formación profesional— y roles orientados al desempeño del equipo, es decir, se encuentran definidos en función de su comportamiento con sus pares al interior del equipo, este segundo tipo de rol, es conocido como rol de equipo y al parecer, su ausencia o presencia, tiene una influencia significativa en los logros del equipo (Aritzeta, Swailes, & Senior, 2005).

El trabajo sobre roles de equipo de Belbin (1981, 1993), es sin lugar a dudas el más conocido entre consultores e investigadores; los roles propuestos por Belbin (1993) pueden ser agrupados en tres diferentes categorías, roles orientados a la acción: Impulsor, Implementador y Finalizador; roles orientados a las personas o roles sociales: Coordinador, Investigador de Recursos y Cohesionador; roles orientados a los procesos cognitivos o roles mentales: Cerebro, Monitor Evaluador y Especialista.

Aparte de la clasificación citada en el párrafo anterior, Belbin ofrece algunos mecanismos para la identificación del rol primario que un individuo puede asumir en un equipo de trabajo —en función de su comportamiento; no obstante, su principal aportación —a juicio de los autores— son las relaciones de interacción analizadas entre los roles al interior de un equipo de trabajo, relaciones que consideran la posición que ocupan dentro de la estructura organizacional:

- Jefe → Subordinado (Colaborador),
- Colaborador → Colaborador, y
- Subordinado (Colaborador) → Jefe.

Dicho análisis deriva en un conjunto de recomendaciones que permiten integrar equipos de trabajo con base en roles compatibles.

El interés de los autores radica en las relaciones entre los integrantes de equipos de desarrollo software —equipos de proyecto— es decir, roles con la misma posición y autoridad en la organización —estructura de equipos planos— por tal motivo, en este trabajo se utiliza como referencia el segundo tipo de relaciones.

La Tabla 1 ilustra las relaciones de Colaborador-Colaborador, aunque es importante comentar que los roles que no se citan en las relaciones de compatibilidad, aunque no son preferidos, tampoco presentan problemas de incompatibilidad (Aguilar, 2008).

Tipo	Rol	Roles compatibes	Roles incompatibles
Acción	Impulsor (Sharper: S)	Investigador de Recursos	Coordinador, Cohesionador.
	Implementador (Implementer: I)	Coordinador, Investigador de recursos, Monitor Evaluador, Especialista, Finalizador	Implementador, Cerebro
	Finalizador (Completer-Finisher: CF)	Implementador	Investigador de recursos, Monitor Evaluador
Sociales	Coordinador (Chairman: CH)	Cohesionador, Implementador	Impulsor
	Investigador de Recursos (Resouce Investigator: RI)	Cohesionador, Implementador	Finalizador, Especialista
	Cohesionador (Teamworker: TW)	Cohesionador, Cerebro	Impulsor

Tipo	Rol	Roles compatibes	Roles incompatibles
Mentales	Cerebro (Plant: P)	Coordinador, Investigador de recursos, Cohesionador	Monitor Evaluador, Cerebro, Especialista, Implementador
	Monitor Evaluador (Monitor-Evaluator: ME)	Coordinador, Implementador	Finalizador, Monitor Evaluador, Cerebro
	Especialista (Specialist: SP)	Implementador, Cohesionador	Cerebro, Investigador de recursos

Tabla 1 – Compatibilidad entre Roles de Equipo en estructuras planas (Aguilar, 2008)

3. Trabajos relacionados

Entre los estudios reportados sobre el uso de la teoría de Roles de Belbin en el contexto de la IS, se encuentra el trabajo de Henry & Stevens (1999) quienes exploraron la utilidad de los roles de Belbin en la mejora de la efectividad de los equipos, particularmente, la influencia del liderazgo, dichos autores llevaron a cabo un experimento controlado con estudiantes, y entre sus hallazgos concluyen que la inclusión de un solo líder en el equipo, mejora su rendimiento, en comparación con equipos que tienen varios líderes o que no incluyen dicho rol.

Otro estudio reportado es el de Pollock (2009) en este trabajo se explora la influencia de contar con una diversidad de roles y personalidades, como estrategia para mejorar la efectividad del equipo, el autor comenta que, aunque no encontró evidencia de que la diversidad influya de alguna manera, pudo observar que la presencia de ciertos roles, como son el caso del Impulsor, Coordinador o Finalizador, pueden incrementar la efectividad.

Estrada & Peña (2013) reportan un experimento controlado con estudiantes desempeñando actividades vinculadas con las fases de requisitos, diseño y codificación, las autoras concluyen que algunos roles presentan mayor aportación en ciertas actividades, resaltando al rol Implementador en la tarea de codificación.

El Cuerpo Académico de Ingeniería de Software para la Educación, con la colaboración de investigadores de otras instituciones, ha realizado experimentos controlados con estudiantes en diversas tareas y/o actividades vinculadas con el proceso software. Un primer estudio reportado en (Aguileta, Ucán, & Aguilar, 2017) se compara la calidad de la legibilidad del código generado por equipos integrados con roles compatibles —con base en la Teoría de Belbin— con equipos formados con integrantes asignados aleatoriamente, los resultados obtenidos permitieron concluir que los equipos que usaron como referencia la Teoría de Belbin, presentan resultados significativamente mejores que aquellos integrados mediante un criterio de aleatoriedad. En un segundo estudio reportado en (Aguilar, Díaz, & Ucán, 2019) vinculado con la tarea de medición software, se encontraron resultados contradictorios entre un primer experimento y su réplica experimental, en dicho estudio no se pudo concluir la existencia de diferencias

significativas entre las variables vinculadas, ni con el producto (número de puntos de función), ni con el proceso (tiempo utilizado). En otro experimento reportado en (Aguilar, Muñoz, Díaz, & Ucán, 2020) se exploran las diferencias entre la calidad de los productos generados por los equipos, en el proceso de desarrollo de requisitos, el experimento incorporó la evaluación de los productos por parte de dos expertos en requisitos; los resultados —obtenidos en ambas evaluaciones— mostraron diferencias significativas entre los tratamientos, obteniendo resultados significativamente mejores en cuanto a la calidad del documento de especificación de requisitos, los equipo integrados con roles compatibles.

4. Gestión de Proyectos Software

El proceso software integra tareas vinculadas tanto con el desarrollo de software, como con la gestión de los proyectos en los que dicho software es desarrollado (Bourque & Fairley, 2014). La gestión de proyectos es definida como la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades de un proyecto, para satisfacer los requisitos del mismo (PMI, 2013); en el caso de proyectos software, la Planificación y el Control son las principales tareas para su éxito (Rakos, 1990).

De acuerdo con Cuevas (2002) la planificación trata de definir desde el inicio del proyecto, todo aquello que pueda ser predecible, intentando minimizar la incertidumbre sobre los tres aspectos fundamentales de todo proyecto software: costo, plazo y funcionalidad. Weigers (1996) por su parte, considera que son cinco las dimensiones que deberían ser gestionadas en los proyectos software: Prestaciones, Calidad, Costo, Tiempo de entrega (Plazos) y Equipo de proyecto.

5. Planificación del Experimento Controlado

El presente estudio tiene como propósito explorar, si los equipos de desarrollo integrados con base en la Teoría de Roles de Belbin presentan un desempeño significativamente diferente de aquellos obtenidos por equipos integrados en forma aleatoria —denominados Tradicionales. Considerando las dimensiones propuestas por Weigers (1996), podemos decir que el experimento considera como factor a controlar, al equipo de proyecto, mantiene los plazos de entrega y las prestaciones, como parámetros homogéneos, y considera como variables dependientes: costo, calidad y las ganancias

5.1. Factor y Tratamientos

El factor considerado se corresponde con la estrategia utilizada para la integración de los equipos de proyecto, particularmente dos tratamientos diferenciados son considerados:

- *Equipos Compatibles (EC)*: Equipos de trabajo de tres integrantes, conformados con roles compatibles, de acuerdo con la Teoría de Belbin.
- *Equipos Tradicionales (ET)*: Equipos de trabajo de tres integrantes, seleccionados en forma aleatoria.

5.2. Hipótesis y Variables

El primer par de hipótesis estadísticas utiliza como variable dependiente, una métrica vinculada con la planificación: el *Costo* estimado para los materiales de los productos por desarrollar.

- H01: La media del costo estimado por los materiales requeridos de los ET es igual a la media del costo de los materiales estimados por los EC.
- H11: La media del costo estimado por los materiales requeridos de los ET difiere de la media del costo de los materiales estimados por los EC.

La segunda variable se encuentra vinculada con la tarea de seguimiento, en particular, con la calidad del producto que va desarrollando, para ello se utilizó como métrica la *Penalización* del equipo, la cual se integra con: (1) la penalización por las faltas cometidas durante el desarrollo (Mala Calidad: MC), y (2) la penalización por no cumplir con el número de productos propuesto (Incumplimiento: INC); las hipótesis estadísticas derivadas de dicha variable son las siguientes:

- H02: La mediana de la penalización fijada a los ET al final de la actividad, es igual a la media de la penalización asignada a los EC.
- H12: La mediana de la penalización fijada a los ET al final de la actividad, difiere de la media de la penalización asignada a los EC.

Finalmente, la tercera variable se encuentra vinculada con el desempeño total de tarea, utilizaremos como métrica las *Ganancias* del equipo; las hipótesis estadísticas derivadas para dicha variable son las siguientes:

- H03: La media de las ganancias de los ET es igual a la media de las ganancias obtenidas por los EC.
- H13: La media de las ganancias de los ET difiere de la media de las ganancias obtenidas por los EC.

La variable costo se obtuvo de los materiales contabilizados que fueron solicitados por los equipos, en cuanto a la variable penalización, ésta se obtuvo al final de la etapa de validación al sumar las penalizaciones por faltas y penalizaciones por incumplimiento; la tercera variable, ganancias, se obtiene de restar a los ingresos por los artefactos generados, los costos de los materiales y las penalizaciones.

5.3. Escenario Ficticio para la Tarea de Gestión

Con el propósito de generar un escenario que permitiese al equipo humano ejercitarse en actividades de gestión, se diseñó una tarea basada en escenarios ficticios (Aguilar, de Antonio, & Prieto, 2006), el objetivo en la tarea fue el de construir el mayor número de fachadas (casas) distintas posibles. Una fachada debe estar compuesta por una pared, techo, ventana(s) y puerta(s); las ventanas y puertas están pre-construidas de una sola pieza y los otros componentes tienen dimensiones predefinidas (ver figura 1).

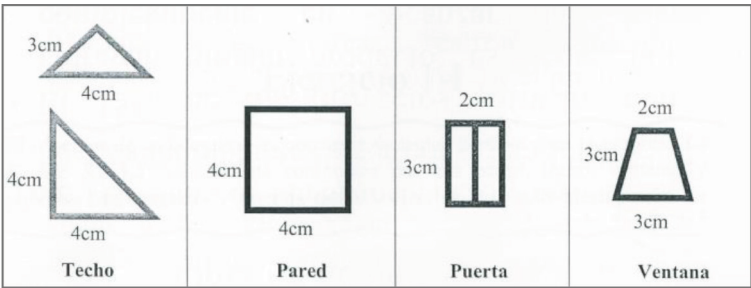


Figura 1 – Componentes pre-construidos para la construcción de Fachadas

Los equipos de trabajo deberán, en un tiempo establecido (30 Mins.), estimar el material requerido —con base en un catálogo proporcionado— y el número de fachadas a construir (Planificación), en un segundo período de tiempo limitado (40 Mins.), deberán ejecutar y dar seguimiento (Control) a la construcción de las fachadas para cumplir en tiempo con el número de fachadas propuesto.

Para la tarea de construcción, se establecieron un conjunto de condiciones: (a) cada fachada debe tener al menos uno de los cuatro tipos de componentes, (b) los componentes que configuran una fachada deberán estar unidas por cinta adhesiva, (c) cada fachada construida (en el plazo) tiene un valor de 50 pts., y (d) cada equipo debe adquirir el material necesario, previo a la construcción.

Finalizado el tiempo de construcción, con el apoyo de un comité de control de calidad —integrado por representantes de cada equipo— se procederá a la validación de las entregas de cada equipo (30 Mins.). Las penalizaciones a los equipos se establecieron en dos tipos: (a) penalizaciones por incumplimiento (INC), 15 unidades por fachada no entregada y (b) penalizaciones por mala calidad (MC), 2 unidades por defecto.

5.4. Diseño Experimental

El diseño experimental más apropiado para el estudio, es el diseño factorial con una fuente de variación y dos tratamientos (ver Tabla 2). La variable dependiente es una métrica numérica que se obtiene a través del instrumento de evaluación de los productos generados por los equipos de desarrollo.

Factor	Tratamientos	Variables Dependientes
Integración del equipo	Equipos Compatibles (EC)	Métricas Numéricas: Costos, Penalización y Ganancias
	Equipos Tradicionales (ET)	

Tabla 2 – Diseño factorial con una fuente de variación

5.5. Propuesta del Análisis Estadístico

Para el análisis de datos que serán recolectados, se generará una sección de estadística descriptiva con tablas de resumen de los valores obtenidos, así como gráficos de caja y bigotes para analizar visualmente el comportamiento de los mismos.

En el caso del análisis inferencial, las variables costo y ganancias se obtuvieron con una métrica numérica en una unidad de medida monetaria, la cual se corresponde con una escala de razón, para estos casos se seleccionó el análisis de varianza de una vía, ya que permite realizar pruebas de hipótesis para determinar si existe o no diferencias significativas entre las medias de los valores recogidos en la variable dependiente, para los diferentes niveles del factor (tratamientos). En el caso de la variable penalización, si bien es una métrica numérica, su análisis permite identificar que combina la métrica de penalización por incumplimiento, con la métrica de faltas en el producto, dos métricas numéricas que cualitativamente no son homogéneas; en el caso de la segunda, si bien es contabilizada con un mismo valor en todos los casos —2 puntos— las faltas nos son cualitativamente homogéneas, por lo que es necesario considerarla en una escala ordinal; la métrica utilizada obliga a elegir una prueba no paramétrica, como la U de Mann-Whitney, la cual permite comparar las medianas de dos muestras; es por ello que las hipótesis estadísticas H_{02} y H_{12} utilizan como parámetro la mediana, y no la media como en las otras hipótesis.

6. Ejecución del Experimento Controlado

Con base en el plan del experimento, se trabajó con estudiantes de la Licenciatura en Ingeniería de Software que se oferta en la Universidad Autónoma de Yucatán, en particular, alumnos inscritos a uno de los grupos de la asignatura Fundamentos de Ingeniería de Software que se impartió en el semestre agosto-diciembre de 2018. A lo largo del curso, el profesor —autor principal— analizó con el grupo temas relacionados con los procesos de desarrollo y gestión de la Ingeniería de Software, así como aspectos vinculados con factores humanos intrínsecos en el trabajo bajo el esquema de equipos de proyecto. En una sesión de las últimas semanas del curso, el grupo trabajó con la dinámica descrita en la sección correspondiente al escenario ficticio diseñado para simular la tarea de gestión de software.

6.1. Participantes

La muestra por conveniencia utilizada en el experimento, estuvo conformada por 24 de los 30 estudiantes que cursaban la asignatura denominada “Fundamentos de Ingeniería de Software” que se encuentra ubicada en el primer semestre del Plan de Estudios 2016 de la carrera antes citada. Con los 24 estudiantes se integraron 8 equipos de trabajo de tres integrantes cada uno, cuatro equipos con roles compatibles (EC), es decir, roles que no tienen conflicto entre ellos y otros cuatro equipos como grupo de control, integrados con estudiantes asignados de forma aleatoria (Equipos tradicionales: ET). Para la integración de los EC se utilizó la información obtenida de cada estudiante —rol primario— luego de administrar al grupo el instrumento de autopercepción propuesto por Belbin. La Tabla 3 ilustra la asignación de los equipos a los tratamientos.

Tratamientos	Equipos
EC	I, II, III, IV
ET	V, VI, VII, VIII

Tabla 3 – Asignación de sujetos experimentales por tratamiento

6.2. Sujetos Experimentales

Dado que las mediciones se obtendrían de los productos generados por los equipos de trabajo, el grupo de sujetos experimentales se encuentra integrado por los ocho equipos de trabajo. La integración de los cuatro equipos basados en la teoría de Belbin (Equipos con roles Compatibles: EC) se ilustra en la Figura 2; las letras dentro de los círculos indican el identificador del rol de equipo desempeñado por cada participante (consultar Tabla 1), por otro lado, el número en romano al centro de cada una de las configuraciones de los EC, representa al identificador del equipo –sujeto experimental.

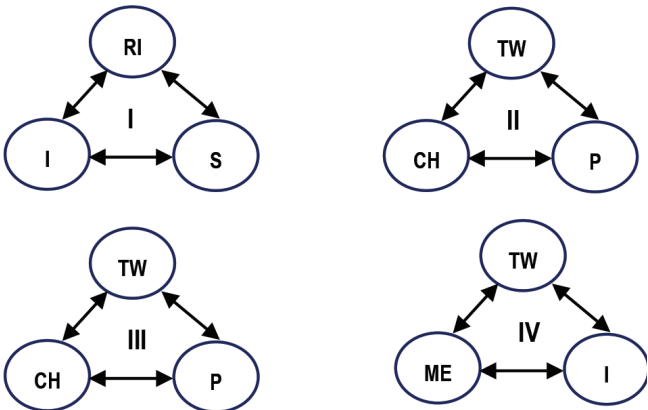


Figura 2 – Configuración de los cuatro equipos compatibles (EC)

6.3.Sesión Experimental

El experimento controlado se realizó en dos sesiones diferentes; en una primera sesión se administró el instrumento de autopercepción (operativamente se utilizaron los últimos 30 minutos de una sesión de clases presenciales) con el que pudo identificar el rol primario que puede desempeñar cada participante. En una segunda sesión –sesiones de 120 minutos– durante los primeros 15 minutos el profesor explicó la dinámica de la actividad, así como la integración de los equipos, seguidamente se dispuso de 30 minutos para organizar al equipo y acordar (con base en la estimación y diseños realizados) el número de fachadas por desarrollar, como tercera actividad, se dispuso de 40 minutos para el desarrollo de las fachadas reportadas y planificadas; concluida la actividad de desarrollo, el profesor dispuso de un cuarto plazo de 30 minutos para integrar un comité de calidad para evaluar los productos generado por los ocho equipos, el comité estuvo

conformado con un representante de cada equipo, manteniendo la restricción de que un representante no podría participar en la evaluación de su equipo.

Finalmente, el profesor utilizó los últimos 5 minutos de la sesión para el cierre de la dinámica instruccional, en la que se tuvo como intención, simular actividades de gestión software, con base en un escenario ficticio. La Tabla 4 presenta los datos recolectados con la dinámica experimental.

Equipo	Fachadas		Costos	Penalización			Ganancias
	Planeadas	Construidas		MC	INC	Total	
I	7	7	65	0	0	0	285
II	3	3	40	8	0	8	102
III	4	4	55	2	0	2	143
IV	6	6	45	16	0	16	237
V	6	5	40	2	15	17	193
VI	6	1	75	0	75	75	-100
VII	7	6	60	6	15	21	219
VIII	4	3	40	0	45	45	65

Tabla 4 – Datos Obtenidos del Experimento

7. Análisis de Datos del Experimento

En esta sección se presenta el análisis estadístico descriptivo e inferencial — realizado con apoyo del Software *Statgraphics*— de los datos recolectados con el experimento controlado.

7.1. Análisis Descriptivo

La Tabla 5 presenta algunas de las medidas más importantes de tendencia central y variabilidad para la variable Costos. Podemos observar que la media, la desviación estándar y el rango son menores en los EC; en el caso de la media la diferencia es ligera, sin embargo, la varianza presenta una diferencia de cerca de 6 unidades y el rango de 10.

Factor	#	μ	Mediana	O'	Min	Max	Rango
EC	4	51.25	50	11.0868	40.0	65.0	25.0
ET	4	53.75	50	17.0171	40.0	75.0	35.0

Tabla 5 – Resumen Estadístico de la Variable Costos

Para comparar visualmente los dos tratamientos, se generó un diagrama de caja y bigotes, dicho gráfico permite observar la dispersión y simetría de ambos conjuntos de datos. En la Figura 3 es posible identificar menor dispersión y mayor simetría de los EC en comparación de los ET, no obstante, el comportamiento de la gráfica no parece indicar diferencias significativas entre ambos conjuntos.

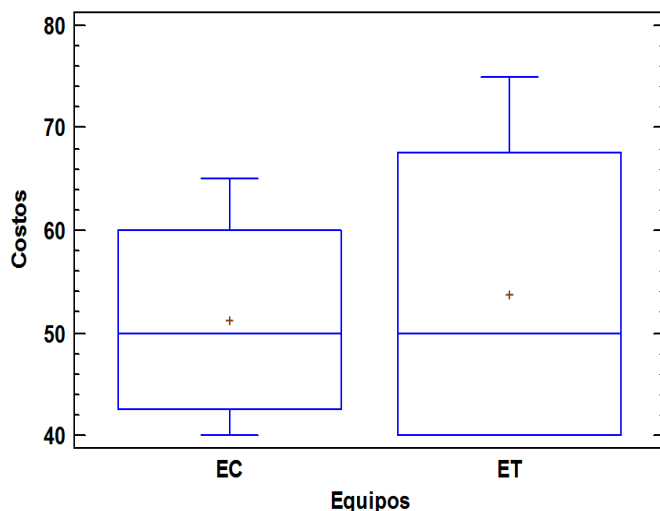


Figura 3 – Diagrama de Caja y bigotes de los tratamientos para la variable Costos

En cuanto a la variable penalización, la Tabla 6 contiene algunas de las medidas más importantes, se puede identificar que todas medidas derivadas de los EC, presentan valores muy por debajo de los que presentan los ET.

Factor	#	μ	Mediana	σ	Min	Max	Rango
EC	4	6.5	5.0	7.18795	0.0	16.0	16.0
ET	4	39.5	33.0	26.7021	17.0	75.0	58.0

Tabla 6 – Resumen Estadístico de la Variable Penalización

Con las diferencias reportadas en la Tabla 6, resulta de singular importancia comparar visualmente los dos tratamientos, y para ello se generó también un diagrama de caja y bigotes. En la Figura 4 es posible observar una menor dispersión de los EC en comparación de los ET, no obstante, lo más notable es el desfase del conjunto de datos hacia valores menores de toda la caja de los EC, de hecho, el valor máximo de los EC es menor que el mínimo de los ET, eso induce a pensar que existen diferencias estadísticamente significativas entre las penalizaciones obtenidas por los EC y las obtenidas por los ET.

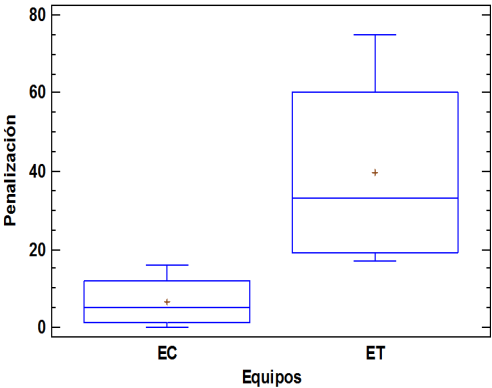


Figura 4 – Diagrama de Caja y bigotes de los tratamientos para la variable Penalización

Finalmente, para la variable ganancias, la Tabla 7 contiene el resumen de las medidas más importantes, en la tabla se puede observar que la media y mediana en los EC, presentan valores mayores que los ET, así mismo, a desviación estándar es menor en los EC.

Factor	#	μ	Mediana	O'	Min	Max	Rango
EC	4	192.25	191.0	84.3776	102.0	285.0	183.0
ET	4	94.25	129.0	145.948	-100.0	219.0	319.0

Tabla 7 – Resumen Estadístico de la Variable Ganancias

El gráfico de caja y bigotes de la Figura 5 permite identificar visualmente, menor dispersión y mayor simetría en los EC comparados con los ET, no obstante, el comportamiento de ambos conjuntos de datos no parece indicar diferencias significativas.

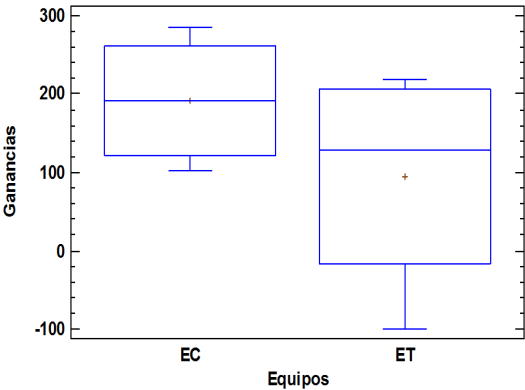


Figura 5 – Diagrama de Caja y bigotes de los tratamientos para la variable Ganancias

7.2. Análisis Inferencial

Con el propósito de evaluar estadísticamente las diferencias entre los tratamientos de las variables de razón Costos y Ganancias, se aplicó el Análisis de Varianza de una vía; el resultado de evaluar con ANOVA se ilustra en la Tabla 8.

Variable	Razón F	p-valor
Costos	0.06	0.8137
Ganancias	1.35	0.2891

Tabla 8 – ANOVA para la variable Costos y Ganancias

En ambas variables el valor p de la prueba F es mayor que 0.05 por lo cual, las hipótesis de nulidad H_{01} y H_{03} no pueden ser rechazadas, es decir, en ambos casos no existen diferencias estadísticamente significativas entre las medias de los dos tratamientos con un 5% de significancia.

Es importante mencionar que el modelo ANOVA tiene asociado tres supuestos que es necesario validar antes de utilizar la información que ofrece (Gutiérrez & De la Vara, 2012); los supuestos del modelo son: (1) Los errores experimentales de sus datos se distribuyen normalmente, (2) No existe diferencia entre la varianza de los tratamientos (Homocedasticidad) y (3) existe independencia entre las muestras.

Para validar el primer supuesto, se optó por generar un gráfico de probabilidad normal de los residuos; es una técnica gráfica para evaluar si un conjunto de datos se distribuye de acuerdo con la distribución normal. Como se puede ver en la gráfica que se ilustra en la Figura 6, en ambas variables, costos (a) y ganancias (b), los puntos no muestran desviaciones de la diagonal, por lo que es posible asumir que los residuos tienen una distribución normal.

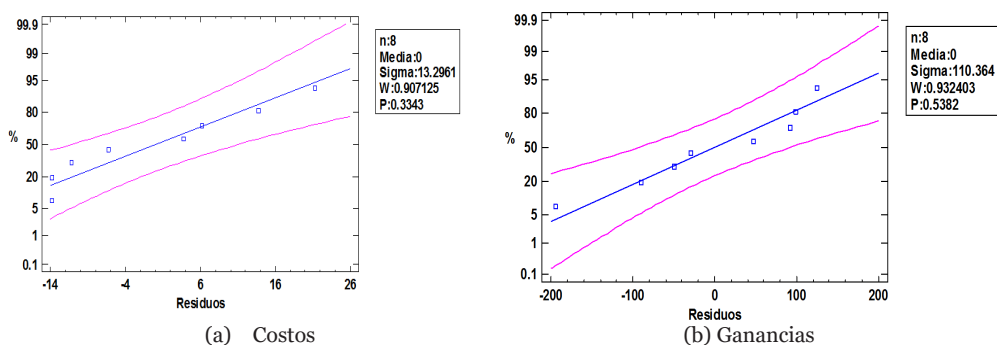


Figura 6 – Gráfico de Probabilidad Normal

Por su parte, la prueba de Levene permite evaluar diferencias significativas entre las varianzas de los dos conjuntos de datos (homocedasticidad); en la Tabla 9 se observa que en ambas variables (costos y ganancias) el p valor es mayor que 0.05, lo cual indica

que no existe una diferencia estadísticamente significativa entre las desviaciones estándar de los tratamientos, con un nivel del 95.0% de confianza.

Variable	Prueba	p-valor
Costos	1.71429	0.2383
Ganancias	1.3356	0.2917

Tabla 9 – Prueba de Levene para las Variables Costos y Ganancias

Para validar el supuesto de independencia de los datos, se generó un gráfico de residuales versus secuencia para cada variable. La Figura 7 permite identificar visualmente, en ambos casos, que no existe una tendencia en los datos, lo cual indica que los datos provienen de poblaciones independientes.

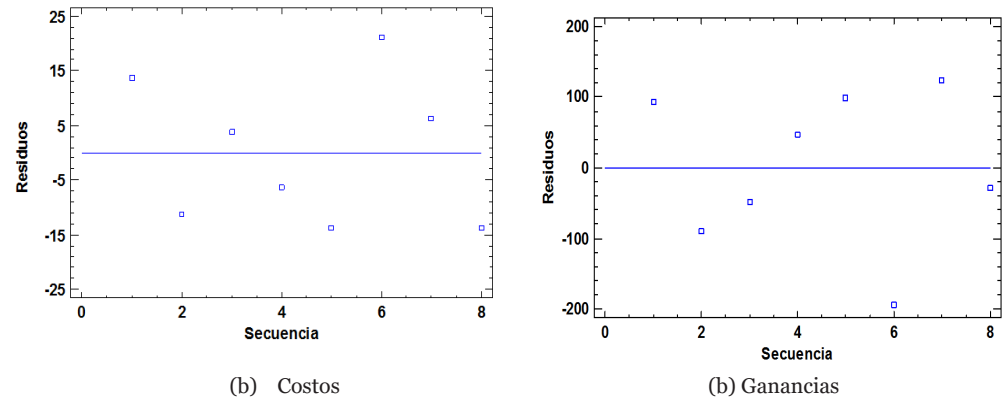


Figura 7 – Gráfico de Residuos vs. Secuencia

En cuanto a la variable penalización, como se mencionó en la sección 5.5, se utilizó la prueba de U de Mann-Whitney también conocida como prueba de Wilcoxon-Mann-Whitney, dicha prueba no paramétrica puede utilizarse para comparar las medianas de dos muestras independientes (Siegel & Castellan, 1995). En la Tabla 10 se identifica que el p-valor es menor que 0.05, lo cual indica no existe nulidad en las diferencias, es decir, que existe una diferencia estadísticamente significativa entre las medianas, con un nivel de confianza del 95.0%.

Equipos	Rango Promedio	W	p-valor
EC	2.5	16.0	0.0303826
ET	6.5		

Tabla 10 – Prueba de U de Mann-Whitney para las Variables Penalización

8. Conclusiones y Trabajos Futuros

En este artículo se presentó un experimento controlado con estudiantes de una carrera de Ingeniería de Software, el estudio compara métricas vinculadas con la planificación (costos), control (penalizaciones) y finalización del producto (ganancias), obtenidas del desempeño de dos tipos de equipo de trabajo, equipos integrados con base en la teoría de Belbin y equipos integrados aleatoriamente. Para la ejercitación de la tarea (desarrollo del proyecto), se diseñó un escenario ficticio basado en la construcción de fachadas, en el cual los sujetos experimentales (equipos de trabajo) deberían planificar y controlar el desarrollo de sus proyectos, de acuerdo con un conjunto de condiciones y restricciones establecidas, simulando de esta manera, la interacción entre las dimensiones citadas por Weigers (1996). Los resultados del experimento mostraron que, aunque en promedio los EC muestran obtención de mayores ganancias, las diferencias respecto de los ET no son estadísticamente significativas; no obstante, en lo que respecta a las penalizaciones, los EC difieren significativamente de los ET, obteniendo los primeros, un desempeño de mejor calidad. Con lo anterior, podemos concluir, que el uso de la teoría de roles de Belbin para la integración de equipos de trabajo, en lo que respecta a la tarea de gestión, al parecer influye positivamente en los proyectos, al generar equipos más fiables.

Como parte de los trabajos en curso, los autores han realizado un par de experimentos en el proceso de Diseño de Bases de Datos, uno de los estudios se reportará en un número especial de la presente revista a finales de 2020; en el caso del segundo estudio, resta realizar la validación de los diseños generados por los equipos de estudiantes, y con ello analizar los datos para obtener los resultados y las conclusiones que de los primeros se deriven.

Agradecimientos

Agradecemos el apoyo brindado por la Secretaría de Educación Pública (México) a través del proyecto P/PROFEXCE-2020-31MSU0098J-13.

Referencias

- Aguilar, R. (2008). *Una Estrategia Asistida por Entornos Virtuales Inteligentes (Tesis Doctoral)*. Universidad Politécnica de Madrid, Madrid, España.
- Aguilar, R., De Antonio, A., & Prieto, M. (2006). Un procedimiento basado en escenarios ficticios para evaluar el entrenamiento de equipos asistido por entornos virtuales colaborativos. *Revista Colombiana de Computación*, 7(2), 24-37.
- Aguilar, R., Díaz, J., & Ucán, J. (2019). Influencia de la Teoría de Roles de Belbin en la Medición de Software: Un estudio exploratorio. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, (31), 50-65.
- Aguilar, R., Muñoz, M., Díaz, J., & Ucán, J. (2020). Explorando la influencia de los Roles de Belbin en la Especificación de requisitos de software. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, (36), 34-39.

- Aguileta, A., Ucán, J., & Aguilar, R. (2017). Explorando la influencia de los roles de Belbin en la calidad del código generado por estudiantes en un curso de ingeniería de software. *Revista Educación en Ingeniería*, 12(23), 93-100.
- Aritzeta, A., Swailes, S., & Senior, B. (2005). *Team roles: Psychometric evidence, construct validity and team building*. University Hull.
- Belbin, M. (1981). *Management teams: Why they succeed or fail*. John Wiley & Sons.
- Belbin, R.M. (1993). *Team roles at Work*. Elsevier Butterworth Heinemann.
- Bourque, P., & Fairley, R. (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK V3.0)*. IEEE Computer Society.
- Cuevas, G. (2002). *Gestión del Proceso Software*. Centro de Estudios Ramón Areces.
- Estrada, E., & Peña, A. (2013). Influencia de los roles de equipo en las actividades del desarrollador de software. *Revista Electrónica de Computación, Informática, Biomédica y Electrónica*, 2(1), 1-19.
- Gutiérrez, H., & De la Vara, R. (2012). *Análisis y Diseño de Experimentos* (3ª ed). Mc Graw Hill.
- IEEE-CS (1990). *IEEE Standard Glossary of Software Engineering Terminology (IEEE Std 610.12-1990)*. IEEE Computer Society Press.
- Henry, S., & Stevens, K. (1999). Using Belbin's leadership role to improve team effectiveness: An empirical investigation. *The Journal of Systems and Software*, 44, 241-250.
- Humphrey, W. (2000). *Introduction to the Team Software Procesess*. Addison Wesley Longman Inc.
- Juristo, N., & Moreno, A. (2001). *Basics of Software Engineering Experimentation*. Kluwer Academic Publishers.
- Morales, N., & Vega, V. (2018). Factores Humanos y la Mejora de Procesos de Software: Propuesta inicial de un catálogo que guíe su gestión. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Información*, (29), 30-42. <http://dx.doi.org/10.17013/risti.29.30-42>
- PMI (2013). *A guide to the project management body of knowledge* (5th Ed). Project Management Institute.
- Pollock, M. (2009). Investigating the relationship between team role diversity and team performance in information systems teams. *Journal of Information Technology Management*, 20(1), 42-55.
- Rakos, J. (1990). *Software Project Management for Small to Medium Sized Projects*. Prentice Hall.
- Siegel, S., & Castellan, N. (1995). *Estadística no paramétrica aplicada a las ciencias de la conducta* (4ª ed.). Trillas.
- Weigers, K. (1996). *Creating a Software Engineering Culture*. Dorset House Publishing.

Critérios Editoriais

A RISTI (Revista Ibérica de Sistemas e Tecnologias de Informação) é um periódico científico, propriedade da AISTI (Associação Ibérica de Sistemas e Tecnologias de Informação), que foca a investigação e a aplicação prática inovadora no domínio dos sistemas e tecnologias de informação.

O Conselho Editorial da RISTI incentiva potenciais autores a submeterem artigos originais e inovadores para avaliação pelo Conselho Científico.

A submissão de artigos para publicação na RISTI deve realizar-se de acordo com as chamadas de artigos e as instruções e normas disponibilizadas no sítio Web da revista (<http://www.risti.xyz/>).

Todos os artigos submetidos são avaliados por um conjunto de membros do Conselho Científico, não inferior a três elementos.

Em cada número da revista são publicados entre cinco a oito dos melhores artigos submetidos.

Criterios Editoriales

La RISTI (Revista Ibérica de Sistemas y Tecnologías de la Información) es un periódico científico, propiedad de la AISTI (Asociación Ibérica de Sistemas y Tecnologías de la Información), centrado en la investigación y en la aplicación práctica innovadora en el dominio de los sistemas y tecnologías de la información.

El Consejo Editorial de la RISTI incentiva autores potenciales a enviar sus artículos originales e innovadores para evaluación por el Consejo Científico.

El envío de artículos para publicación en la RISTI debe hacerse de conformidad con las llamadas de los artículos y las instrucciones y normas establecidas en el sitio Web de la revista (<http://www.risti.xyz/>).

Todos los trabajos enviados son evaluados por un número de miembros del Consejo Científico de no menos de tres elementos.

En cada número de la revista se publican cinco a ocho de los mejores artículos enviados.

Os asociados da AISTI recebem a RISTI gratuitamente, por correio postal. Torne-se associado da AISTI. Preencha o formulário abaixo e envie-o para o e-mail aistic@gmail.com

Los asociados de la AISTI reciben la RISTI por correo, sin costo alguno. Hazte miembro de la AISTI. Rellena el siguiente formulario y remítelo al e-mail aistic@gmail.com



Formulário de Associado / Formulario de Asociado

Nome/Nombre: _____

Instituição/Institución: _____

Departamento: _____

Morada/Dirección: _____

Código Postal: _____ Localidade/Localidad: _____

País: _____

Telefone/Teléfono: _____

E-mail: _____ Web: _____

Tipo de Associado e valor da anuidade:

☐ Individual - 35€

☐ Instituição de Ensino ou I&D/Institución de Educación o I&D - 250€

☐ Outro (Empresa, etc.) - 500€

NIF/CIF: _____

Data/Fecha: __/__/____ Assinatura/Firma: _____



Revista Ibérica de Sistemas e Tecnologias de Informação
Revista Ibérica de Sistemas y Tecnologías de Información

©AISTI 2020 <http://www.aisti.eu>

