



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 1 9 • O c t o b e r 1 9



©AISTI 2019 <http://www.aisti.eu>

Nº 34

Edição / Edición

Nº 34, 10/2019

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 Coimbra

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

Gonçalo Paiva Dias, Universidade de Aveiro

Jose Antonio Calvo-Manzano Villalón, Universidad Politécnica de Madrid

Luís Paulo Reis, Universidade do Porto

Manuel Pérez Cota, Universidad de Vigo

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

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

Adriano Pasqualotti, Universidade de Passo Fundo, BR

Alberto Fernández, Universidad Rey Juan Carlos, ES

Alberto Freitas, FMUP, Universidade do Porto, PT

Alcinia Zita Sampaio, IST, Universidade de Lisboa, PT

Alejandro Medina, Instituto Nacional de Astrofísica, Óptica y Electrónica, MX

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 do Minho, PT

Ana Azevedo, Instituto Politécnico do Porto, 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

Andreia Bos, Instituto Federal do Rio Grande do Sul, BR

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

Antoni Lluís Mesquida Calafat, Universidad de les Illes Balears, ES

António Abel de Castro, ISEP, 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 Manuel Andrade, Universidade Católica Portuguesa, PT

António Palma dos Reis, ISEG, Universidade de Lisboa, PT

António Pedro Costa, Universidade de Aveiro, PT

António Pereira, Instituto Politécnico de Leiria, PT

António Teixeira, Universidade de Aveiro, 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

Baltasar García Pérez-Schofield, Universidad de Vigo, ES

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

Boris Almonacid, Global Change Science, CL

Bráulio Alturas, ISCTE - Instituto Universitário de Lisboa, PT

Brenda L. Flores-Rios, Universidad Autónoma de Baja California, MX

Brígida Mónica Faria, ESS, Instituto Politécnico do Porto, PT

Carlos Alexandre Silva, Instituto Federal de Minas Gerais, PT

Carlos Costa, Universidade de Aveiro, 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 Juiz, Universitat de les Illes Balears, ES
Carlos Rabadão, Politécnico de Leiria, PT
Carlos Rompante Cunha, Politécnico de Bragança, PT
Ciro Martins, Universidade de Aveiro, PT
Claudio de la Riva, Universidade de Oviedo, ES
Cristina M.R. Caridade, ISEC, Politécnico de Coimbra, PT
Dalila Durães, Politécnico do Porto, PT
Daniel Polónia, Universidade de Aveiro, PT
Daniel Riesco, Universidad Nacional de San Luis, AR
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
Diana Lancheros-Cuesta, Universidad de La Salle / Universidad Cooperativa de Colombia, CO
Diego Ordóñez-Camacho, Universidad Tecnológica Equinoccial, EC
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
Emiliano Reynares, CIDISI - UTN FRSF - CONICET, AR
Enric Mor, Universitat Oberta de Catalunya, ES
Erika Upegui, Universidad Distrital Francisco José de Caldas, CO
Eurico Carrapatoso, FEUP, Universidade do Porto, PT
Eusébio Ferreira da Costa, Escola Superior de Tecnologias de Fafe, PT
Fábio Magalhães, Universidade Nove de Julho, BR
Fábio Marques, Universidade de Aveiro, PT
Feliz Gouveia, Universidade Fernando Pessoa, 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, Universidade do Minho, PT
Francisco Javier Lena-Acebo, Universidad de Cantabria, ES
Francisco Restivo, Universidade Católica Portuguesa, PT
Franyelit María Suárez, Universidad UTE, EC
Frederico Branco, Universidade de Trás-os-Montes e Alto Douro, PT
Gabriel Alberto García-Mireles, Universidad de Sonora, MX
Gabriel Guerrero-Contreras, Universidade de Cádiz, ES
Gerardo Gonzalez Filgueira, Universidad da Coruña, ES
Germano Montejano, Universidad Nacional de San Luis, AR
Guilhermina Lobato Miranda, Universidade de Lisboa, PT
Hélder Gomes, Universidade de Aveiro, PT
Hélder Zagalo, Universidade de Aveiro, PT
Helena Garbarino, Universidad ORT Uruguay, UY
Hélia Guerra, Universidade dos Açores, PT
Henrique S. Mamede, Universidade Aberta, PT
Henrique Santos, Universidade do Minho, PT
Higino Ramos, Universidad de Salamanca, ES
Inês Domingues, CI-IPOP, PT
Isabel Pedrosa, Instituto Politécnico de Coimbra, PT
Isabel Lopes, Politécnico de Bragança, PT
Isaura Ribeiro, Universidade dos Açores, 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 Pascual Faria, FEUP, Universidade do Porto, PT
João Paulo Costa, Universidade de Coimbra, PT
João Rocha da Silva, FEUP, Universidade do Porto, PT
João Tavares, FEUP, Universidade do Porto, PT
Joaquim Ferreira, Universidade de Aveiro, PT
Joaquim José Gonçalves, Instituto Politécnico do Cávado e do Ave, PT
Joaquim Madeira, Universidade de Aveiro, PT
Joaquim Pinto, 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
Jorge Dias Villegas, Universidad de la Frontera, CL
Jorge Pires, IBM R&D Labs - Prague, CZ
Jose Alfonso Aguilar, Universidad Autonoma de Sinaloa, MX
José Alvarez-Garcia, Universidad de Extremadura, ES
José Borbinha, IST, Universidade de Lisboa, PT
José Braga de Vasconcelos, Universidade Atlântica, 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 Luis Pardo Díaz, Global School of Business. Empowement and Entrepreneursip, 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 Superior da Maia, PT

Jose M Molina, Universidad Carlos III de Madrid, ES
José Machado, Universidade do Minho, PT
Jose Maria Alvarez Rodríguez, Universidad Carlos III de Madrid, ES
José Martins, Universidade de Trás-os-Montes e Alto Douro, 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 Cavique, Universidade Aberta, PT
Luis Chamba Eras, Universidad Nacional de Loja, EC
Luis Fernandez-Sanz, Universidad de Alcalá, ES
Luís Ferreira, Instituto Politécnico do Cávado e do Ave, PT
Luis Enrique Sánchez Crespo, Universidad de Castilla-La Mancha, ES
Luis Rosales, Universidad Politécnica Antonio José de Sucre, VE
Luis Vilán-Crespo, Universidad de Vigo, ES
Luisa María Romero-Moreno, Universidad de Sevilla, 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
Margarita Ramirez Ramirez, Universidad Autonoma de Baja California, MX
Mari de los Milagros Gutierrez, Universidad Tecnológica Nacional, AR
Maria Amelia Eliseu, Mackenzie Presbyterian University, BR
Maria de la Cruz Del Río-Rama, Universidade de Vigo, ES
Maria Cecília Rosa, Politécnico da Guarda, PT
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 Castro, Instituto Politécnico do Porto, PT
Maria João Ferreira, Universidade Portucalense, PT
Maria João Gomes, Universidade do Minho, PT
Maria José Angélico, Instituto Politécnico do Porto, PT
Maria José Marcelino, Universidade de Coimbra, PT
Maria José Sousa, Universidade Europeia, PT
Mário Zenha-Rela, Universidade de Coimbra, 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
Miguel Olivero, Universidad de Sevilla, ES
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
Patrícia Cabral de Azevedo Tedesco, Universidade Federal de Pernambuco, BR
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 Fernandes, Instituto Superior Miguel Torga, PT
Pedro Miguel Moreira, Instituto Politécnico de Viana do Castelo, PT
Pedro Nogueira Ramos, ISCTE - Instituto Universitário de Lisboa, PT
Pedro Sánchez Palma, Universidad Politécnica de Cartagena, ES
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
Rui Rodrigues, Instituto Superior Miguel Torga, PT
Ramiro Delgado, Universidad de las Fuerzas Armadas, EC
Ramon Alcarria, Universidad Politécnica de Madrid, ES
Raul Laureano, ISCTE - Instituto Universitário de Lisboa, PT
Reinaldo Bianchi, Centro Universitário da FEI, BR
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
Rodrigo Rocha Silva, Universidade de Coimbra, PT
Rogerio Garcia, Universidade Estadual Paulista, 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
Rute Abreu, Politécnico da Guarda, PT
Samuel Sepúlveda, Universidad de La Frontera, CL
Samuel Silva, Universidade de Aveiro, PT
Santiago Gonzales Sánchez, Universidad Inca Garcilaso de la Vega, PE
Sara Balderas-Díaz, Universidad de Cádiz, ES
Sara Paiva, Politécnico de Viana do Castelo, PT
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
Sérgio Nunes, ISEG, Universidade de Lisboa, 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
Tomás San Feliu Gilabert, Universidad Politécnica de Madrid, ES
Valéria Farinazzo Martins, Universidade Presbiteriana Mackenzie, BR
Vania Baldi, Universidade de Aveiro, PT
Verónica Vasconcelos, ISEC, Politécnico de Coimbra, PT
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
iversidad del Sinú, CO

Era de la Cuarta Revolución Industrial

Age of the Fourth 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.34.0

Introducción

Los artículos presentados en este número fueron seleccionados de los trabajos aceptados en la Décimo Segunda Edición del Seminario Internacional de Ciencias de la Computación (SICC) 2019, realizado en la Ciudad de Medellín, Antioquia, Colombia durante los días 23, 24 y 25 de octubre del 2019. Este seminario ha tenido una tasa de aceptación del 16.32% de los artículos enviados.

En este número de la Revista Ibérica de Sistemas y Tecnologías de la Información (RISTI), se abordan temas relacionados con la Era de la Cuarta Revolución Industrial. Los trabajos de este número son las versiones originales presentadas por los autores en comunicación oral durante el congreso SICC 2019 y que fueron seleccionados después de realizarse un proceso de evaluación por pares doble ciego.

Cuarta revolución industrial

El término Industria 4.0 o Cuarta Revolución Industrial se refiere a una amplia gama de conceptos. Clasificar dichos conceptos y relacionarlos a una sola disciplina, así como distinguirlos específicamente es difícil (Lasi, Fettke, Kemper, Feld, & Hoffmann, 2014). Sin embargo, es posible tener en cuenta algunos conceptos fundamentales que se destacan al referirse a la Cuarta Revolución Industrial. Estos conceptos fundamentales se orientan en la definición de la industria 4.0 en el marco de Fábricas Inteligentes o *Smart Factory*, Tecnología Inteligente o *Smart Intelligent*, pasando por sistemas ciberfísicos, que hacen referencia a la fusión del nivel físico y el digital a tal nivel que la representación física y

digital no puede diferenciarse de manera razonable. También se consideran 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 (Sanchez, 2019).

En este contexto es posible afirmar que los enfoques e ideas en el ámbito de la Cuarta Revolución Industrial están situados entre disciplinas como ingeniería eléctrica, administración de empresas, informática, tecnología y software, negocios e información, ingeniería de software e ingeniería mecánica.

En términos generales, con el surgimiento de la Cuarta Revolución Industrial como un requisito de la red de producción global, las empresas y los países están conectados más estrechamente a través de redes de cadena de suministro en todo el mundo (Lasi et al., 2014).

La Cuarta Revolución Industrial significa que todas las innovaciones en tecnología digital maduran, especialmente en el sector manufacturero, durante el proceso de transformación. Elementos relacionados con tecnologías como Internet especialmente avanzada, inteligencia artificial y robótica, tecnología avanzada de sensores, tecnología en la nube que acelera la transferencia y utilización de datos, fabricación digital (impresión 3D), nuevos servicios y modelos de comercialización, teléfonos inteligentes y algoritmos para herramientas de navegación, entrega y transporte, servicios y vehículos autónomos, están incluidos en la cadena de valor global para que puedan ser compartidos por muchas compañías de muchos países (Cakmakci, 2019).

Por lo tanto, en la Cuarta Revolución Industrial, hay diferentes tecnologías que se unen de distintas maneras pero con el mismo fin. En este sentido, no solo cambia el mundo a nivel industrial sino que las mismas organizaciones deben cambiar. Gestionar las organizaciones globales para que enfrenten los retos de ser globales en esta nueva Era de la Cuarta Revolución Industrial es otro de los cambios imparables. Las organizaciones deben desarrollar mejores capacidades para ser interconectadas e internacionales. Deben lograr definir formas de revolucionarse con el mundo e impactar, cambiando incluso el enfoque de gestión de proyectos para afrontar retos relacionados con la Industria 4.0 (Cakmakci, 2019).

Como resultado, se espera un cambio industrial que va desde la orientación del producto o servicio que se genera hasta su relación con el mundo que la rodea, incluso en las industrias más tradicionales. Es posible visualizar en un futuro cercano la aparición de nuevos tipos de empresas que adopten nuevos roles específicos dentro del proceso de fabricación.

Con la planificación, análisis, modelado, diseño, implementación y mantenimiento (en resumen: el desarrollo) de sistemas de información tan complejos, dinámicos e integrados, surge una tarea atractiva y al mismo tiempo desafiante para las disciplinas que se incluyen hoy por hoy en esta revolución industrial, que puede asegurar y desarrollar aún más la competitividad de las empresas industriales.

La Industria 4.0 se centra en el establecimiento de productos inteligentes y procesos de producción. Por lo tanto, es necesario buscar alternativas para hacer frente a la necesidad de un rápido desarrollo de productos, una producción flexible y entornos

complejos (Vyatkin, Salcic, Roop, & Fitzgerald, 2007)(Brettel, Friederichsen, Keller, & Rosenberg, 2014). Por consiguiente, se habla de entornos de fabricación inteligente. Los productos inteligentes y personalizados comprenden el conocimiento de su proceso de fabricación y la aplicación del consumidor, además, lideran de forma independiente su camino a través de la cadena de suministro (Henning, 2013) (Brettel et al., 2014).

En resumen, los temas de la Cuarta Revolución Industrial son de gran importancia para la academia, a tal punto que el interés de la ciencia y la investigación en el área de las Ciencias de la Computación no es ajena al aporte que se debe realizar. Por lo tanto, hacen parte de los avances investigativos que se presentan en los trabajos seleccionados para ser publicados en este número de RISTI. Estos trabajos han sido seleccionados porque abordan temáticas de interés en la actualidad y se enmarcan en soluciones del mundo real con un enfoque asociado a la Era de la Cuarta Revolución Industrial.

Estructura

En el primer artículo los autores presentan una propuesta enfocada a la educación y hacen énfasis en los resultados obtenidos a partir del diseño, implementación y validación de una herramienta de software 3D, que permitió configurar robots Lego Mindstorms EV3 y VEX Robotics EDR, para cumplir una tarea específica. Con esta propuesta se pretende propiciar la adquisición de competencias técnicas tan necesarias en la Era de la Cuarta Revolución Industrial alrededor de temas como electrónica, informática, mecánica, y otras competencias transversales como el trabajo en equipo, la resolución de problemas y el pensamiento crítico. De los aspectos más relevantes de esta investigación está el uso de robots educativos, teniendo en cuenta de manera esencial el contexto y el usuario final del robot para lograr una solución. Específicamente esta propuesta tiene en cuenta los requerimientos iniciales del usuario del robot para, mediante el uso de la robótica pedagógica, solucionar una necesidad. Esta alternativa se convierte en el punto de partida de una educación que verdaderamente logre un impacto y un cambio de mentalidad, aprovechando los escenarios y la infraestructura existentes.

En el segundo artículo se presenta un enfoque de descripción de imágenes desde la perspectiva semántica y se evalúa mediante un análisis léxico-sintáctico para comprobar que las descripciones dadas sean parte del lenguaje definido. Dentro de los resultados obtenidos, los más relevantes muestran que, al tomar en cuenta las imágenes como un todo y no como objetos aislados se puede obtener información importante para describirla, incluso podrían realizarse inferencias sobre lo que sucede en la escena que conforma la imagen, por ello es muy importante la verificación sintáctica ya que de esta manera se evita añadir ruido al sistema y se trabaja solo con elementos válidos. Además, se muestra la validación sintáctica la cual realiza una verificación de las clases para solo procesar las válidas y de esta manera evitar añadir ruido al sistema con palabras que no son clases reconocidas tales como: conectores, numerales, etc.

En el tercer artículo se plantea un proceso de descubrimiento de conocimiento en el texto no estructurado de las historias clínicas del área de traumatología del Hospital San Vicente Fundación mediante minería de texto. Para el estudio se aplicaron técnicas de preparación de texto como eliminación de palabras no relevantes, sustitución de términos, eliminación de acentos y derivación de palabras. El resultado obtenido es la

conformación de un modelo de las palabras más relevantes en los registros clínicos del Hospital en el área de traumatología.

En el cuarto artículo los autores buscan mejorar la toma de decisiones de revisiones de artículos científicos a partir de la evaluación del revisor, es decir, predecir si un artículo científico debería ser aceptado o rechazado en función de los comentarios aportados por los revisores. Además, con su propuesta, los autores buscan lograr cuantificar la consistencia entre los comentarios de un revisor con respecto a su evaluación, usualmente expresada en una escala *multiclase*. Aunque con varios aspectos por mejorar, el trabajo muestra resultados alentadores, dado que presenta un beneficio cuantitativo de un aumento significativo en el tamaño de los datos y a la vez un mejor rendimiento de un clasificador basado en los datos aumentados en comparación a uno basado solo en los datos originales.

El quinto artículo aborda el problema de la transmisión de datos en redes de sensores inalámbricas (WSN), con garantías de tiempo real. Este es un tema que tiene referentes importantes en la comunidad científica internacional, pero que aún no dispone de una solución, que satisfaga este requerimiento completamente. El trabajo que se presenta en este artículo está relacionado con un método (algoritmo) de planificación para la transmisión de datos en tiempo real con redes de sensores inalámbricas (WSN), basado en técnicas de planificación estática y aplicado a redes que utilicen espacios de tiempo (time-slot) para la transmisión de datos en eventos periódicos, donde se garantice un tiempo acotado de transmisión.

El sexto artículo tiene como propósito la implementación de un controlador difuso en línea, empleando una plataforma del internet de las cosas (IoT), para obtener un conjunto de datos que se procesan, empleando redes neuronales, con el objetivo de obtener un modelo que permita la identificación de la dinámica de un sistema térmico. Con este trabajo se pretende evidenciar que la aplicación de herramientas de la inteligencia artificial y el internet de las cosas en contextos industriales traerá beneficios en el desarrollo de procesos con sistemas más eficientes y con capacidad de autogestión. Lo anterior considerando que la Cuarta Revolución Industrial trae consigo nuevas oportunidades en creación de productos y servicios dando paso a trabajos que requieren nuevos conocimientos, además de un alto componente de creatividad e innovación. Sobre todo, cuando el mundo en general busca ser cada vez más competitivo y tener una industria que satisfaga los requerimientos modernos.

El séptimo artículo está relacionado con la competitividad que las empresas de desarrollo de software buscan, respondiendo justamente a la exigencia de la Cuarta Revolución Industrial a nivel de satisfacción de requisitos. Por lo anterior, hoy en día, organizaciones como la ISO están enfocadas en el desarrollo de estándares que impulsen su competitividad, reforzando el uso de prácticas probadas de ingeniería de software y sean adecuados a las características específicas de este tipo de empresas, como lo es la serie de estándares ISO/IEC 29110. El objetivo de los autores es realizar una comparativa entre el estado del arte y el estado de la práctica referente a la implementación de dicho estándar, en entidades muy pequeñas. Los resultados más importantes muestran, que las principales debilidades están relacionadas con la falta de ejecución de actividades de verificación y validación, y el desarrollo y/o documentación de casos y procedimientos de prueba.

Finalmente, el octavo artículo está muy relacionado con la búsqueda de lograr calidad y competitividad en esta Era de la Cuarta Revolución Industrial. Este trabajo estudia mecanismos de medición de la calidad del diseño arquitectural de software ya que se ha identificado una carencia en métricas de calidad, sobre todo en aquellas enfocadas en medir el factor de flexibilidad y de herencia de implementación de una arquitectura de clases. Por lo tanto, los autores de este trabajo proponen un conjunto de cinco métricas de calidad, tres de ellas son para medir el factor de herencia de implementación, son: Factor de Herencia de Implementación (FHI), Factor de Herencia de Implementación por Jerarquía de Clases (FHIJ) y Factor Herencia de Implementación de una Arquitectura de Clases (FHIAC); las dos restantes son para medir el factor de flexibilidad, estas son: Factor de Flexibilidad de Clase (FFC) y Factor Medio de Flexibilidad de Clases (FMFC).

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 2019. 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 Segunda Edición del Seminario Internacional de Ciencias de la Computación.

Referencias

- Brettel, M., Friederichsen, N., Keller, M., & Rosenberg, M. (2014). How virtualization, decentralization and network building change the manufacturing landscape: An Industry 4.0 Perspective. *International Journal of Mechanical Industrial Science and Engineering*, 8(1), 37–44.
- Cakmakci, M. (2019). Interaction in Project Management Approach Within Industry 4.0. In *Advances in Manufacturing II* (pp. 176–189). Springer.
- Henning, K. (2013). Recommendations for implementing the strategic initiative INDUSTRIE 4.0. Germany: Federal Ministry of Education and Research.
- Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, 6(4), 239–242.
- Sanchez, D. O. M. (2019). Sustainable Development Challenges and Risks of Industry 4.0: A literature review. In *2019 Global IoT Summit (GIoTS)* (pp. 1–6).
- Vyatkin, V., Salcic, Z., Roop, P. S., & Fitzgerald, J. (2007). Now that's smart! Industrial Electronics Magazine. *IEEE*, 1(4), 17–29.

Índice / Index

EDITORIAL

Era de la Cuarta Revolución Industrial.....	ix
<i>Gloria Piedad Gasca-Hurtado, Liliana Esther Machuca-Villegas</i>	

ARTIGOS / ARTICULOS / ARTICLES

Herramienta de robótica educativa basada en Lego Mindstorms y VEX Robotics mediante software 3D y diseño mecatrónico	1
<i>Luis Carlos Correa Ortiz, Margarita María Vallejo Jiménez, John Jairo Martínez Puerta, Juan Alejandro Trujillo Posada</i>	
La Semántica de las Imágenes y el Análisis de su Contenido	20
<i>Catalina-Alejandra Vázquez-Rodríguez, Raúl Pinto-Elías</i>	
Descubrimiento de Conocimiento en Historias Clínicas mediante Minería de Texto	29
<i>Ana Isabel Oviedo Carrascal, David Sanguino Cotte, Natalia Andrea Restrepo Arango, Andrés Felipe Patiño Vélez</i>	
Heurísticas para Data Augmentation en NLP: Aplicación a Revisiones de Artículos Científicos	44
<i>Rubén Sánchez Acosta, Claudio Meneses Villegas, Brian Keith Norambuena</i>	
Planificación estática para la transmisión de datos en tiempo real con WSN	54
<i>Sixto Enrique Campaña Bastidas, Jorge Mario Londoño Peláez</i>	
Configuración de un Control de Temperatura en un Sistema Embebido de Bajo Costo, usando Herramientas de Inteligencia Artificial y el Internet de las Cosas	68
<i>Carlos Mario Correa Torres, Jhon Fredy Valencia Gómez</i>	
Análisis de las debilidades que presentan las Entidades Muy Pequeñas al implementar el estándar ISO/IEC 29110: Una comparativa entre estado del arte y el estado de la práctica	85
<i>Mirna Muñoz, Manuel Peralta, Claude Y. Laporte</i>	
Métricas para la Medición del Factor de Flexibilidad y el Factor de Herencia de Implementación de Sistemas de Software	97
<i>Orlando Ortiz-Gutierrez, René Santaolaya-Salgado, Olivia-Graciela Fragoso-Díaz, Juan-Carlos Rojas-Pérez</i>	

Herramienta de robótica educativa basada en Lego Mindstorms y VEX Robotics mediante software 3D y diseño mecatrónico

Luis Carlos Correa Ortiz¹, Margarita María Vallejo Jiménez², John Jairo Martínez Puerta³, Juan Alejandro Trujillo Posada¹

lcco@umanizales.edu.co, mvallejoj@sena.edu.co, jjmartinezp@sena.edu.co, juanalejandro@umanizales.edu.co

¹ Facultad de Ciencias e Ingeniería, Universidad de Manizales, Carrera 9 No. 19-03, 17001, Manizales, Colombia.

² Sena Tecnoacademia Risaralda, Calle 73 Bis Carrera 21, 661001, Dosquebradas, Colombia

³ Centro Atención Sector Agropecuario, Sena Risaralda. Sistema de Investigación Sennova, Carrera 8 #26-79, 600002, Pereira, Colombia

DOI: 10.17013/risti.34.1-19

Resumen: La robótica propicia la adquisición de competencias técnicas en áreas tales como electrónica, informática, mecánica, y transversales como trabajo en equipo, resolución de problemas y pensamiento crítico. La mayoría de los currículos en las instituciones de educación básica, media y superior del país que utilizan robots educativos no tienen en cuenta el contexto ni el usuario final del robot, incluidas las dos instituciones de este estudio: Tecnoacademia Sena Risaralda y Universidad de Manizales. En este trabajo se expone el diseño, implementación y validación de una herramienta de software 3D, que permitió configurar robots Lego Mindstorms EV3 y VEX Robotics EDR, para cumplir una tarea específica, teniendo en cuenta los requerimientos iniciales del usuario del robot. Se usaron herramientas de diseño mecatrónico basadas en el despliegue de la función de calidad QFD y Design Thinking. El software hará parte de los currículos de ambas instituciones en sus cursos de robótica.

Palabras-clave: Robótica educativa; Ingeniería; Sensores; Modelado 3D; Diseño mecatrónico.

Educational robotics tool based on Lego Mindstorms and VEX Robotics using 3D software and mechatronic design

Abstract: Robotics encourages acquiring many technical skills in areas such as electronics, computer science, mechanics and also soft skills as teamwork, problem solving and critical thinking. Most of the curricula of elementary, middle and higher education institutions in Colombia that incorporate educational robots do not consider the context or the end user of the robot, including the two institutions involved in this study: Tecnoacademia Sena Risaralda and University of Manizales. In this work, design, implementation and testing of a 3D software tool is described.

This tool allows configuring Lego Mindstorm EV3 and VEX Robotics EDR robots to compel a specific task, considering the initial requirements of the end user. Mechatronics design tools as QFD and design thinking was used. Finally, the developed software will be incorporated in robotic courses in both institutions.

Keywords: Educational Robotics; Engineering; Sensors; 3D Modeling; Mechatronic Design.

1. Introducción

La alfabetización científica es un proceso central en el desarrollo de las sociedades, existiendo razones tanto económicas como de movilidad social que invitan a fomentar y potenciar su desarrollo. Según la OCDE (*Organización para la Cooperación y el Desarrollo Económicos*), el alfabetismo científico no sólo involucra el tener conocimientos científicos sino la capacidad de aplicarlos funcionalmente en el medio donde se desenvuelven las personas (García & Reyes, 2012). La educación STEM (*Science, Technology, Engineering and Mathematics*) creció de la preocupación por el bajo número de futuros profesionales en comparación con la alta demanda de profesionales STEM, en detrimento de la competitividad económica y educativa (Mizell & Brown, 2017). La robótica educativa hace parte de la educación STEM (Silva-Ortigoza et al., 2014) y más recientemente STEAM (incorporando la palabra *Arts*) (OEA, 2016), el concepto se hace más amplio incluyendo innovación, creatividad y solución de problemas de la vida real (Ghitis & Alba, 2014).

El resultado de una investigación en robótica educativa en diferentes partes del mundo arroja la tendencia de dos enfoques; el enfoque pedagógico, y el enfoque técnico. En el pedagógico se habla de la utilización de la robótica como una estrategia para la enseñanza de varias áreas escolares como matemáticas, cálculo, ciencias, informática, diseño, etc., y por otro lado en el enfoque técnico se centra en la llamada robótica multiagente o cooperativa y en la investigación para generar innovación en el hardware y software de robótica (Salas-López et al., 2012).

En el mundo se promueve el uso de la robótica como herramienta educativa en instituciones de educación básica primaria y secundaria, con el fin de fortalecer las competencias y el desempeño académico del educando mediante herramientas innovadoras que permitan una construcción más dinámica del conocimiento (Cabello-Ochoa & Carrera-Farran, 2017) y además mediante STEM (Rivadeneira, 2019). Numerosas investigaciones demuestran el interés global por la inserción de herramientas robóticas en las aulas de clase. En 1975, en Francia, aparece una primera utilización con fines educativos de la robótica. En 1989, en México y 1998 en Costa Rica (Acuña, 2004). En España (Pinto, Barrera, & Pérez, 2010), Chile (Monsalves, 2019) y Panamá (Moreno et al., 2012) se generan entornos de aprendizaje basados fundamentalmente en la acción de los estudiantes, quienes simulan fenómenos y mecanismos, diseñando y construyendo prototipos robóticos.

1.1. Robótica educativa en Colombia

En Colombia la Universidad Pedagógica Nacional lideró en 2004 proyectos para la formación de docentes en robótica educativa. En 2008, la Universidad del Cauca con

“Plataforma de Robótica y Automática Educativa de Computadores para Educar”. En Cali, el Colegio Bolívar, el Instituto Nuestra Señora de la Asunción INSA y la Fundación Gabriel Piedrahita Uribe han implementado talleres y en la Universidad Nacional de Colombia se realizó la construcción de un robot móvil didáctico para trabajo con niños de básica primaria (Pinto et al., 2010).

En la Universidad de Cundinamarca, con estudiantes de algunos colegios de Chía, se han construido “robots autónomos seguidores de línea” (Silva-Ortigoza et al., 2014) y se han conformado semilleros para desarrollar diferentes proyectos en áreas interdisciplinarias (Sánchez, 2003), creando la comunidad virtual “Mundo Robótica”, donde la comunidad educativa y académica interesada en la temática participa en un espacio virtual en el que pueden compartir sus experiencias, conocer nuevas prácticas de otros miembros de la red y construir conocimientos en torno a la robótica educativa (Bravo & Forero, 2012).

Los primeros intentos por incorporar didácticas diferentes en el Sena (Servicio Nacional de Aprendizaje) se hicieron bajo la denominación de Tecnología Básica Transversal (TBT), desde finales de la década de los noventa, bajo la modalidad de formación por proyectos. La estrategia ha sido estructurar el aprendizaje de los alumnos a través de pequeños proyectos, a fin de impulsar el desarrollo de competencias transversales, soportadas en metodologías activas como pueden ser la resolución de problemas, la elaboración de proyectos, la simulación, el análisis o el estudio de casos entre las más empleadas (Sáez-Carreras, Bas-Peña, & Bedmar-Moreno, 2007). Esta estrategia multidisciplinaria basada en proyectos tiene sus orígenes en la Universidad Aalborg en Dinamarca en 1974, en colaboración con la industria de ese entonces (Dym, Agogino, Eris, Frey, & Leifer, 2005).

La formación correspondiente tanto a la educación secundaria como a los primeros semestres de la educación superior, tiene un enfoque diferente, no centrado en el perfil ocupacional de los estudiantes sino en el desarrollo de competencias en ciencia, tecnología e innovación, aprendizaje esencial hoy en día a tempranas etapas del desarrollo humano. Una de las metas más importantes de la educación colombiana es reforzar y ayudar en la incorporación de nuevas tendencias en el sector educativo y para tal fin se incorporan nuevas tecnologías en las instituciones educativas desde dos ámbitos: las TIC (Tecnologías de la Información y la Comunicación) y desde la tecnología como medio de construcción (Corchuelo, 2015).

El Ministerio de Educación Nacional de Colombia (2008) realizó una investigación dirigida a estudiantes de secundaria, proponiendo un modelo de ambientes de aprendizaje en robótica, concluyendo que no existe un consenso a nivel mundial sobre características estándar. Afirma además que formar en tecnología es mucho más que ofrecer una capacitación para manejar artefactos y expone algunos desafíos:

- Motivar mediante curiosidad científica y tecnológica, mostrando pertinencia en la realidad local y su contribución a la satisfacción de necesidades.
- Reconocer el saber tecnológico como solución a problemas que contribuyen a la transformación del entorno.
- Participación de los individuos en la evaluación y control de la tecnología en la sociedad, desarrollando una reflexión crítica.

- Reconocer la existencia de varios enfoques para la solución de problemas de tecnología, como el diseño, la innovación, la detección de fallas y la investigación.
- Brindar a las personas herramientas para participar asertivamente en su entorno.

Existen currículos que enseñan a diseñar, construir y programar kits de robótica, como resultado de la experimentación de la robótica pedagógica en el mundo. En Colombia y especialmente en la zona cafetera algunos currículos están basados en kits robóticos pertenecientes a la Tecnoacademia Sena Risaralda (Vallejo-Jiménez, 2018) y a la Universidad de Manizales, tales como Lego Mindstorm EV3® (Chetty, 2015) y VEX Robotics®. Estos currículos, basados en la metodología STEM (Correll, Wailes, & Slaby, 2014), muchas veces se limitan a la enseñanza del funcionamiento del robot, pero sin enfocarse en dar solución a las problemáticas o necesidades del entorno, mediante el desarrollo de prototipos iniciales para ello, excluyendo a las competencias, que se dan cuando los estudiantes poseen dominio sobre las plataformas robóticas.

En este ambiente de aprendizaje innovador los estudiantes ocupan la mayor parte del tiempo simulando fenómenos y mecanismos, diseñando y construyendo prototipos que son representaciones micro de la realidad tecnológica circundante, o son sus propias invenciones (Vallejo-Jiménez, Martínez-Puerta, Bedoya-Agudelo, & Salgado, 2018). La puesta en marcha de un proyecto de robótica requiere del conocimiento de diversas áreas, pues el robot es un artefacto mecatrónico que surge de la combinación de mecánica, electrónica y ciencias de la computación, campos que se influyen mutuamente y que finalmente son dotados con cierta inteligencia o funciones autónomas (Isermann, 2008). Todas las áreas influyen en las demás, esto significa que se debe llevar a cabo ingeniería simultánea con el objetivo de diseñar un sistema global integrado y también crear efectos sinérgicos (Isermann, 2008), (Odorico, 2005).

1.2. Metodología para construcción de prototipos

Cuando se habla de diseñar soluciones a través de prototipos robóticos, pueden resultar productos ineficientes, incremento de costos por ajustes y largos períodos de desarrollo del producto, todo esto cuando el diseño no se realiza de manera concurrente (Steimer, Fischer, & Aurich, 2017). Es allí donde se aplican las metodologías de diseño de productos, en especial de diseño mecatrónico, que permiten estimar tiempos de diseño y prototipado rápido debido a las exigencias de la industria y al ciclo de vida cada vez más corto de los productos (De Silva & Behbahani, 2013).

Existen diversas metodologías y herramientas para el diseño industrial de prototipos (Dym et al., 2005) (Brown, 2008) (Sena, 2015) (Gómez-González, Lecuona-López, & Hernández-Monsalve, 2016). También existen metodologías centradas en el diseño mecatrónico (Dode, Greig, Zolfaghari, & Neumann, 2016) (Corchuelo, 2015), entre ellas la ingeniería simultánea o concurrente (Macas, 2017); fábrica digital (Kuhn, 2006); metodología de diseño para sistemas mecatrónicos, guía VDI 2206 (Dode et al., 2016) (Gausemeier, Dorociak, Pook, Nyßen, & Terfloth, 2010); modelo de tres ciclos de ingeniería de producto (Kurdve, Zackrisson, Wiktorsson, & Harlin, 2014); desarrollo de doble hélice de sistemas de producción y productos (Pessôa & Trabasso, 2016); ingeniería de sistemas basada en modelos (Steimer et al., 2017); Cociente de Diseño Mecatrónico,

MDQ, por sus siglas en inglés (De Silva & Behbahani, 2013); métodos ágiles (Cooke, 2012) (Edin Grimheden, 2013); despliegue de la función de calidad y Casa de Conceptos HoC (por sus siglas en inglés) (Punz, Hehenberger, Follmer, & Zeman, 2011) (Hassan, Siadat, Dantan, & Martin, 2010) y Design Thinking (Brown, 2008).

No existe un consenso que defina cual metodología es mejor, pues estas se ajustan según el objetivo o prototipo que se desee construir, pero la mayoría conciben ciertos pasos o etapas en común para la construcción de un artefacto mecatrónico (Carvajal, 2013):

- Diseño conceptual: Se define la función de cada disciplina por módulos de diseño mecánico, eléctrico-electrónico y diseño de software; en cada una se definen parámetros y variables de diseño.
- Modelado: Se realiza el modelado gráfico y/o modelado matemático a través de software especializado para cada disciplina.
- Simulación: Se construye la simulación en un ambiente virtual, por módulos disciplinares o integrados parcialmente, dando como resultado la simulación del prototipo mecatrónico a través de software especializado.
- Integración: Se define la integración de los módulos disciplinares, optimización del diseño integrado, aplicación de software especializado, desarrollo del diseño en ambiente de Ingeniería concurrente con ajustes y pruebas de la función transdisciplinar del producto.

En las metodologías verificadas se observa poca uniformidad para la elección de las herramientas que permitan llevar a cabo las etapas anteriores para la construcción de prototipos, existiendo una gran variedad de ellas.

Uno de los primeros intentos en el país para desarrollar y adaptar herramientas concretas para el desarrollo de prototipos lo hace el Sena en su programa Tecnoparque Colombia, quienes apropiaron desde el año 2015 una variante del enfoque *Design Thinking* (Dym et al., 2005)(Sena, 2015) mediante el cual se desarrollan prototipos con base en una necesidad existente y bajo la premisa que el diseño en ingeniería puede verse materializado mediante la aplicación de herramientas definidas de Diseño Industrial. En otras instituciones como la Universidad de Manizales, estos procesos de construcción de artefactos mecatrónicos están regidos por el currículo existente y no por una metodología estándar, dando como resultado que los estudiantes construyan exitosamente estructuras robóticas y programan sus artefactos de manera correcta, pero sin un propósito, norte o contexto definido, para luego de ser construido, encontrarle una utilidad.

En las diferentes sedes de las Tecnoacademia Sena presentes en el país se vienen desarrollando procesos similares en la concepción de un robot para solucionar una necesidad usando algunas herramientas Design Thinking, pero no se tienen establecidas unas herramientas estándar en todas las sedes que permitan que el diseño mecatrónico sea óptimo. Además, las limitaciones de presupuesto hacen que la integración con el hardware de los robots esté al alcance de pocas instituciones educativa de educación básica, media e incluso superior. Una de las posibilidades es centrarse en la realización de una herramienta de software centrada en la etapa de Integración descrita anteriormente, en donde no sea necesario tener el hardware del robot al alcance de los estudiantes para desarrollar la solución.

El objetivo del estudio es implementar una herramienta de software 3D que permita configurar robots Lego Mindstorms EV3 y VEX Robotics EDR para cumplir una tarea específica teniendo en cuenta los requerimientos iniciales del usuario del robot. Para cumplir este objetivo se hace necesario como primera medida levantar los requerimientos tanto en términos de la herramienta robótica, como del software a desarrollar, para posteriormente implementar la herramienta y finalmente validarla en el marco de los cursos de robótica de ambas instituciones.

La herramienta plantea una necesidad en un contexto determinado y a partir de esta y de los requerimientos que el cliente tenga (quien usará el robot) se construye la solución, al contrario de lo que se hacía anteriormente, buscar el sentido al artefacto robótico ya construido. Puede usarse adicionalmente tanto como instrumento de enseñanza aprendizaje en un curso de robótica en el cual no exista el hardware de robots Lego y VEX como una herramienta de apoyo centrada en las necesidades del usuario. El software, además de recoger los requerimientos del usuario y transformarlos en criterios técnicos, posee un tutorial de las partes del robot y su funcionamiento. Es allí donde se convierte en el primer acercamiento al funcionamiento de los robots, como sensibilización e introducción a los cursos de robótica de ambas instituciones. El aprendizaje significativo en los educandos tiene que ver con el acercamiento a metodologías de diseño mecatrónico centrado en el usuario y en las problemáticas, usando herramientas como QFD. No se pretende encasillar el software para que sea usado en los cursos de robótica de una manera particular, se desean exponer sus ventajas para que pueda ser aprovechado de la mejor forma posible en los cursos.

2. Materiales y Métodos

La herramienta 3D se diseñó en la plataforma de desarrollo en tiempo real Unity® (<https://unity.com/es>) como un juego para el sistema operativo Windows, en donde inicialmente al jugador o usuario del robot se le plantea la necesidad específica de encontrar un objeto mediante Lego Mindstorms EV3 o VEX EDR. En primer lugar, el software le hace once preguntas al usuario relacionadas con lo que desea que haga su robot para resolver el reto: ¿Es importante que su robot se mueva en todas direcciones? ¿O que el robot evada obstáculos? ¿Piensa que su robot debe buscar el objeto en un escenario cerrado o interior? ¿O desea que sea en uno exterior?, a las cuales el usuario debe responder asignando valores en una escala de cero a cinco a cada pregunta. Un valor de cero significa que no es importante que el robot posea esa característica y uno de cinco significa que la considera indispensable.

A continuación, se le plantea la elección del robot en el escenario que se planteó anteriormente (interior o exterior). Puede elegir entre una configuración de Lego Mindstorms EV3 o de VEX EDR. Cada uno de los robots fueron diseñados pieza por pieza, por lo tanto, antes de la elección el jugador puede analizarlo, podrá manipularlo mediante vistas, además de ver el nombre de las partes que lo componen y de ver una corta descripción de lo que hace cada una.

Luego de elegir el robot, se presenta una animación del comportamiento del robot en el escenario correspondiente. La finalidad de la herramienta no es programar los movimientos del robot para que puedan visualizarse, es la elección de los sensores y

actuadores adecuados para el robot, dependiendo de los deseos del jugador. El proyecto se realizó en Unity pues permitió integrar las piezas del diseño 3D de otro tipo de software.

2.1. Etapas

El software fue concebido para cursos introductorios de robótica basada en Lego y VEX, usando herramientas de diseño mecatrónico, centradas en la problemática. Los estudiantes de la Tecnoacademia Sena Risaralda pertenecen a los grados 7º a 10º de educación secundaria de los colegios de la región. En este caso la formación es básica y no involucra conceptos de cinemática, dinámica o incluso electrónica avanzada, siendo las 140 horas de formación por año insuficientes para este desarrollo. Por parte de la Universidad de Manizales, los estudiantes pertenecen a los tres primeros semestres de Ingeniería de Sistemas y Telecomunicaciones y el curso contempla 64 horas. En ambos casos, la metodología de trabajo se basa en el desarrollo de competencias en ciencia, tecnología e innovación con las metodologías STEAM y Design Thinking, adaptadas e implementadas por ambas instituciones.

El proyecto se realizó en dos etapas, cada una de ellas con una metodología diferente:

- Etapa 1: Se establecieron los requerimientos o deseos del usuario de los robots y estos se tradujeron en criterios técnicos para su construcción, todo a través de la metodología de diseño mecatrónico basada en Despliegue de la Función de Calidad o QFD.
- Etapa 2: Estos criterios técnicos fueron el insumo para la realización del software 3D en Unity a través de metodología Design Thinking de la Red Tecnoparque Sena.

A continuación, se enumeran las diferentes actividades que se llevan a cabo en cada etapa.

2.2. Etapa 1

Para las plataformas de robótica educativa (Lego Mindstorms EV3 y VEX EDR) disponibles tanto en la Tecnoacademia Sena Risaralda como en la Universidad de Manizales, se establecieron dos configuraciones, que corresponden a su vez a dos escenarios (interiores y exteriores), según la Tabla 1.

Plataforma Robótica	Configuración de Robot	Escenario	Características Escenario
Lego Mindstorms EV3	Lego1	Interiores	Espacio conocido y cerrado como casas, talleres, almacenes, bodegas, etc.
VEX EDR	Vex1		
Lego Mindstorms EV3	Lego1	Exteriores	Espacio desconocido, abierto y abrupto como el campo, canchas, estadios, etc.
VEX EDR	Vex1		

Tabla 1 – Plataforma Robótica – Escenario.

La finalidad de la Etapa 1 fue establecer las características técnicas para el robot en cada escenario, a partir de los requerimientos o deseos de los clientes, que correspondieron a las preguntas iniciales del juego, respondidas por los estudiantes. Estas respuestas fueron el insumo de la matriz de la metodología de diseño mecatrónico QFD, cuyos resultados arrojados correspondieron a los requerimientos técnicos de los robots (sensores, actuadores, etc.).

Para ello se hizo un diseño de experimento, de la siguiente manera:

1. Objeto del experimento: Obtener los criterios técnicos del robot para cada uno de los escenarios (interiores y exteriores).
2. Fuentes de variación: Los pesos asignados a los requerimientos de los clientes, valores de 0 (sin importancia) a 5 (importante).
3. Factores de tratamiento: factores cualitativos reflejados en la tabla 2.

Categoría de los factores	#	Factores de tratamiento	Niveles
<i>Facilidad de manipulación por parte del usuario</i>	1	Manipulación del robot por parte del usuario	Fácil (se puede tomar con ambas manos) y Difícil (no se puede manipular con ambas manos)
	2	Tipo de interfaz	Amigable o poco amigable
<i>Facilidad de programación</i>	3	Cumplimiento de tarea con orden inicial	Cumple o no cumple
	4	Posibilidad de elección del modo de la tarea	Cumple o no cumple
<i>Facilidad de movimiento</i>	5	Movimiento en cualquier dirección	Se mueve o no se mueve
	6	Evasión de obstáculos	Evade o no evade
	7	Movimiento autónomo por un tiempo prudente	Se mueve o no se mueve
	8	Movimiento en exteriores	Se elige el escenario exterior o no se elige
	9	Movimiento en interiores	Se elige el escenario interior o no se elige
<i>Interacción con el ambiente</i>	10	Reacción a la luz	Reacciona o no reacciona
	11	Reacción a colores	Reacciona o no reacciona

Tabla 2 – Diseño de Experimento.

4. Factores externos:
 - Marca del robot: Lego o VEX. No se puede medir al inicio.
 - Grado de escolaridad de la unidad experimental: Colegio o cursando universidad, si se puede establecer al inicio.
5. Unidades experimentales:
 - Cursos 1.1, 1.2, 1.3 y 1.4 de Estudiantes de Robótica de la Tecnoacademia Sena Risaralda
 - Curso 2.1 de estudiantes de Robótica de la Universidad de Manizales.
 - Bloques: 5 (cada uno de los cursos).
6. Regla de asignación: Por bloque

7. Medidas que se realizarán: Variables respuesta: 12 criterios técnicos para la configuración en interiores y 12 criterios técnicos para la configuración en exteriores, medidos en porcentaje de importancia o peso, reflejados en la tabla 3.

1	Movimiento por ruedas
2	Movimiento por orugas
3	Baterías recargables
4	Sensor de color
5	Sensor ultrasónico
6	Sensor <i>push-button</i> (final de carrera)
7	Girosensor
8	Actuador con motor mediano
9	Monitoreo por computadora en tiempo real
10	Programación mediante software por computador
11	Sensor de luz
12	Programación in situ (en el mismo robot)

Tabla 3 – Variables Respuesta.

8. Instrumento de medida: Matriz de Despliegue de la Función de Calidad QFD
9. Experimento piloto: Tecnoacademia Sena Risaralda: Cuatro cursos con cuatro subgrupos cada uno. Universidad de Manizales: Un curso con cuatro subgrupos.
10. Modelo matemático: Relaciones, pesos y fórmulas propios de la QFD (Hassan et al., 2010).
11. Análisis estadístico: Media de las variables respuesta por cada escenario. Se eligieron los 6 criterios con mayor ponderación relativa, por escenario, para la construcción de los modelos Lego1, Lego2, Vex1 y Vex2.

2.3. Etapa 2

La metodología que se toma es la establecida por la Red Tecnoparque del SENA (Sena, 2015) y adaptada de IDEO (Brown, 2008) para el desarrollo de prototipos:

1. Inspiración: Se establece el escenario PON: problema, oportunidad y/o necesidad, el cual corresponde a la realización del software con robots. Se realiza una vigilancia tecnológica en tres temas prioritarios: software para la programación de la herramienta, piezas o robots modelados, software para modelar las piezas faltantes.
2. Ideación: Se desarrolla el concepto integral de diseño de software, se bosqueja la parte visual y los menús, además se establecieron los requerimientos de interacción con el usuario y las posibles salidas con sus métricas, además de las configuraciones de los robots.
3. Prototipado: Se realiza la programación, se modelan las partes de los robots faltantes y se realizan las animaciones finales. Luego se hace tanto la prueba del software como la validación del prototipo con los usuarios. En todas las etapas se proponen cambios, además de compartir las historias y experiencias aprendidas en el proceso de desarrollo de la solución.

3. Resultados

3.1. Etapa 1

Inicialmente se realizó un inventario de las características de los robots Lego Mindstorms EV3 y del VEX EDR, en cuanto a sensores y actuadores, los cuales fueron recopilados por los estudiantes de ambas instituciones al inicio del curso.

A medida que los estudiantes adquirieron las competencias necesarias en el manejo de las plataformas robóticas se les presentaron ambos escenarios y se les preguntó cuáles eran las características relevantes que un robot debería tener para cumplir una misión cualquiera en cada escenario.

Luego, con ayuda de los docentes, se agruparon estas características en categorías y se eliminaron las que eran similares o repetidas, quedando las categorías descritas en la tabla 2.

Al inicio del curso los estudiantes realizaron un inventario de los elementos que conforman a las plataformas LEGO EV3 y VEX; posteriormente identificaron las características y elementos comunes a ambas plataformas y se eligieron las más relevantes. Estas etapas se describen en la figura 1.



Figura 1 – Etapas contempladas en el proyecto.

A cada bloque asignado en el diseño de experimentos se le pidió que asignaran grados de importancia o pesos a los deseos del cliente, por tipos de escenarios (interiores y exteriores), tomando como valores 0 a 5. Luego se realizó el promedio de estos valores, dando como resultado dos grupos de deseos del cliente priorizados, uno para robots que se mueven en interiores y otro para robots que se mueven en exteriores (tabla 4).

No.	Requerimientos del Cliente	Interiores	Exteriores
1	Que sea fácil de manipular	5	4
2	Que tenga una interfaz amigable para darle órdenes	5	4
3	Que cumpla su tarea con una orden inicial	3	5
4	Que pueda elegirse el modo de hacer la tarea (customización)	3	2
5	Que se mueva en cualquier dirección	4	3
6	Que evada obstáculos	3	4
7	Que se mueva de manera autónoma por un tiempo prudente	5	5
8	Que se mueva en exteriores	0	5
9	Que se mueva en interiores	5	0
10	Que reaccione con la luz	1	4
11	Que responda a los colores	4	1

Tabla 4 – Pesos promedio de los requerimientos del cliente por cada tipo de escenario.

Esta información de pesos es el insumo inicial de la matriz de diseño mecatrónico QFD. Al final, a cada bloque se le presentaron dos posibles escenarios, uno de exteriores y otro de interiores, cada uno con un robot Lego Mindstorm EV3 y otro VEX que den respuesta a los requerimientos planteados desde el inicio.

Luego de ingresar los deseos del cliente, se ingresan los requerimientos técnicos en la matriz QFD. Posteriormente se obtienen las siguientes gráficas (figuras 2 y 3), con los pesos promedio de los requerimientos técnicos para cada configuración (tablas 5 y 6):

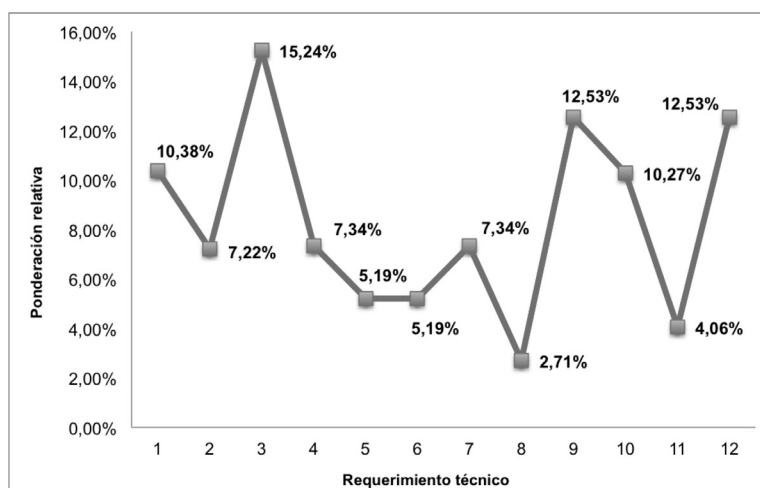


Figura 2 – Ponderación relativa (interiores).

No.	Ponderación relativa	Criterio Técnico
3	15,24%	Baterías recargables
9	12,53%	Monitoreo por computadora en tiempo real
12	12,53%	Programación in situ (en el mismo robot)
1	10,38%	Ruedas (conf. diferencial)
10	10,27%	Programación mediante software por computador
7	7,34%	Girosensor
4	7,34%	Sensor de color
2	7,22%	Orugas (2 orugas)
5	5,19%	Sensor ultrasónico
6	5,19%	Sensor <i>push- button</i>
11	4,06%	Sensor de luz
8	2,71%	Actuador con motor mediano

Tabla 5 – Ponderación relativa (interiores).

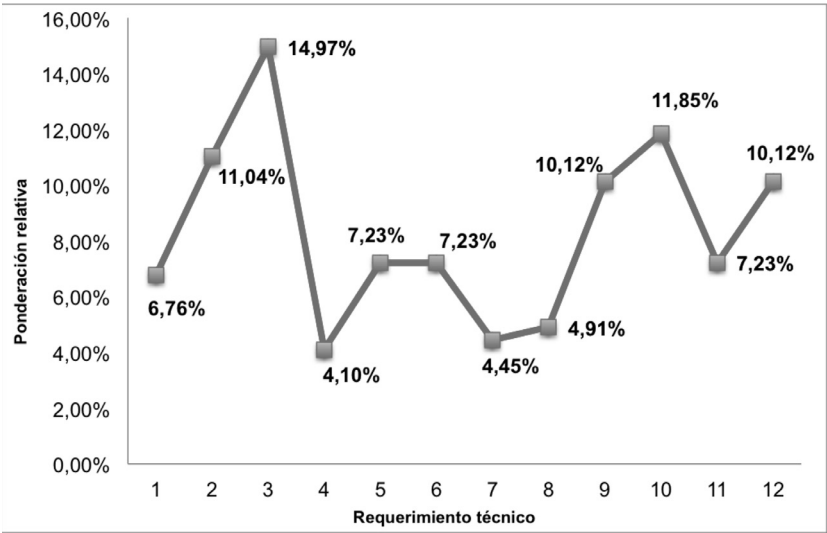


Figura 3 – Ponderación relativa (exteriores).

No.	Ponderación relativa	Criterio Técnico
3	14,97%	Baterías recargables
10	11,85%	Programación mediante software por computador
2	11,04%	Orugas
9	10,12%	Monitoreo por computadora en tiempo real
12	10,12%	Programación in situ (en el mismo robot)

No.	Ponderación relativa	Criterio Técnico
5	7,23%	Sensor ultrasónico
6	7,23%	Sensor <i>push-button</i>
11	7,23	Sensor de luz
1	6,76%	Ruedas (conf. diferencial)
8	4,91%	Actuador con motor mediano
7	4,45%	Girosensor
4	4,10%	Sensor de color

Tabla 6 – Ponderación relativa (exteriores).

3.2. Etapa 2

1. Inspiración: Luego de modelar el problema y la vigilancia tecnológica, se elige Unity para realizar la herramienta y se encuentran pocas piezas modeladas para los robots. Se eligieron *Solidworks®* e *Inventor®* para modelar las piezas faltantes, cuyo resultado se observa en la Figura 4.
2. Ideación: Se realiza el bosquejo para los escenarios interiores y exteriores (Figura 5) y se establecen las preguntas que se le hacen al jugador, junto con las respuestas y sus métricas. Se elige para interiores las configuraciones de robots Lego “Base motriz” y VEX EDR “Clawbot” (Figura 6). Para exteriores se realiza la configuración “tanque” de Lego y para VEX EDR se usa “Clawbot” tomando como variante el desplazamiento por orugas.
3. Prototipado: Se realizó la programación y se modelaron las partes de los robots faltantes, al igual que las animaciones finales. Luego se hizo un testeo del software y luego el prototipo se validó funcionalmente con un grupo de estudiantes de ambas instituciones.

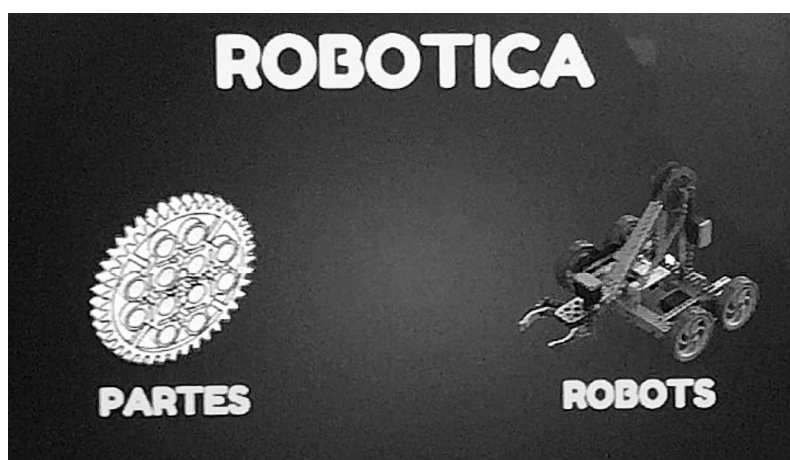


Figura 4 – Menú interactivo del Software.



Figura 5 – Escenario de interiores para robot VEX.

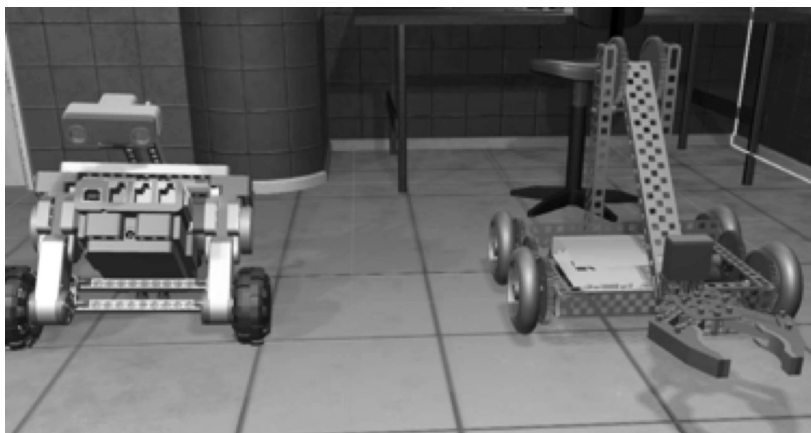


Figura 6 – Configuraciones de robots para interiores. Lego y VEX.

4. Discusión

Es importante aclarar que la metodología de despliegue de la función de calidad QFD es un proceso secuencial, el cual debe ser preparado y planeado mediante pesos y relaciones. Por ejemplo, luego de que se establecieron los requerimientos iniciales del cliente mediante las preguntas, se estableció el grado de relación que existe entre cada requerimiento del cliente versus cada requerimiento técnico, usando una escala ponderada no lineal para enfatizar claramente la importancia de los valores. Los grados de relación se clasificaron en relación fuerte, moderada y débil, con valores de 9, 3 y 1 respectivamente. Si no existía relación, simplemente el campo se dejó en blanco. Además, se estableció una relación fuerte, débil o nula entre cada par de características técnicas, es decir, de que manera influía una característica técnica en otra. Todo esto es propio de la metodología QFD y debe realizarse por expertos conocedores de las

herramientas robóticas. En este caso, estos valores fueron elegidos por los docentes de las materias de robótica de ambas instituciones.

Se observa en los resultados del escenario interiores que la ponderación relativa más alta corresponde a las baterías recargables, por lo tanto, es un requisito esencial para tener en cuenta en la configuración de los robots Lego y VEX. Entre mas baterías de repuesto posea el robot, mas autonomía tendrá. También se aprecia que el movimiento debe ser por ruedas, más que por orugas y que los sensores principales que deben tener son el giroscopio y el sensor de color. El giroscopio puede ser útil en interiores donde el espacio es conocido y por lo tanto se podrían programar sus movimientos más precisamente. También está la opción de incorporar el sensor ultrasónico, en la medida en que el entorno en interiores no sea completamente conocido.

En el escenario de exteriores, la ponderación relativa más alta corresponde a las baterías recargables, por lo tanto, es un requisito esencial en la configuración de los robots Lego y VEX también para esta configuración. Se observa que el movimiento debe ser por orugas, más que por ruedas, lo cual es compatible con el movimiento en exteriores en terrenos difíciles o desconocidos. Además, los sensores principales relevantes son el sensor ultrasónico y el sensor *push-button*, teniendo en cuenta que el robot debe evadir obstáculos en un entorno desconocido. También está la opción de incorporar el sensor de luz, en la medida en que el entorno en exteriores tenga poca luz día.

Se tiene planeado repetir el experimento, esta vez teniendo en cuenta factores externos que se puedan medir en el inicio, como el tipo de robot o el grado de escolaridad de la unidad experimental, debido a que es diferente para los estudiantes de ambas instituciones, pues se quiso tener primero un referente, debido a que los grupos tenían el mismo grado de conocimiento respecto a los robots. Además, se estudia la posibilidad de incorporar elementos de realidad virtual en la herramienta. Sería interesante realizar el experimento con estudiantes que no hayan usado previamente las herramientas de robótica educativa y conocer las diferentes configuraciones que pueden realizar. De la misma manera, se contempla incluir en el desarrollo la posibilidad de dar cumplimiento a los requerimientos haciendo uso de plataformas robóticas adicionales de acuerdo con la disponibilidad de las mismas.

Es pertinente aclarar que, aunque se encontraron aplicaciones de software relacionadas con el funcionamiento y simulación de diferentes tipos de robot, no se encontraron herramientas de software de diseño mecatrónico para robótica móvil. Un referente es el programa V-rep (<http://www.coppeliarobotics.com>) el cual permite simulaciones de cada una de las piezas que forman un robot, como motores, orugas, hélices o patas, se puede controlar el movimiento del robot y configurar sus sensores como si se tratase de una máquina real, pero aprender su manejo requiere un tiempo considerable y no utiliza herramientas de diseño mecatrónico para la selección de los accesorios de cada robot. Una comparación de diferentes tipos de software de robots lo hace el grupo de Robótica Educativa de la Universidad de Alcalá (<http://www.roboticaeducativa.org>) en donde identifican diferentes características de algunos de los softwares mas populares en robótica, pero no se halló tampoco referencia a la incorporación de herramientas de diseño mecatrónico en estas aplicaciones.

La identificación, aplicación y validación de diferentes herramientas de diseño mecatrónico mediante el uso de la robótica pedagógica para solucionar una necesidad, se convierte en el punto de partida de una educación que verdaderamente logre un impacto y un cambio de mentalidad, aprovechando los escenarios y la infraestructura existentes. Esto permitirá identificar las fortalezas, gustos e intereses de los educandos, adquiriendo las competencias necesarias para que puedan continuar con su formación en niveles superiores o realizar trabajos relacionados con la ciencia y la tecnología que le sean útiles a la comunidad.

El hecho de que los docentes de ambas instituciones estén utilizando técnicas de diseño centradas en lo humano para entender mejor a sus estudiantes, hace que los participantes se sientan comprometidos a cambiar su ambiente de aprendizaje. Además, en la medida que el docente se convierta en el diseñador de su propia experiencia en el aula, profesionaliza su rol y permite un cambio valioso y autentico, impulsado por las necesidades de los estudiantes, en vez de hacerlo por un mandato administrativo.

Referencias

- Acuña, A. (2004). Robótica y aprendizaje por Diseño. *Revista La Educ@ción de la OEA (Organización de los Estados Americanos)*, 139, 1–12. <http://www.educoas.org/portal/bdigital/lae-ducacion/139/pdfs/139pdf7.pdf>
- Bravo, F., & Forero, A. (2012). La robótica como un recurso para facilitar el aprendizaje y desarrollo de competencias generales. *Teoría de la Educación: Educación y Cultura en la Sociedad de la Información*, 13(2), 120–136. Recuperado de: <http://www.redalyc.org/articulo.oa?id=201024390007>
- Brown, T. (2008). Design Thinking. *Harvard Business Review*, 86(9), 62–72.
- Cabello-Ochoa, S., & Carrera-Farran, F. X. (2017). Diseño y validación de un cuestionario para conocer las actitudes y creencias del profesorado de educación infantil y primaria sobre la introducción de la robótica educativa en el aula. *Electrónica de Tecnología Educativa*, 60, 1–22. doi: 10.21556/edutec.2017.60.871
- Carvajal, J. (2013). Revisión y análisis de diseño mecatrónico para diseño curricular transdisciplinario de programas de ingeniería multidisciplinares. *Scientia et Technica*, 18(1), 86–94.
- Chetty, J. (2015). Lego Mindstorms: Merely a toy or a powerful pedagogical tool for learning computer programming?. In *Proceedings de la 38 Australasian Computer Science Conference ACSC '15*, 111–118. Recuperado de <https://pdfs.semanticscholar.org/e04f/23c697c48013ae2bd1c4c485ad015163fc34.pdf>
- Cooke, J. (2012). *Everything you want to know about agile*. Ely, Cambridgeshire, UK: IT Governance Publishing.
- Corchuelo, M. (2015). Propuesta de lineamientos para el desarrollo de ambientes de aprendizaje en robótica a través del estudio de experiencias (Tesis de Maestría). Universidad de la Sabana, Colombia. Recuperado de <http://hdl.handle.net/10818/20274>

- Correll, N., Wailes, C., & Slaby, S. (2014). A One-Hour Curriculum to Engage Middle School Students in Robotics and Computer Science, Using Cubelets. En: Ani Hsieh, M., & Chirikjian, G. (eds), *Distributed Autonomous Robotic Systems Springer Tracts in Advanced Robotics, vol 104* (pp. 165-176). doi: 10.1007/978-3-642-55146-8_12
- De Silva, C., & Behbahani, S. (2013). A design paradigm for mechatronic systems. *Mechatronics*, 23(8), 960–966. doi: 10.1016/j.mechatronics.2012.08.004
- Dode, P., Greig, M., Zolfaghari, S., & Neumann, W. (2016). Integrating human factors into discrete event simulation: a proactive approach to simultaneously design for system performance and employees' well being. *International Journal of Production Research*, 54(10), 3105–3117. doi: 10.1080/00207543.2016.1166287
- Dym, C., Agogino, A., Eris, O., Frey, D., & Leifer, L. (2005). Engineering Design Thinking, Teaching and Learning. *Journal of Engineering Education*, 94(1), 103–120. doi: 10.1002/j.2168-9830.2005.tb00832.x
- Edin Grimheden, M. (2013). Can agile methods enhance mechatronics design education? *Mechatronics*, 23(8), 967–973. doi: 10.1016/j.mechatronics.2013.01.003
- García, Y., & Reyes, D. (2012). Robótica educativa y su potencial mediador en el desarrollo de las competencias asociadas a la alfabetización científica. *Revista Educación y Tecnología*, 2, 42–55.
- Gausemeier, J., Dorociak, P., Pook, S., Nyßen, A., & Terflöth, A. (2010). Computer-aided cross-domain modeling of mechatronic systems. In *Proceedings of Design 2010, the 11th International Design Conference*, Dubrovnik, Croatia, 723-732. Recuperado de https://www.designsociety.org/download-publication/29417/computer-aided_cross_domain_modeling_of_mechatronic_systems
- Ghitis, T., & Alba, J. A. (2014). Los robots llegan a las aulas. *Infancias imágenes*, 13(1), 143–147.
- Gómez-González, C., Lecuona-López, M., & Hernández-Monsalve, M. (2016). Análisis de bibliografía sobre políticas de diseño. *Iconofacto*, 12(19), 268–293. doi: 10.18566/iconofact.v12.n19.a12
- Guzey, S., Moore, T., & Harwell, M. (2016). Building Up STEM: An Analysis of Teacher-Developed Engineering Design-Based STEM Integration Curricular Materials. *Journal of Pre-College Engineering Education Research (J-PEER)*, 6(1). doi: 10.7771/2157-9288.1129
- Hassan, A., Siadat, A., Dantan, J., & Martin, P. (2010). Conceptual process planning – an improvement approach using QFD, FMEA, and ABC methods. *Robotics and Computer-Integrated Manufacturing*, 26(4), 392–401. doi: 10.1016/j.rcim.2009.12.002
- Isermann, R. (2008). Mechatronic systems - Innovative products with embedded control. *Control Engineering Practice*, 16(1), 14–29. doi: 10.1016/j.conengprac.2007.03.010

- Kuhn, W. (2006). Digital Factory - Simulation Enhancing the Product and Production Engineering Process. In *Proceedings of the 2006 Winter Simulation Conference*, IEEE, 1899–1906. doi: 10.1109/wsc.2006.322972
- Kurdve, M., Zackrisson, M., Wiktorsson, M., & Harlin, U. (2014). Lean and green integration into production system models – experiences from Swedish industry. *Journal of Cleaner Production*, 85, 180–190. doi: 10.1016/j.jclepro.2014.04.013
- Macas, E. (2017). Definición y Estado del Arte de la Ingeniería Concurrente La Manufactura Integrada por Computador y la Mecatrónica. *INNOVA Research Journal*, 2(10), 44–60. doi: 10.33890/innova.v2.n10.2017.286
- Ministerio de Educación Nacional de Colombia (2008). Ser competente en tecnología: ¡Una necesidad para el desarrollo! *Serie Guías No. 30: Orientaciones generales para la educación en tecnología*. Recuperado de: https://www.mineducacion.gov.co/1621/articles-160915_archivo_pdf.pdf
- Mizell, S., & Brown, S. (2017). The Current Status of STEM Education Research 2013–2015. *Journal of STEM Education*, 17(4), 52–56. Recuperado de <https://www.jstem.org/jstem/index.php/JSTEM/article/view/2169/1815>
- Monsalves, S. (2019). Estudio sobre la utilidad de la robótica educativa desde la perspectiva del docente. *Revista de Pedagogía*, 32(90), 81–117.
- Moreno, I., Muñoz, L., Serracín, J., Quintero, J., Patiño, K., & Quiel, J. (2012). La robótica educativa, una herramienta para la enseñanza-aprendizaje de las ciencias y las tecnologías. *Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*, 13(2), 74–90.
- Odorico, A. (2005). La robótica desde una perspectiva pedagógica. *Revista de Informática Educativa y Medios Audiovisuales*, 2(5), 33–48.
- Organización de los Estados Americanos (2016). La indagación como estrategia para la educación STEAM. Red Regional sobre Educación en Ciencia, Tecnología, Ingeniería, Artes y Matemáticas – EducaSTEAM. Recuperado de <http://recursos.portaleducoas.org/sites/default/files/Final%20OEA%20Indagacio%CC%81n.pdf>
- Pessôa, M., & Trabasso, L. (2016). The Lean Product Design and Development Journey, 19–41. *Cham, Switzerland: Springer International Publishing*. doi: 10.1007/978-3-319-46792-4_2
- Pinto, M., Barrera, N., & Pérez, W. (2010). Uso de la robótica educativa como herramienta en los procesos de enseñanza. *Ingeniería Investigación Y Desarrollo*, 10(1), 15–23.
- Punz, S., Hehenberger, P., Follmer, M., & Zeman, K. (2011). Customer-oriented hierarchical concept development - application of the HoC approach in mechatronic design. In *2011 IEEE International Conference on Mechatronics*. doi: 10.1109/icmech.2011.5971285

- Rivadeneira, P. (2019). La robótica como una herramienta para facilitar el aprendizaje y desarrollo de las competencias STEM en los integrantes del equipo de robótica Pólux de la Institución Educativa Juan Nepomuceno Cadavid (Tesis de Maestría). Universidad EAFIT, Medellín, Colombia. Recuperado de: <http://hdl.handle.net/10784/12087>
- Sáez-Carreras, J., Bas-Peña, E., & Bedmar-Moreno, M. (2007). *Pedagogía social y educación social: historia, profesión y competencias*. Madrid: Pearson Educación.
- Salas-López, G., Sandoval-González, O., Herrera-Aguilar, I., Martínez-Sibaja, A., Portillo-Rodríguez, O., & Vilchis-González, A. (2012). Design and development of a planar robot for upper extremities rehabilitation with visuo-vibrotactile feedback. *Procedia Technology*, 3, 147–156. doi: 10.1016/j.protcy.2012.03.016
- Sánchez, M. (2003). Implementación de estrategias de robótica pedagógica en las instituciones educativas. In Blog Juan Carlos Sánchez Guerrero, Eduteka ICESI. Recuperado de: <http://www.oocities.org/es/juancarlosanchezguerrero/ger/ii/local/RoboticaPropuesta.pdf>
- Servicio Nacional de Aprendizaje (2015). Book Metodología de desarrollo de proyectos de base tecnológica. Bogotá: Sena Red Tecnoparque.
- Silva-Ortigoza, R., Márquez-Sánchez, C., Carrizosa-Corral, F., Hernández-Guzmán, V., García-Sánchez, J., Taud, H., Marciano-Melchor, M., & Álvarez-Cedillo, J.A. (2014). Obstacle Avoidance Task for a Wheeled Mobile Robot – A Matlab-Simulink-Based Didactic Application. MATLAB. *Applications for the Practical Engineer*, 79–102. doi: 10.5772/58392
- Steimer, C., Fischer, J., & Aurich, J. (2017). Model-based Design Process for the Early Phases of Manufacturing System Planning using SysML. In *Procedia CIRP*, 60, 163–168. doi: 10.1016/j.procir.2017.01.036
- Vallejo-Jiménez, M. (2018). Diseño e implementación de prototipo de robot Lego Mindstorms EV3 para apropiación del concepto de la luz. *Revista Tecnoacademia Con-ciencia y técnica*, 2, 12–16. Recuperado de: <http://revistas.sena.edu.co/index.php/conciencia/article/viewFile/1953/2072>
- Vallejo-Jiménez, M., Martínez-Puerta, J., Bedoya-Agudelo, S., & Salgado, N. (2018). SENA Tecnoacademia Risaralda and Caldas as a Collaborative Learning Scenario in Robotics. En: Cristóbal J. Nettle, Miguel A. Solis (eds), *Revista del 4th Congress on Robotics and Neuroscience*, 39–45. Recuperado de <https://crone2018.pubpub.org/pub/hdk7rpow>

La Semántica de las Imágenes y el Análisis de su Contenido

Catalina-Alejandra Vázquez-Rodríguez, Raúl Pinto-Elías

cvazquez@cenidet.edu.mx, rpinto@cenidet.edu.mx

Tecnológico Nacional de México/CENIDET, Depto. Ciencias Computacionales, Interior Internado Palmira S/N, Col. Palmira, C.P. 62490, Cuernavaca, Morelos, México.

DOI: 10.17013/risti.34.20-28

Resumen: Las imágenes transmiten información de diferentes maneras, la descripción puede ser referente a la forma de los objetos, colores, texturas; sin embargo, también pueden transmitir conceptos, los cuales estarán formados por el conjunto de objetos presentes en una imagen. En este trabajo se presenta un enfoque de descripción de imágenes desde la perspectiva semántica y se evalúa mediante un análisis léxico-sintáctico para comprobar que las descripciones dadas sean parte del lenguaje definido, por ello se utiliza un módulo de verificación sintáctica el cual resultó ser de gran utilidad dado que no siempre los datos de entrada pertenecerán al conjunto de trabajo, si bien es un modelo adaptable y con posibilidades de crecimiento, es necesario definir los datos a trabajar; por ello la verificación sintáctica evita añadir “ruido” al modelo para garantizar que las clases que entren sea una clase válida y pueda realizarse un refinamiento de las descripciones.

Palabras-clave: Semántica del contenido de imágenes; estructuración de la información; análisis léxico del contenido de imágenes.

The Semantics of Images and the Analysis of their Content

Abstract: the images transmit information in different ways, the description can be referring to the shape of objects, colors, textures; however, they can also convey concepts, which will be formed by the set of objects present in an image. In this work, an image description approach is presented from the semantic perspective and is evaluated through a lexical-syntactic analysis to verify that the descriptions given are part of the defined language, so a syntactic verification module is used which turned out to be very useful since the input data will not always belong to the work set, although it is an adaptable model and with growth possibilities, it is necessary to define the data to work; Therefore, the syntactic verification avoids “noise” to the model to ensure that the classes that enter are a valid class and a refinement of the descriptions can be made.

Keywords: Coupling; flexibility; interface inheritance; implementation inheritance.

1. Introducción

La descripción de las imágenes es una tarea que ha intentado resolverse computacionalmente por muchos años, dicha tarea muchas veces es confundida con el reconocimiento de objetos (Mottaghi et al., 2014), el cual consiste en identificar los elementos y objetos de las imágenes (Álvarez, Salzmann, & Barnes, 2014), mientras que la descripción de las imágenes va enfocada a describir el contenido de ellas como un todo, es decir, interpretar la escena presente (Xu, Schwing, & Urtasun, 2014).

La descripción no ha sido una tarea fácil de realizar a lo largo de los años, se han realizado desde descripciones de bajo nivel tales como análisis del color, textura y forma (Agpal, Singh, Kaleka, & Sharma, 2012); sin embargo, se detectó que esto no bastaba y se continuó el trabajo describiendo las imágenes basándose en el modelo (Content-Based Image Retrieval) en la década de 1990. Mediante este modelo se incorporaba exclusivamente el código lingüístico (Gudivada & Raghavan 1995) y la tarea de recuperación se reducía esencialmente a la búsqueda de las palabras clave empleadas al describir imágenes (Zhou, Li, & Tian, 2017).

Posteriormente al ver las limitaciones con el uso de etiquetas para las descripciones de imágenes, se comenzó a trabajar la denominada recuperación de información visual basada en la semántica (Álvarez et al. 2014). Este enfoque se caracteriza, por la unión de la descripción del contenido visual como colores, texturas formas etc. y la descripción lingüística en una imagen dada de la descripción humana como vendrían siendo algunos etiquetados, Ground Truth, etc. (Zhao, Shi, & Wang 2017).

La descripción semántica de imágenes mediante la recuperación de información visual basada en la semántica resultó ser una mejora al igual que los modelos ontológicos (Kong, Lin, Bansal, Urtasun, & Fidler, 2014), pero en ocasiones, las descripciones podrían tener errores sintácticos de las etiquetas descriptivas, lo que llevaría a un erróneo procesamiento de la información.

Al profundizar en el campo de las propiedades de las imágenes, más aumenta la complejidad de la traducción o interpretación de la relación entre los atributos intrínsecos y objetos de una imagen (Cruz, Pérez, & Cantero, 2009). Los seres humanos expresan con relativa facilidad los atributos de alto nivel incluyendo las propiedades semánticas, al observar una imagen es muy sencillo interpretar amor, tristeza, calor etc., gracias al empleo de las palabras; en cambio, la dificultad va aumentando conforme trata de expresar con palabras los atributos de más bajo nivel como lo son el color, forma, textura; eso provoca finalmente una disfunción en el momento de la recuperación de imágenes.

A continuación, se describe el contenido de este trabajo. En la sección 2 se encuentra la información de los datos con los que se trabajó, en la sección 3 el estado del arte, en la sección 4 la estructura del sistema, en la sección 5 la experimentación y resultados, en la sección 6 las conclusiones, en la sección 7 trabajo futuro, y finalmente las referencias.

2. Datos y clases

En el presente trabajo se realizan descripciones semánticas del contenido de las imágenes. Se utilizó el banco Pascal VOC 2012, el cual es un banco de imágenes muy

popular, para construir y evaluar algoritmos para la clasificación de imágenes, detección de objetos y segmentación. Pascal VOC agrupa sus imágenes de tres formas: el primer grupo corresponde a las imágenes originales, el segundo grupo corresponde a los Ground Truth, y el tercer grupo corresponde a la clasificación de las clases existentes en las imágenes.

Se trabajó con 5 categorías y 26 subcategorías las cuales se muestran en la Tabla 1.

Categorías	Subcategorías
<i>Personas</i>	Adulto, Niño, Bebé
<i>Animales</i>	Gato, Vaca, Perro, Ave, Caballo, Oveja
<i>Transporte</i>	Avión, Bicicleta, Bote, Autobús, Coche, Motocicleta, Tren
<i>Objetos</i>	Botella, Silla, Mesa, Maceta, Sofá, Pantalla
<i>Entorno</i>	Construcciones, Cielo, Pavimento, Vegetación

Tabla 1 – Categorías y subcategorías trabajadas para la descripción semántica y sintáctica del contenido de imágenes.

3. Estado del arte

Para describir imágenes es necesaria una explicación de los elementos de las imágenes. Son muy extensas las posibilidades de descripción, desde los atributos visuales básicos como el color, composición, textura, distribución de los elementos que aparecen en ella, hasta los sentimientos subjetivos, representación de una ideología o pensamiento.

La descripción tiene como misión representar numérica o textualmente las características o propiedades de las imágenes que ingresan para formar el banco de datos, estas pueden ser propiedades intrínsecas y extrínsecas.

Las propiedades intrínsecas de la imagen corresponden a rasgos visuales que caracterizan toda la imagen como su color, textura, forma y las relaciones espaciales; a este tipo de características suelen denominarse descripciones de bajo nivel (Civelek, Lüy, & Mamur, 2017). Las propiedades extrínsecas de la imagen corresponden a todo lo contenido en la imagen no propiamente visual, estas propiedades están divididas en nivel medio y nivel alto. En el nivel medio se trabaja la detección automática de límites, contornos, objetos como rostros, sillas, coches etc. y de conceptos extraídos de la imagen como la identificación de día o de noche, si es verano o invierno.

En el nivel alto se trabajan los elementos que se incluyen en los metadatos como autor, título, localización geográfica, fecha, formato o propiedades de carácter subjetivo denominadas semánticas extraídas a raíz de la contemplación de la imagen y dando una interpretación de ellas, las cuales suelen incorporarse en el apartado de descripciones o notas en los metadatos (Iancu, 2018).

Los sistemas de recuperación de imagen pueden representar de manera automática y con facilidad las características de bajo nivel o propiedades intrínsecas de las imágenes, pues estos atributos son inherentes a la imagen. El problema se complica cuando se aborda la representación automática de las propiedades extrínsecas, describir el concepto de frío,

rostro, mascota en términos de color, no es tarea sencilla, y mucho más complejo sería expresar mediante estos atributos de bajo nivel un concepto subjetivo como el amor, la alegría, el dolor o la intranquilidad. En consecuencia, al profundizar en el campo de las propiedades de la imagen, más aumenta la complejidad de la traducción o interpretación de la relación entre los atributos intrínsecos y objetos de una imagen (Xu et al., 2014).

Los seres humanos expresan con relativa facilidad los atributos de alto nivel incluyendo las propiedades semánticas, al observar una imagen es muy sencillo interpretar amor, tristeza, calor etc., gracias al empleo de las palabras; en cambio, la dificultad va aumentando conforme trata de expresar los atributos de más bajo nivel como lo son el color, forma, textura; eso provoca finalmente una disfunción en el momento de la recuperación de imágenes cuando se realiza una búsqueda.

Si se desea expresar una consulta o realizar una búsqueda mediante un código textual, y las descripciones de las imágenes están expresadas mediante un código visual, se crea una brecha, este fenómeno recibe el nombre de vacío semántico. Para eliminar o reducir este vacío semántico es necesario favorecer la equiparación entre el código visual y las correspondientes propiedades de alto nivel mediante técnicas de retroalimentación entre otras para mejorar la respuesta del sistema (García & Cillán, 1998).

La inteligencia artificial (IA) no sólo se ocupa de mecanismos generales relacionados con la búsqueda de soluciones en un espacio dado, o de cómo representar y utilizar el conocimiento de un determinado dominio de discurso. Otro aspecto, es el que corresponde a los mecanismos y/o procesos inferenciales, que consideraremos como el punto de partida de los llamados modelos de razonamiento. En cualquier dominio, la propagación del conocimiento (Xu et al., 2014), por medio de programas de Inteligencia artificial se efectúa siempre siguiendo un modelo de razonamiento bien definido. Estos modelos de razonamiento forman parte del motor de inferencias si se habla de sistemas de producción, o de las estructuras de control del conocimiento, si se habla de cualquier otro tipo de sistemas de IA, y contribuyen de manera decisiva a organizar correctamente la búsqueda de soluciones.

Pueden plantearse diversas maneras de representar el conocimiento para posteriormente lograr descripciones semánticas, mediante, ontologías, lógica de primer orden, lógica difusa etc. (Ducrot, 2000). La representación del conocimiento y el razonamiento juegan un papel central en la Inteligencia Artificial. La investigación en IA comenzó tratando de identificar los mecanismos generales responsables del comportamiento inteligente, rápidamente se hizo evidente que los métodos generales y poderosos no son suficientes para obtener el resultado deseado, es decir, el comportamiento inteligente (Martínez, 2013).

Casi todas las tareas que puede realizar un ser humano que se considera que requieren inteligencia también en ocasiones se basan en una gran cantidad de conocimiento, por ejemplo, la mayoría de las declaraciones humanas son ambiguas, y esta ambigüedad es una característica esencial, no sólo del lenguaje, sino también de los procesos de clasificación, del establecimiento de taxonomías y jerarquías, y de los procesos de razonamiento en sí mismos (Álvarez et al., 2014).

El tema de las descripciones semánticas se ha abordado mediante diversos enfoques a lo largo de los años, sin embargo, sigue siendo un problema sin resolver por

completo, es sencillo describir los elementos que conforman una imagen, pero interpretar la escena dados los elementos es una tarea más compleja (Pérez & Merino 2018).

En la Figura 1 se muestra la importancia de la ubicación de los objetos dentro de una imagen, donde, se puede apreciar qué en ambas imágenes se encuentran los mismos objetos, la diferencia radica en el lugar que se encuentran cada uno de ellos lo que modifica como se relacionan.



Figura 1 – Diagrama del proceso de descripción y verificación semántica.

4. Estructura del sistema

Cuando entran los datos pertenecientes a los Ground Truth de Pascal VOC se realizan 4 pasos. En el primer paso se extraen los nombres de las clases presentes en la imagen, en el segundo se realiza una verificación sintáctica de las clases para evitar añadir ruido al sistema, en el tercero se enumeran la cantidad de objetos detectados de cada clase, en el cuarto se calculan las coordenadas x,y de cada objeto y finalmente se obtienen descripciones semánticas de la escena de la imagen y algunas inferencias. En la Figura 2 se muestra el diagrama del proceso de descripción semántica de imágenes.

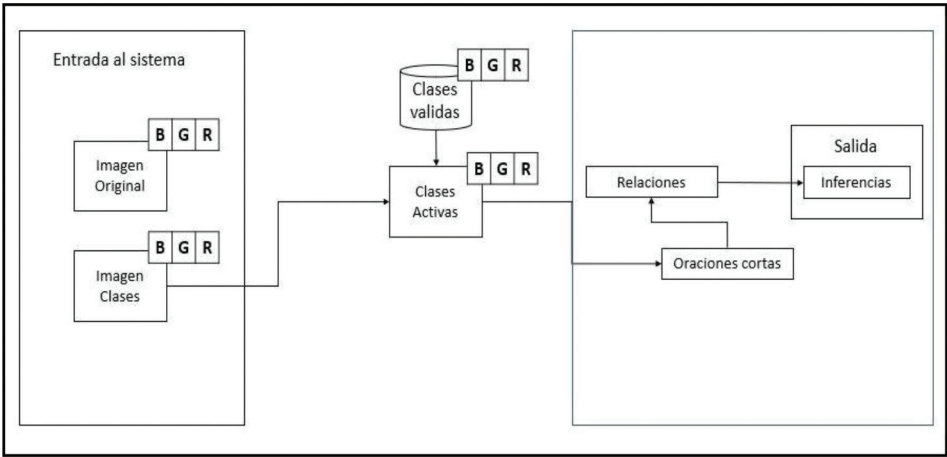


Figura 2 – Diagrama del proceso de descripción y verificación semántica.

5. Experimentación y resultados

La experimentación se llevó a cabo con imágenes del banco Pascal VOC 2012, se utilizaron los originales y los Ground Truth, los cuales muestran a detalle donde se ubican los elementos descritos de las imágenes, por lo cual se puede obtener la ubicación de cada objeto con precisión y de esta manera saber si un objeto se encuentra, sobre, al lado, debajo etc. lo cual apoya a la descripción y a las inferencias que pudieran realizarse sobre la escena de la imagen. En este trabajo las descripciones fueron variantes en el aspecto de algunas más detalladas que otras, pues la descripción depende directamente de la cantidad de objetos contenidos en las imágenes. En la Figura 3a se muestra la imagen original, en la Figura 3b el Ground Truth de la original, en la Figura 3c la descripción de su contenido y finalmente, en la Figura 3d el análisis sintáctico del contenido mediante Lex-Bison para verificar que las clases son las definidas; así como su correcta escritura para evitar errores ortográficos, de singular-plural, etc.

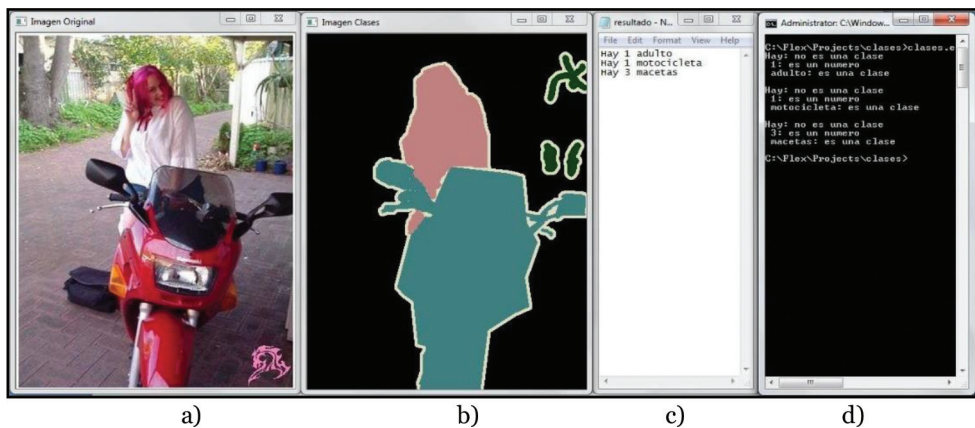


Figura 3 – Imagen de adulto en moto, su *Ground Truth*, la descripción semántica y verificación sintáctica.

Como se puede apreciar en la Figura 3a se tiene una persona adulta, una moto y vegetación. En la Figura 3b que corresponde al Ground Truth es posible ver fácilmente de color verde los objetos “macetas” contenidos en la imagen original, posteriormente en la Figura 3c, el algoritmo de descripción de contenido semántico de imágenes realizó su trabajo enlistando los elementos y la cantidad de objetos que están dentro del conjunto de clases con las que se está trabajando. Finalmente, en la Figura 3d se realiza el análisis sintáctico de las sentencias descriptivas dadas por el sistema de descripción semántica de imágenes, donde verifica si cada palabra es parte de la gramática creada correspondiente a las clases con las que se trabaja. La Figura 4 muestra una escena de casa con dos adultos y un bebé.

En la Figura 4a, se tiene la imagen original tomada del banco Pascal VOC 2012 al igual que su correspondiente Ground Truth de la Figura 4b. En la Figura 4c, se muestra la descripción de los elementos y se puede observar como el sistema es capaz de diferenciar

entre una persona adulta y una persona pequeña (niño, bebé); en este caso, se trata de un bebé y en la Figura 4d su verificación sintáctica.

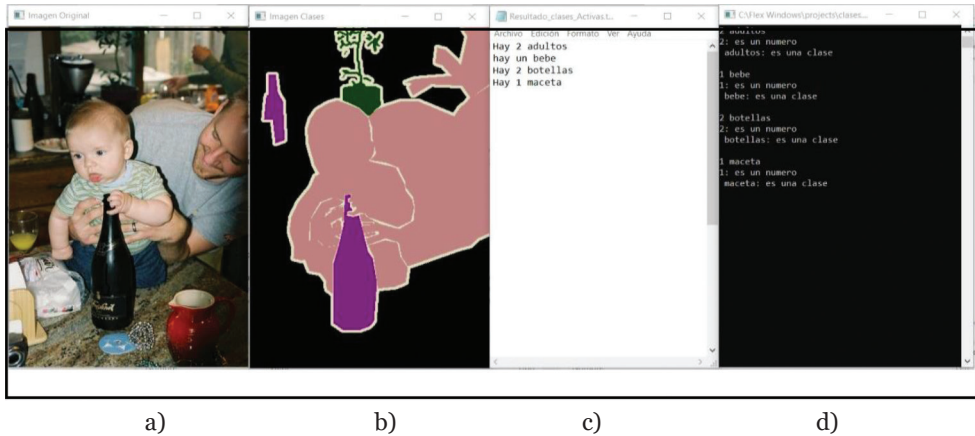


Figura 4 – Imagen de familia en casa, su Ground Truth, la descripción semántica y verificación sintáctica.

En la Figura 5 se muestra una escena de una persona montando un caballo.

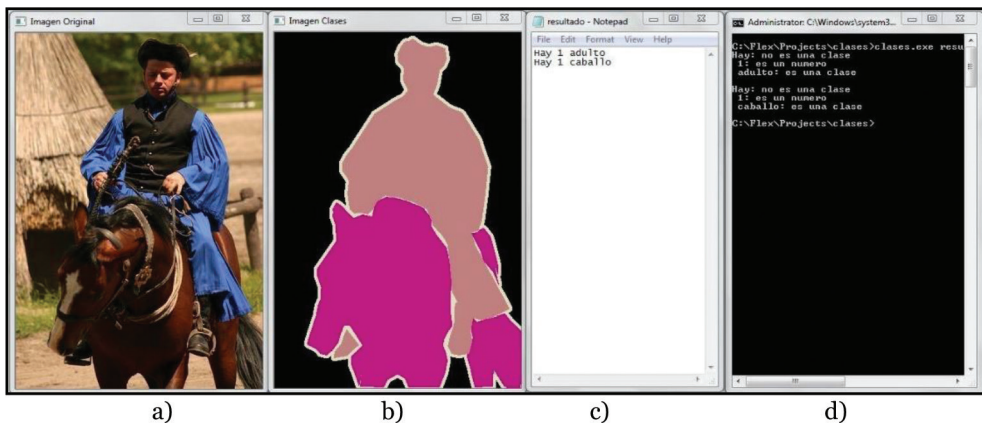


Figura 5 – Imagen de persona montando un caballo. Su Ground Truth, la descripción de los elementos de la imagen acorde a las clases definidas y su verificación sintáctica.

La Figura 5a corresponde a la original, la Figura 5b al Ground Truth, la Figura 5c la descripción de su contenido y finalmente en la Figura 5d, al análisis sintáctico de las descripciones. En los Ground Truth, se puede apreciar que el color de las clases se mantiene, para la clase “persona” se utiliza el color rosa claro.

Los resultados obtenidos muestran que al tomar en cuenta las imágenes como un todo y no como objetos aislados se puede obtener información importante para describir la imagen como un todo, incluso podrían realizarse inferencias sobre lo que sucede en la escena que conforman la imagen, por ello es muy importante la verificación sintáctica ya que de esta manera se evitó añadir ruido al sistema y se trabajó solo con elementos válidos.

6. Conclusiones

El detalle de la descripción de las imágenes cambia entre una y otra, esto debido a que, en algunos casos las imágenes contarán con una cantidad mayor de objetos con respecto de otras. La riqueza de las descripciones será mayor cuando haya más objetos en las imágenes, cuando tengan visibles detalles del entorno tales como: vegetación, cielo, edificios, etc. Se logró que el sistema haga una diferenciación de las etapas de la clase “persona”, es decir, si se trata de un adulto o de un bebé se considera la diferencia ya que son derivaciones de dicha clase y al tener la información estructurada en árboles jerárquicos se logra obtener ese detalle de las subcategorías. Finalmente, se ha mostrado la validación sintáctica la cual realiza una verificación de las clases para solo procesar las válidas y de esta manera evitar añadir ruido al sistema con palabras que no son clases reconocidas tales como: conectores, numerales, etc.

7. Trabajo futuro

El sistema de descripciones y análisis semántico de las imágenes se seguirá complementando para que sea capaz de dar descripciones más naturales como en el caso de la Figura 3, en lugar de enunciar los elementos que aparecen, sea posible realizar interpretaciones a manera que pueda decirse que se está montado un caballo.

El entorno de la imagen será tomado en cuenta para describir si se trata de una imagen de interior, exterior, rural o urbano, así como inferir si esta frío, si es noche, día, etc. Finalmente se considera realizar una estructura de almacenamiento de la información que permita acceder y a diferentes niveles de la información, para de esta manera tener la posibilidad de crear descripciones desde diferentes perspectivas.

Agradecimientos

Al CONAcYT por su apoyo para la realización de este trabajo enmarcado en el Doctorado en Ciencias de la computación en Tecnológico Nacional de México/ CENIDET.

Referencias

- Agpal, J., Singh, J., Kaleka, S., & Sharma, R. (2012). Different Approaches of CBIR Techniques. *International Journal of Computers & Distributed Systems*, 1(2), 76–78.
- Álvarez J.M., Salzmann M., & Barnes N. (2014). Large-scale semantic co-labeling of image sets. In *IEEE Winter Conference on Applications of Computer Vision*, Steamboat Springs, CO, USA, pp. 501-508. IEEE. doi: 10.1109/WACV.2014.6836060.

- Civelek, Z., Lüy, M., & Mamur, H. (2017). A new fuzzy logic proportional controller approach applied to individual pitch angle for wind turbine load mitigation. *Renewable Energy*, 111, 708–717.
- Cruz, M. N., Perez, M. P., & Cantero, T. C. (2009). Influencia de la motivación intrínseca y extrínseca sobre la transmisión de conocimiento. *CIRIEC-España, Revista de Economía Pública, Social y Cooperativa*, 187–211.
- Ducrot, O. (2000). La elección de las descripciones en semántica argumentativa léxica. *Revista iberoamericana de discurso y sociedad*, 2(4), 23–44.
- García, M., & Cillán, J., (1998). Las señales de calidad: atributos extrínsecos del producto. *Revista anual de estudios económicos y empresariales*, 81–116.
- Gudivada, V. N., & Raghavan, V. V. (1995). Content based image retrieval systems. *Computer*, 28(9), 18–22. doi: 10.1109/2.410145.
- Iancu, I. (2018). Heart disease diagnosis based on mediative fuzzy logic. *Artificial Intelligence in Medicine*, 89, 51–60. doi: 10.1016/j.artmed.2018.05.004.
- Kong, C., Lin, D., Bansal, M., Urtasun, R., & Fidler, S. (2014). What Are You Talking About? Text-to-Image Coreference. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, pp. 3558–3565. doi: 10.1109/CVPR.2014.455.
- Martínez, C. J. (2013). La recuperación automatizada de imágenes: retos y soluciones. *Revista General De Información Y Documentación*, 23(2), 423–436. doi.org/10.5209/rev_RGID.2013.v23.N2.43137.
- Mottaghi, R., Chen, X., Liu, X., Cho, N. G., Lee, S. W., Fidler, S., & Yuille, A. (2014). The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 891–898. IEEE. doi.org/10.1109/CVPR.2014.119.
- Pérez, P., & Merino, M. (2018). Definición de semántica. Recuperado a partir de: <https://definicion.de/semantica>.
- Xu, J., Schwing, A.G., & Urtasun, R. (2014). Tell Me What You See, and I Will Show You Where It Is. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3190–3197.
- Zhao, H., Shi, J., & Wang, X. (2017). Pyramid Scene Parsing Network. In *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp.6230–6239. IEEE. doi:10.1109/CVPR.2017.660.
- Zhou, W., Li, H., & Tian, Q. (2017). Recent Advance in Content-based Image Retrieval: A Literature Survey. *International Journal Advanced Networking and Applications*, 10(1), 3741–3757.

Descubrimiento de Conocimiento en Historias Clínicas mediante Minería de Texto

Ana Isabel Oviedo Carrascal¹, David Sanguino Cotte², Natalia Andrea Restrepo Arango¹, Andrés Felipe Patiño Vélez¹

ana.oviedo@upb.edu.co, david.sanguino@sanvicentefundacion.com, nataliaandrear@gmail.com, patino.andres@gmail.com

¹ Universidad Pontificia Bolivariana, Circular 1 No. 70-01, 050031, Medellín, Colombia

² Hospital San Vicente Fundación, Calle 64 No. 51D-154, 050010, Medellín, Colombia

DOI: 10.17013/risti.34.29-43

Resumen: Las instituciones clínicas presentan una alta generación de datos no estructurados tanto en el registro de procedimientos en texto libre por parte del personal médico, como por las imágenes y videos generados por las ayudas diagnósticas. En este trabajo se plantea un proceso de descubrimiento de conocimiento en el texto no estructurado de las historias clínicas del área de traumatología del Hospital San Vicente Fundación mediante minería de texto. Para el estudio se aplicaron técnicas de preparación de texto como eliminación de palabras no relevantes, sustitución de términos, eliminación de acentos y derivación de palabras. Respecto a los procesos de minería se aplicaron técnicas de aprendizaje supervisado y no supervisado como árboles de decisión, regresión logística, k-vecinos más cercanos, clustering jerárquico y reglas de asociación. El resultado obtenido es la conformación de un modelo de las palabras más relevantes en los registros clínicos del Hospital en el área de traumatología.

Palabras-clave: Minería de texto; minería de datos de la salud; procesamiento del lenguaje natural.

Knowledge Discovery in Medical Records through Text Mining

Abstract: The clinical institutions generate a large amount of unstructured data both in the registration of procedures in free text by medical staff, and by the images and videos generated by diagnostic aids. This paper proposes a process of knowledge discovery in the unstructured text of the medical records of the trauma area of the San Vicente Foundation Hospital through text mining. Text preparation techniques were applied such as elimination of non-relevant words, substitution of terms, elimination of accents and derivation of words. Regarding mining processes, supervised and unsupervised learning techniques were applied such as decision trees, logistic regression, nearest k-neighbors, hierarchical clustering and association rules. The result obtained is the conformation of a model of the most relevant words in the clinical records of the Hospital in the area of traumatology.

Keywords: Text mining; health data mining; natural language processing.

1. Introducción

Análisis mundiales muestran que cerca del 80% de la información en las organizaciones se encuentra almacenada como datos no estructurados, los cuales requieren procesos de organización interna para la generación automática de conocimiento (Kharrazi et al., 2018). Entre los datos no estructurados se encuentra el texto libre, el cual requiere procesos avanzados de analítica y de procesamiento de lenguaje natural para descubrir conocimiento relevante.

La minería de texto puede entenderse como un área interdisciplinaria utilizada para encontrar conocimiento útil a partir de documentos de texto no estructurado, mediante la aplicación de métodos que incluyen técnicas de procesamiento de lenguaje natural, recuperación de información, aprendizaje automático, estadística, lingüística computacional, además de aquellos métodos aplicados a la minería de datos convencional. La minería de texto es similar a la minería de datos en el sentido de que se hacen exploraciones en grandes cantidades de información para extraer nuevo conocimiento. La diferencia radica en que, mientras la minería de datos se enfoca en datos estructurados, la minería de texto se enfoca en datos no estructurados, por lo que se hace necesario el uso de técnicas adicionales que permitan un adecuado procesamiento de lenguaje natural (Sukanya & Biruntha, 2012) (Otsuka & Matsushita, 2012).

La minería de texto es aplicable a todos los procesos en los que se involucre el lenguaje escrito, lo cual abre las posibilidades a prácticamente todos los procesos humanos, como medicina, mercadeo, educación, idiomática, internet, etc. El caso de las instituciones clínicas en particular es de alta generación de datos no estructurados, debido a la naturaleza de los procesos que allí se tienen, puesto que el personal médico registra procedimientos mediante redacción de texto libre, además de las imágenes generadas por las ayudas diagnósticas. El conocimiento oculto en estos datos tiene un alto potencial para las instituciones clínicas respecto al mejoramiento de la calidad de sus servicios, que hoy en día se ven medidos en indicadores de gestión como índice de mortandad, días promedio de estancia de los pacientes y egresos por cama, entre otros (Hospital San Vicente Fundación, 2017).

La minería de texto ha sido ampliamente utilizada en la salud para el estudio de diferentes enfermedades. En (Kushima & Nikama, 2012) se realizó un trabajo de minería de texto sobre los registros médicos realizados por enfermeras y doctores para los pacientes con hepatitis crónica en el Hospital Universitario de Miyazaki en Japón, con el propósito de explotar síntomas similares. En (Pereira & Agostinho, 2013) se propone un proceso automático de clasificación para diagnósticos de epilepsia, mediante la aplicación de la técnica de k-vecinos más cercanos permitiendo mapear los códigos de las enfermedades de acuerdo al estándar preestablecido por ICD-9 (Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud). En (Vijayakrishnan & Stewart, 2014) se encuentra una investigación en la que se aplica minería de texto para la detección temprana de las señales y síntomas de fallas cardíacas, dentro de pacientes participantes en programas de salud preventiva. En (Karystianis et al., 2015) se usaron 56 mil prescripciones de medicamentos para la atención primaria escritas en texto libre, las cuales fueron analizadas con reglas de diccionario y composición léxica que permitieron la estructuración de datos. En (Lucini et al., 2017) se aplican métodos de

minería de texto para predecir futuras hospitalizaciones y altas utilizando registros médicos tempranos del departamento de emergencias. En (Judd, 2018) se analiza el texto de registros médicos electrónicos para definir las opciones de tratamiento para pacientes con dolor lumbar, usando la herramienta clínica cTAKES (*NLP Apache Clinical Text Analysis and Knowledge Extraction System*), el algoritmo de aprendizaje automático utiliza siete años de notas clínicas extraídas del médico de atención primaria para clasificar el patrón de dolor de espalda de 20 pacientes.

De la revisión de la literatura se puede inferir que la minería de datos tiene un gran potencial para el diagnóstico temprano de enfermedades, perfilamiento de pacientes, generación de historia clínica, búsqueda de información y variables de interés en investigación médica, entre otros (Oviedo & Sánchez, 2017). Entre los retos para el sector está el hecho que los datos médicos son complejos y difíciles de analizar, por lo que se debe trabajar en estandarizar la terminología médica para aplicar procesos de minería de textos que generen datos de alta calidad para obtener resultados óptimos desde los procesos de minería (Sun et al., 2018).

Con el objetivo de aportar en el descubrimiento de conocimiento en historias clínicas, en este artículo se realiza un estudio de los registros clínicos de pacientes de traumatología del Hospital Universitario San Vicente Fundación mediante minería de texto, como apoyo al análisis de las palabras utilizadas en los diagnósticos del área.

El Hospital Universitario San Vicente Fundación es una de las principales instituciones hospitalarias de Colombia, siendo una institución privada sin ánimo de lucro que presta servicios de salud con énfasis en la atención del paciente de alta complejidad. La unidad de Urgencias Adultos es uno de los principales servicios que ofrece el Hospital a la comunidad, en el cual se brindan servicios de primeros auxilios, cirugía y unidad de cuidado intensivo. Esta unidad, es considerada como el primer centro de atención de trauma del país, siendo la principal puerta de entrada de los pacientes, con un 96% del total de los 85.000 pacientes que en promedio atiende el Hospital durante cada año. El 26% de los casos atendidos en urgencias se debe a trauma, área en la cual se centra el presente proyecto.

En el Hospital, el proceso de documentación de la historia clínica se realiza de dos maneras: una estructurada, en donde se almacenan los datos básicos del paciente, los tratamientos y procedimientos realizados, además de los medicamentos que recibe; otra no estructurada, donde se redacta de manera libre los acontecimientos médicos del paciente por cada ingreso al hospital, técnicamente cada ingreso recibe el nombre de “episodios”. La evolución médica de un episodio es consignada de forma transaccional, generando una serie de documentos por medio de texto libre donde se diligencian los campos: SUBJETIVO, OBJETIVO, ANÁLISIS Y PLAN. El campo SUBJETIVO almacena lo que dice el paciente, el campo OBJETIVO almacena lo que el médico encuentra en el paciente al examinarlo, el campo ANALISIS almacena lo que el médico piensa según su criterio y el campo PLAN almacena el tratamiento que debe seguir el paciente según el criterio del médico.

En el presente trabajo se desarrolla un modelo de palabras para el área de traumatología del Hospital San Vicente Fundación, analizando los campos SUBJETIVO, OBJETIVO, ANÁLISIS y PLAN de las historias clínicas mediante técnicas de minería de texto.

2. Materiales y Métodos

En esta sección se describe la metodología CRISP-DM usada en el proceso de minería, se detallan los datos analizados y se presenta el diseño de los modelos analíticos desarrollados en este trabajo para el descubrimiento de conocimiento en historias clínicas del Hospital San Vicente Fundación.

2.1. Metodología CRISP-DM para el proceso de minería de datos

CRISP-DM (*C*ross-*I*ndustry *S*tandard *P*rocess for *D*ata *M*ining) es una metodología de circulación libre, destacada por su rigurosidad y completitud a través de 6 fases: entendimiento del negocio, entendimiento de los datos, preparación de los datos, modelado, evaluación y despliegue (Wirth & Hipp, 2000).

El entendimiento del negocio se centra en la comprensión de los objetivos y requisitos del proyecto desde una perspectiva empresarial, formalizando el objetivo de desarrollar un modelo de palabras para el área de traumatología del Hospital San Vicente Fundación, analizando las historias clínicas mediante técnicas de minería de texto. El entendimiento de los datos realiza una aproximación estadística de la información, realizando un perfilamiento de los campos analizados de las historias clínicas correspondientes al diagnóstico según el código CIE-10 y los campos de texto libre SUBJETIVO, OBJETIVO, ANÁLISIS y PLAN. La preparación de datos es una de las fases más importantes y con frecuencia es la que más tiempo requiere para realizar actividades como limpieza y transformación de los datos, que en este trabajo requiere la aplicación de técnicas de procesamiento de lenguaje natural. El modelado involucra la aplicación de técnicas de aprendizaje de máquinas que den solución al problema planteado. La evaluación utiliza métricas para determinar si los resultados del modelado cumplen los criterios de calidad esperados, requiriendo una comprensión de los objetivos del negocio. Finalmente, el despliegue consiste en utilizar los nuevos conocimientos generados para implementar acciones en la organización.

2.2. Datos analizados

Los datos analizados corresponden a 19.078 pacientes, 30.958 episodios, 174.474 documentos y 351.627 registros de traumatología en un año seleccionado por el Hospital. En la tabla 1 se especifican los atributos seleccionados para el estudio de las historias clínicas de los pacientes de traumatología.

Variable	Tipo de Variable	Descripción de la variable
<i>Episodio</i>	Númerica	Número consecutivo asociado a cada ingreso del paciente al hospital
<i>Documento</i>	Númerica	Número consecutivo asociado a cada ingreso de información en los campos de texto. La base contiene tantos registros para un documento, como diagnósticos se haya asociados al documento.
<i>Código del diagnóstico</i>	Categórica	Código alfanumérico del diagnóstico. Campo asociado a la clasificación internacional CIE-10.
<i>Descripción del diagnóstico</i>	Categórica	Descripción del diagnóstico

Variable	Tipo de Variable	Descripción de la variable
<i>Análisis</i>	Texto	Campos “Análisis”: lo que el médico piensa según su especialidad y criterio
<i>Objetivo</i>	Texto	Campos “Objetivo”: lo que el médico le encuentra al paciente al examinarlo
<i>Plan</i>	Texto	Campo “Plan”: lo que el médico considera según su especialidad debe seguir en el tratamiento del paciente
<i>Subjetivo</i>	Texto	Campo “Subjetivo”: lo que dice el paciente o el acompañante

Tabla 1 – Variables de la historia clínica seleccionadas para el estudio.

El diagnóstico es la variable objetivo considerada en este estudio, ya que se plantea la creación de un modelo de palabras para cada código CIE-10. Los diagnósticos con mayor cantidad de registros asociados se presentan en la tabla 2.

Código	Diagnóstico	Cantidad	%
So69	Traumatismo intracraneal, no especificado	18.232	4,8%
To7X	Traumatismos múltiples, no especificados	11.115	2,9%
So99	Traumatismo de la cabeza, no especificado	6.158	1,6%
S822	Fractura de la diafisis de la tibia	6.124	1,6%
So65	Hemorragia subdural traumática	5.262	1,4%
T814	Infección consecutiva a procedimiento, no clasificada en otra parte	5.046	1,3%
T212	Quemadura del tronco, de segundo grado	4.974	1,3%
T202	Quemadura de la cabeza y del cuello de segundo grado	4.846	1,3%
S211	Herida de la pared anterior del torax	4.576	1,2%
T810	Hemorragia y hematoma que complican un procedimiento, no clasificados en otra parte	4.442	1,2%

Tabla 2 – Lista de los 10 diagnósticos con mayor cantidad de registros.

Las variables de tipo texto fueron sometidas a un proceso de preparación conocido como “Bolsa de Palabras”, en el cual se crea un diccionario con todas las palabras disponibles en el conjunto de documentos y se representan por medio de una matriz de términos. Esta representación ignora el orden en el cual aparecen las palabras y sólo tienen en cuenta la frecuencia de aparición (Cummins et al., 2018).

En el enfoque de “Bolsa de Palabras” se aplica un proceso de limpieza a los campos tipo texto conformado por 7 pasos:

- Eliminación de registros con campos nulos.

- Eliminación de caracteres especiales (tabulaciones, caracteres no imprimibles), signos de puntuación, números y espacios múltiples.
- Eliminación de artículos y preposiciones (stop words).
- Sustitución de sinónimos y acrónimos, los cuales fueron extraídos con el apoyo del Manual de Normas y Procedimientos en Trauma (Martiniano, Restrepo, & Múnera, 2016), el cual es un libro de referencia tanto en la facultad de medicina de la Universidad de Antioquia como en el Hospital para la formación en el área de trauma. Se realizaron un total de 45.886 sustituciones de términos.
- Eliminación de acentos para evitar problemas ortográficos mediante la sustitución de los acentos del idioma español correspondientes a los siguientes caracteres {á, é, í, ó, ú, ñ, ü}, siendo sustituidos respectivamente por su respectiva letra sin acento {a, e, i, o, u, n, u}.
- Eliminación de palabras no relevantes para el área ya que se presentan constantemente en todos los registros como {"paciente", "anos", "dias", "ahora", "manejo", "buena", "manana", "medico", "mas", "adecuada", "dia", "ayer", "debe", "aun", "ademas", "hoy", "pte"}.
- Reducción de las palabras a sus raíces (stemming): este proceso permite reemplazar las palabras "niña" y "niño" simplemente por la raíz "niñ" y así eliminar la variabilidad de las palabras.

Finalmente, después de ser limpiado el texto es representado de forma numérica por medio de la frecuencia de aparición de las palabras en cada documento. Algunos pasos adicionales para limpiar el texto implican eliminar las palabras que tiene una frecuencia muy baja (se encuentran en muy pocos documentos) y las palabras que tienen una frecuencia muy alta (están en casi todos los documentos).

2.3. Modelos analíticos

Una vez preparado el texto y representado de forma numérica por la frecuencia de las palabras, se aplican técnicas de minería de datos para el descubrimiento de información relevante. En la minería de datos se desarrollan principalmente dos tipos de análisis: predictivos y descriptivos. El análisis predictivo permite analizar datos futuros, prediciendo tanto valores categóricos (clasificación) como numéricos (regresión) por medio de técnicas de aprendizaje supervisado de máquinas. Algunos algoritmos supervisados son árboles de decisión, máquinas de soporte vectorial, naïve bayes, redes neuronales, k vecinos más cercanos y regresiones. Por su parte, el análisis descriptivo permite descubrir conocimiento en los datos actuales por medio de agrupaciones, reglas de asociación y selección de factores aplicando técnicas de aprendizaje no supervisado de máquinas. Algunos algoritmos no supervisados son k-means, clustering jerárquico, a priori, entre otros (Oviedo, Vélez, & Oviedo, 2015).

Para encontrar el modelo de palabras usadas en traumatología, se aplicaron tanto análisis predictivos como descriptivos, creando 5 modelos analíticos. Las técnicas empleadas fueron seleccionadas según la revisión bibliográfica sobre minería de texto aplicada a datos de la salud.

- Modelo 1 - Palabras más frecuentes en el área de traumatología: tiene por objetivo encontrar un ranking de las palabras más comúnmente utilizadas en el área de traumatología mediante la frecuencia de aparición de cada término.

- Modelo 2 - Segmentación de los episodios para encontrar similitudes en los traumas según las palabras: tiene por objetivo buscar similitudes entre los diferentes tipos de traumas según las palabras empleadas por los médicos mediante el algoritmo de clúster jerárquico aglomerativo.
- Modelo 3 - Encontrar las palabras más utilizadas en cada tipo de trauma: tiene como objetivo encontrar las palabras más utilizadas por los médicos en cada tipo de trauma, mediante el algoritmo predictivo de árbol de decisión.
- Modelo 4 - Analizar la ocurrencia conjunta de las palabras en los diferentes tipos de trauma: tiene por objetivo buscar las co-ocurrencias de las palabras teniendo en cuenta los diferentes tipos de trauma, mediante el algoritmo descriptivo apriori.
- Modelo 5 - Predecir el tipo de trauma según las palabras ingresadas por el médico: tiene por objetivo realizar una predicción del tipo de trauma de un episodio, teniendo en cuenta las palabras utilizadas por el médico y las variables estructuradas de la base de datos.

3. Resultados

Los experimentos se realizaron con la herramienta R, debido a su enfoque estadístico con una base analítica flexible y confiable para la informática. A continuación, se reportan los resultados de los modelos analíticos.

3.1. Modelo 1 - Palabras más frecuentes en el área de traumatología

Después de preparar el texto, se calcula la frecuencia de aparición de cada palabra. La Figura 1 presenta la nube de palabras después del proceso de preparación de los campos tipo texto, donde el tamaño de la palabra corresponde a la frecuencia de aparición en los registros. Las palabras se aprecian incompletas debido al proceso de reducción de raíces (*stemming*) aplicado en la preparación de datos.



Figura 1 – Nube de palabras después de aplicar la preparación del texto. El tamaño de la palabra corresponde a la frecuencia de aparición.

Mediante un histograma de frecuencias se verifican las palabras más comunes en el conjunto de datos: evolución, fractura, derecho, trauma, estable, quemadura, continúa, izquierdo, clínico, momento.

3.2. Modelo 2 - Segmentación de los episodios para encontrar similitudes en los traumas

A la matriz de frecuencias de las palabras se aplicó un clúster jerárquico aglomerativo, el cual realiza una matriz de distancias que es posible visualizar mediante un dendrograma. Para este método no es necesario indicar la cantidad de grupos, sino que se traza una línea horizontal que corta la gráfica y agrupa las palabras por las jerarquías que quedan en la parte inferior. A partir de las relaciones vistas en el dendrograma se eligieron cuatro clústeres, como se aprecia en la Figura 2 correspondientes a: (1) lesiones en los miembros superiores e inferiores ocasionados por accidentes de tránsito, (2) lesiones por quemaduras, (3) seguimientos clínicos y (4) seguimientos donde se encuentran involucradas observaciones referentes a la circulación sanguínea.

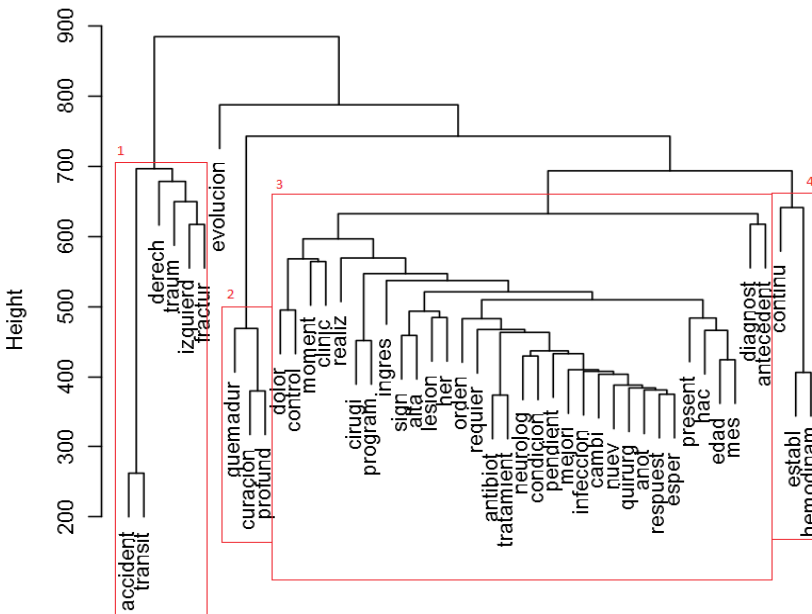


Figura 2 – Selección de clústers del dendrograma para buscar similitudes entre las palabras.

3.3. Modelo 3 - Encontrar las palabras más utilizadas en cada tipo de trauma

A la matriz de frecuencia de las palabras se aplica el algoritmo “Recursive Partitioning And Regression Trees” de árboles de decisión, para hacer una predicción del tipo de trauma según las palabras de la historia clínica. Para la evaluación de este método, se separó el conjunto de entrenamiento del conjunto de prueba con una división aleatoria

70%-30% respectivamente. En el conjunto de prueba, se calculan las medidas de matriz de confusión, precisión, cobertura y área ROC.

La variable objetivo a predecir corresponde a 23 grupos de diagnósticos de la norma CIE-10, así que se realizaron 23 clasificaciones binomiales, obteniendo un árbol de predicción para cada código CIE-10. En la Figura 3 se presentan los árboles para los dos primeros códigos CIE-10, donde un color azul fuerte en las hojas de los árboles indica que sí pertenece al tipo de trauma evaluado, mientras que un color muy claro significa que no pertenece al tipo de trauma. En cada nodo, adicionalmente se indica el porcentaje de los datos que recibe la predicción.

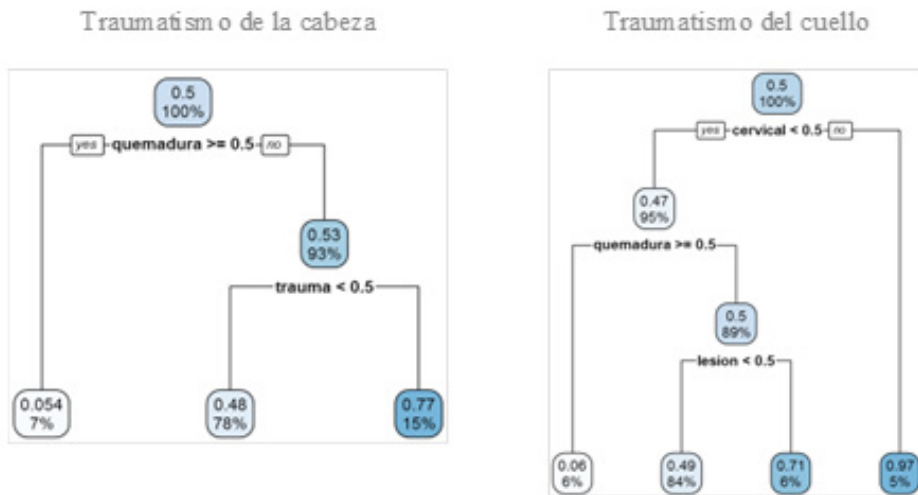


Figura 3 – Árboles de decisión para los dos primeros tipos de trauma según el código CIE-10. Los otros 21 árboles no se presentan en el documento.

Según los árboles de decisión, las palabras más relevantes de los 10 primeros traumas del código CIE-10 se presentan en la tabla 3.

Tipo de Trauma	Palabras relevantes
Traumatismo de la cabeza	• Quemadura • Trauma
Traumatismos del cuello	• Cervical • Quemadura • Lesión
Traumatismos del tórax	• Tórax • Quemadura
Traumatismos del abdomen, de la región lumbosacra de la columna lumbar y de la pelvis	• Quemadura
Traumatismos del hombro y del brazo	• Húmero • Hombro • Quemadura

Tipo de Trauma	Palabras relevantes
Traumatismos del antebrazo y del codo	• Radio • Quemadura • Fractura
Traumatismos de la muñeca y de la mano	• Mano • Quemadura • Dedo
Traumatismos de la cadera y del muslo	• Cadera • Fémur • Quemadura
Traumatismos de la rodilla y de la pierna	• Tibia • Rodilla • Quemadura • Tránsito
Traumatismos del tobillo y del pie	• Pie • Tobillo • Fractura • Tránsito • Quemadura

Tabla 3 – Palabras relevantes de los 10 primeros códigos CIE-10 de traumatología. Los otros 13 códigos analizados no son presentados en este documento.

3.4. Modelo 4 - Analizar la ocurrencia de las palabras en los diferentes tipos de trauma

Después de realizar la preparación de datos, los episodios son representados en vectores binarios que indican la presencia de las palabras en los documentos. Posterior a la representación se aplica el algoritmo apriori, el cual busca las co-ocurrencias de las palabras, es decir las palabras que aparecen de forma conjunta en los diagnósticos de trauma. El método se evalúa mediante la confianza de las reglas. En la Figura 4 se presentan algunas reglas de asociación entre las palabras. Se evidencia un bajo nivel de co-ocurrencia para las palabras analizadas, la gran mayoría de ellos con valores inferiores al 20%. La confianza más alta se presenta con las palabras “quemadura” y “profunda”, indicando que el 57% de las veces que ocurre la palabra “quemadura”, está acompañada de “profunda”.

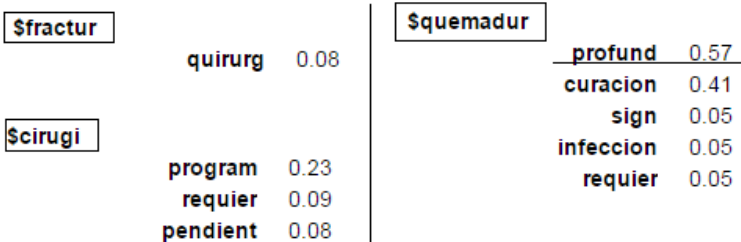


Figura 4 – Reglas de asociación encontradas entre las palabras.

3.5. Modelo 5 - Predecir el tipo de trauma según las palabras ingresadas por el médico

La predicción del tipo de trauma según las palabras empleadas por los médicos fue realizada mediante tres métodos: árbol de decisión, regresión logística y clasificador de vecinos cercanos. Para evaluar los métodos se aplicó una división aleatoria del 70% para los datos de entrenamiento y 30% para los datos de prueba.

El desempeño de los métodos puede apreciarse gráficamente en la Figura 5 que presenta el área ROC para la predicción de los 23 tipos de trauma.

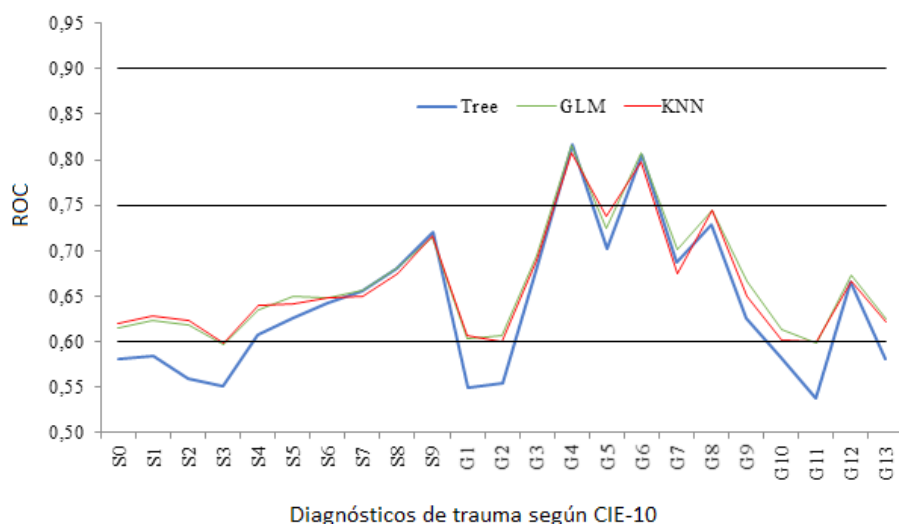


Figura 5 – Gráfica de líneas con el área bajo la curva ROC de los métodos predictivos: árbol de decisión, regresión y k-vecinos más cercanos.

En la Figura 5, los diagnósticos de trauma según CIE-10 son:

S0: Traumatismo de la cabeza	G4: Quemaduras y corrosiones de la superficie externa del cuerpo, especificadas por sitio
S1: Traumatismos del cuello	G5: Quemaduras y corrosiones limitadas al ojo y órganos internos
S2: Traumatismos del tórax	G6: Quemaduras y corrosiones de múltiples regiones del cuerpo y las no especificadas
S3: Traumatismos del abdomen, de la región lumbosacra de la columna lumbar y de la pelvis	G7: Congelamiento
S4: Traumatismos del hombro y del brazo	G8: Envenenamiento por drogas, medicamentos y sustancias biológicas
S5: Traumatismos del antebrazo y del codo	G9: Efectos tóxicos de sustancias de procedencia principalmente no medicinal
S6: Traumatismos de la muñeca y de la mano	G10: Otros efectos y los no especificados de causas externas
S7: Traumatismos de la cadera y del muslo	G11: Algunas complicaciones precoces de traumatismos
S8: Traumatismos de la rodilla y de la pierna	G12: Complicaciones de la atención médica y quirúrgica, no clasificadas en otra parte
S9: Traumatismos del tobillo y del pie	G13: Secuela de traumatismos, de envenenamientos y de otras consecuencias de causa externa
G1: Traumatismos que afectan múltiples regiones del cuerpo	
G2: Traumatismos de parte no especificada del tronco, miembro o región del cuerpo	
G3: Efectos de cuerpos extraños que penetran por orificios naturales	

Según los resultados de la Figura 5, el método de predicción seleccionado es la regresión logística, ya que presenta mayor valor en el área ROC. El procedimiento para realizar la predicción se presenta a continuación:

- Se ingresa el texto del episodio y se realiza el proceso de depuración de texto.
- Posteriormente se construye la matriz de términos del texto ingresado.
- Se aplica el método de predicción.

Para el caso de prueba, se ingresó el siguiente texto: “masculino raza negra edad actualmente hospitalizado intoxicacion causticos consecuencia misma quemaduras tracto gastrointestinal hallazgo incidental tac contr”. Obteniendo los resultados para cada uno de los grupos de trauma, tal como se muestra en la tabla 4. El tipo de trauma al que pertenece el texto según la predicción, es el que tiene mayor puntaje en la tabla de probabilidad. También puede ocurrir, que un texto esté relacionado con varios traumas y esto puede ser útil a los médicos en el momento del diagnóstico. Para el ejemplo, en la tabla 4 se presentan ordenados de mayor a menor probabilidad, los resultados del algoritmo predictor donde los grupos con mayor probabilidad están relacionados con intoxicaciones por ingestión de sustancias, seguido por grupos relacionados con quemaduras.

Grupo	Valor	Descripción del grupo
prob_G8	0,98	Envenenamiento por drogas, medicamentos y sustancias biológicas
prob_G9	0,98	Efectos tóxicos de sustancias de procedencia principalmente no medicinal
prob_G4	0,93	Quemaduras y corrosiones de la superficie externa del cuerpo, especificadas por sitio
prob_G6	0,93	Quemaduras y corrosiones de múltiples regiones del cuerpo y las no especificadas
prob_G5	0,85	Quemaduras y corrosiones limitadas al ojo y órganos internos

Tabla 4 – Resultado del algoritmo de predicción.

4. Discusión

A continuación, se presenta la discusión de resultados de los modelos analíticos.

4.1. Palabras más frecuentes en el área de traumatología

Según su frecuencia de aparición en los episodios, las 10 palabras más utilizadas en los diagnósticos son: Evolución, Fractura, Derecho, Trauma, Estable, Quemadura, Continúa, Izquierdo, Clínico, Momento.

De estas palabras puede apreciarse que {evolución, estable, continúa, clínico, momento} son palabras asociadas al seguimiento del paciente, mientras que las palabras {fractura, derecho, trauma, quemadura, izquierdo} son palabras asociadas al diagnóstico.

4.2. Segmentación de los episodios para encontrar similitudes en los traumas según las palabras

Por medio de un clustering jerárquico aglomerativo se encontraron 4 grupos, los cuales se describen a continuación.

- Cluster 1: Lesiones en los miembros superiores e inferiores ocasionados por accidentes de tránsito
- Cluster 2: Lesiones por quemaduras
- Cluster 3: Seguimientos clínicos
- Cluster 4: Seguimientos con observaciones sobre circulación sanguínea

4.3. Encontrar las palabras más utilizadas en cada tipo de trauma

La creación de los 23 modelos predictivos para cada tipo de trauma según el código CIE-10 permite identificar aquellas palabras relevantes para predecir dichos tipos de trauma, a diferencia del primer modelo realizado donde se seleccionaron las palabras por la frecuencia de aparición. A continuación, se listan algunas de las palabras seleccionadas por los 23 experimentos de predicción: Quemadura, Fractura, Curación, Tránsito, Intoxicación, Cervical, Rodilla, Pie, Fístula, Muñón, Mano, Radio, Lesión, Tibia, Fémur, Tobillo, entre otras.

4.4. Analizar la ocurrencia conjunta de las palabras en los diferentes tipos de trauma

Los resultados encontrados con este modelo analítico concuerdan con los experimentos anteriores, donde las reglas de asociación para las palabras {Evolución, Continúa, Estable, Clínico, Cirugía, Dolor} parecen estar relacionadas a seguimientos de pacientes; mientras que las reglas descritas para las palabras {Fractura, Trauma, Derecha, Izquierda, Quemadura, Diagnóstico, Antecedente} son reglas que por sus palabras asociadas parecen estar relacionadas con documentos en donde se diagnóstica al paciente. Estos resultados sugieren como trabajo futuro crear modelos de palabras por separado para las expresiones textuales de los pacientes y las expresiones médicas.

4.5. Predecir el tipo de trauma según las palabras ingresadas por el médico

De los resultados encontrados, se aprecia que la predicción con mejor desempeño corresponde a los grupos de diagnóstico relacionados con quemaduras, luego de estos se tienen los grupos relacionados con traumatismos en las extremidades y en la cadera con un desempeño aceptable. Por último, con menor desempeño los predictores para los diagnósticos relacionados con traumatismos en cabeza, tórax, abdomen y partes del cuerpo no especificadas.

El bajo desempeño de los predictores en los grupos diagnósticos relacionados con traumatismos en cabeza, tórax y abdomen, puede estar relacionado con las múltiples complicaciones que se presentan en los órganos ubicados en estas partes de cuerpo, lo cual genera términos dispersos a lo largo de todos los registros y por ende los algoritmos no logran crear relaciones entre ellos.

El bajo desempeño de los predictores en los grupos diagnósticos de los grupos G1, G2, G10 y G11, puede deberse a que son grupos que abarcan complicaciones muy generales, presentándose el mismo fenómeno descrito en el párrafo anterior. Además, la generalidad de estos grupos de diagnóstico les permite ser usados para clasificar de manera poco específica los pacientes en su ingreso al área de traumatología. Los grupos mencionados corresponden a las siguientes clasificaciones de trauma: (G1) traumatismos que afectan

múltiples regiones del cuerpo; (G2) traumatismos de parte no especificada del tronco, miembro o región del cuerpo; (G10) Otros efectos y los no especificados de causas externas; y (G11) Algunas complicaciones precoces de traumatismos.

5. Conclusiones

Con los modelos analíticos desarrollados con las historias clínicas del Hospital San Vicente Fundación, fue posible identificar el modelo de palabras usado en el área de traumatología, identificando:

- Las palabras más usadas en el conjunto global de documentos.
- Cuatro grupos de traumas que usan palabras muy similares.
- Las palabras asociadas a diferentes tipos de traumas.
- Las palabras que ocurren de forma conjunta.
- La predicción de tipo de trauma según las palabras del médico.

Para mejorar el desempeño de los modelos analíticos se plantea como sugerencia ampliar el diccionario de sinónimos y acrónimos para realizar una mejor limpieza de los datos. Adicionalmente, se sugiere realizar un ejercicio de extracción de los números presentes en los campos de texto, buscando construir gráficas evolutivas de variables médicas importantes que no se registran en los campos estructurados. Por ejemplo, índices de gravedad en trauma, frecuencia cardíaca, presión arterial, saturación de oxígeno, temperatura corporal, etc.

Referencias

- Cummins, N., Amiriparian, S., Ottl, S., Gerczuk, M., Schmitt, M., & Schuller, B. (2018). Multimodal Bag-of-Words for cross domains sentiment analysis. *En 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4954–4958. Doi: 10.1109/ICASSP.2018.8462660
- Hospital San Vicente Fundación (2017). Hospital Universitario San Vicente Fundación – Indicadores. Recuperado de: http://hospitaluniversitario.sanvicentefundacion.com/documentos/HU_Indicadores_Resultado.pdf
- Judd, M. (2018). A Medical Decision Support Tool Using Text-mining Techniques with Electronic Medical Records. *In Inquiry@ Queen's Undergraduate Research Conference Proceedings*.
- Karystianis, G., Sheppard, T., Dixon, W.G., & Nenadic, G. (2015). Modelling and extraction of variability in free-text medication prescriptions from an anonymised primary care electronic medical record research database. *BMC Medical Informatics and Decision Making*, 16(1), 18.
- Kharrazi, H., Anzaldi, L.J., Hernandez, L., Davison, A., Boyd, C.M., Leff, B., & Weiner, J.P. (2018). The value of unstructured electronic health record data in geriatric syndrome case identification. *Journal of the American Geriatrics Society*, 66(8), 1499–1507. doi:10.1111/jgs.15411

- Kushima, M.A., & Nikama, T. (2012). Text Data Mining of the Electronic Medical Record of the Chronic Hepatitis Patient. *En International multiconference of engineers and computer scientists*, 1, Hong Kong.
- Lucini, F.R., Fogliatto, F.S., DaSilveira, G.J., Neyeloff, J.L., Anzanello, M.J., Kuchenbecker, R.D.S., & Schaan, B.D. (2017). Text mining approach to predict hospital admissions using early medical records from the emergency department. *International journal of medical informatics*, 100, 1–8. Doi: 10.1016/j.ijmedinf.2017.01.001
- Martiniano, J., Restrepo, J., & Múnera, A. (2016). *Manual de normas y procedimientos en trauma*. Medellín: Editorial Universidad de Antioquia.
- Otsuka, N., & Matsushita, M. (2014). Constructing knowledge using exploratory text mining. *En Joint 7th International Conference on and Advanced Intelligent Systems (ISIS)*, 15th International Symposium, Kitakyushu, 1392–1397. Doi: 10.1109/SCIS-ISIS.2014.7044806
- Oviedo, A., & Sanchez, S. (2017). Minería de datos de la salud: Sistema de votación de técnicas analíticas para identificar los factores que influyen en la realización de cirugías estéticas. *Revista Politecnica*, 13, 43–52. Doi: 10.33571/rpolitec
- Oviedo, A., Velez, G., & Oviedo, E. (2015). Minería de datos: aportes y tendencias en el servicio de salud de ciudades inteligentes. *Revista Politecnica*, 11, 111-120. Doi: 10.33571/rpolitec
- Pereira, L.R., & Agostinho, M. (2013). ICD9-based Text Mining Approach to Children Epilepsy Classification. *Procedia Technology*, 9, 1351–1360. doi: 10.1016/j.protcy.2013.12.152
- Sukanya, M., & Biruntha, S. (2012). Techniques on text mining. *En 2012 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, 267–271. Doi: 10.1109/ICACCCT.2012.6320784
- Sun, W., Cai, Z., Li, Y., Liu, F., Fang, S., & Wang, G. (2018). Data processing and text mining technologies on electronic medical records: a review. *Journal of healthcare engineering*. Doi: 10.1155/2018/4302425
- Vijayakrishnan, R.S., & Stewart, W.F. (2014). Prevalence of Heart Failure Signs and Symptoms in a Large Primary Care Population Identified Through the Use of Text and Data Mining of the Electronic Health Record. *Journal of Cardiac Failure*, 20(7), 459–464. Doi: 10.1016/j.cardfail.2014.03.008
- Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining. *En Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, 29–39.

Heurísticas para Data Augmentation en NLP: Aplicación a Revisiones de Artículos Científicos

Rubén Sánchez Acosta, Claudio Meneses Villegas, Brian Keith Norambuena

ruben.sanchezo1@ucn.cl, cmeneses@ucn.cl, brian.keith@ucn.cl

Universidad Católica del Norte, Av. Angamos 0610, 1240000, Antofagasta, Chile.

DOI: 10.17013/risti.34.44-53

Resumen: Las técnicas de *data augmentation* son esenciales para entrenar algoritmos de *machine learning*, donde el conjunto de datos inicial es más pequeño que lo requerido debido a la complejidad del modelo. En modelos de aprendizaje automático, la robustez del proceso de entrenamiento depende altamente de grandes volúmenes de datos etiquetados, los cuales son costosos de producir. Un enfoque eficaz para tratar con este problema es generar automáticamente nuevos ejemplos etiquetados usando técnicas de *data augmentation*. En el procesamiento del lenguaje natural, en particular en el idioma español, hay una falta de técnicas bien definidas que permitan incrementar un conjunto de datos. En este artículo, se proponen un conjunto de heurísticas para *data augmentation* en NLP, las cuales son aplicadas en el dominio de las revisiones de artículos científicos.

Palabras-clave: Data Augmentation; NLP; Revisiones de Artículos.

Heuristics for Data Augmentation in NLP: Application to scientific paper reviews

Abstract: Data augmentation techniques are essential for training machine learning algorithms, where the initial data set is smaller than required due to the model complexity. In machine learning models, the robustness of the training process is highly dependent on large volumes of labeled data, which are expensive to produce. An effective approach to deal with this problem is to automatically generate new tagged examples using data augmentation techniques. In the processing of natural language, particularly in the Spanish language, there is a lack of well-defined techniques that allow increasing a set of data. In this article, we propose a set of heuristics for data augmentation in NLP, which are applied to the domain of reviews of scientific articles.

Keywords: Data Augmentation; NLP; Paper Reviews.

1. Introducción

En el ámbito de la minería de datos y el aprendizaje automático, el rápido aumento en la cantidad de datos disponibles ha facilitado el desarrollo de técnicas tales como *deep learning*,

que dada la complejidad de sus modelos requieren una enorme cantidad de datos para obtener un buen rendimiento. No obstante, no todos los dominios han visto este aumento en la disponibilidad de datos, en muchos casos debido al alto costo, ya sea económico o técnico, que puede significar la obtención de muestras rotuladas con un valor del atributo clase. Considerando estos antecedentes, varios de estos dominios se verían beneficiados de técnicas que permitan obtener datos adicionales con un menor costo asociado, en este contexto es que se utilizan las técnicas de *data augmentation*. En particular, este trabajo se centra en la aplicación de técnicas de *data augmentation* en el contexto de análisis de sentimientos aplicado al dominio de revisiones de artículos científicos, las cuales en su mayor parte corresponden a texto plano rotulados con la evaluación del revisor.

El problema base que se desea mejorar consiste en predecir si un artículo debería ser aceptado o rechazado en función de los comentarios aportados por los revisores. Otro aspecto importante en este dominio es poder cuantificar la consistencia entre los comentarios de un revisor con respecto a su evaluación, usualmente expresada en una escala multiclase. En este trabajo se aborda la primera parte de la investigación, que consiste en ampliar el conjunto de datos aplicando técnicas de *data augmentation*, en una segunda etapa de la investigación se utilizará el enfoque de transferencia de conocimientos, para entrenar un modelo basado en *deep learning* utilizando el conjunto de datos generados en el presente artículo.

Para utilizar técnicas de *data augmentation* de manera efectiva se debe hallar una función que tome un registro como entrada y lo convierta en otro que sea válido para el dominio establecido, manteniendo la clase con que se etiquetó el dato inicial (van Dyk & Meng, 2001). En el dominio específico estudiado en el presente trabajo, se deben encontrar técnicas para convertir una revisión de artículo científico en otra que, a pesar de ser generada artificialmente, su texto tenga sentido como revisión de artículo y mantenga el sentimiento positivo o negativo, en concordancia con el texto original. Para evaluar la efectividad de las técnicas utilizadas se asumirá el siguiente supuesto de concordancia semántica: “Si los cambios realizados al conjunto de datos fueran semánticamente incorrectos, entonces el rendimiento del clasificador disminuirá, en cambio, es lógico suponer que si el rendimiento aumenta entonces no hubo un cambio semántico significativo”.

Las técnicas de *data augmentation* usualmente son definidas en base al dominio donde se utilizarán (Kobayashi, 2018). Algunas de las estrategias abordadas en el presente trabajo, a pesar de poder ser aplicadas a otros dominios, fueron definidas en función de las características del dominio de revisiones de artículos científicos y de la tarea de aprendizaje, es decir el análisis de sentimientos.

Las técnicas de *data augmentation* son ampliamente estudiadas y utilizadas en el dominio de la Visión por Computador (McLaughlin, Martinez, & Miller, 2015), (Fawzi, Samolowitz, Turaga, & Frossard, 2016). En este ámbito es menos complejo su uso debido a que las imágenes son expresadas como vectores, donde se representan las componentes RGB en cada píxel con un número entre 0 y 255 por cada componente. Esta representación tiene la ventaja de que los vectores cubren un espacio continuo, por lo tanto, si se cambian algunos valores en las componentes de los píxeles, la imagen sigue siendo un registro válido para el dominio que cubre. Además, una imagen que se rota, se traslada y se transforman de diferentes maneras, sigue siendo un registro válido muchas veces.

En el dominio del procesamiento de lenguaje natural (NLP, por sus siglas en inglés) (Grosz, Jones, & Webber, 1986), la situación es más compleja, pues cambiar una palabra por otra o alterar caracteres de manera aleatoria puede variar el significado del texto o directamente hacer que pierda todo significado. Por esta razón, cualquier función que se aplique a los datos debe ser elegida de manera cuidadosa para no llegar a corromperlos. En el lenguaje español se dificulta aún más la labor, pues muchas palabras tienen género, número y persona, los cuales al ser alterados pueden originar errores de concordancia en el texto.

Debido a la dificultad que conlleva adquirir datos para el dominio estudiado, ya que por lo general son confidenciales, es necesario recurrir a técnicas de *data augmentation* para mitigar el sobreajuste al realizar tareas de aprendizaje automático como análisis de sentimientos (Keith, Fuentes, & Meneses, 2019). El conjunto de datos utilizado se generó a partir del presentado en (Keith, Fuentes, & Meneses, 2017), cuenta con un total de 300 revisiones, 3 por cada artículo, donde 150 son positivas y 150 son negativas, para evitar un desbalance entre las clases.

Mediante las técnicas de *data augmentation* propuestas el conjunto de datos podrá ser incrementado, de forma tal que puedan realizarse técnicas avanzadas de análisis de sentimientos, como es el caso de *deep learning*, modelando el dominio de mejor manera y evitando el sobreajuste a los datos iniciales. Usualmente el aumento de los datos incrementa la dimensión del problema pues aparecen palabras que anteriormente no eran parte del corpus, por lo que se debe verificar que este incremento no perjudica el aprendizaje sobre el conjunto de datos aumentados.

2. Materiales y Métodos

En esta sección se enumeran primeramente las técnicas de *data augmentation* utilizadas, luego se explica de qué manera se llevó a cabo la experimentación. El proceso completo llevado a cabo se muestra en la Figura 1.

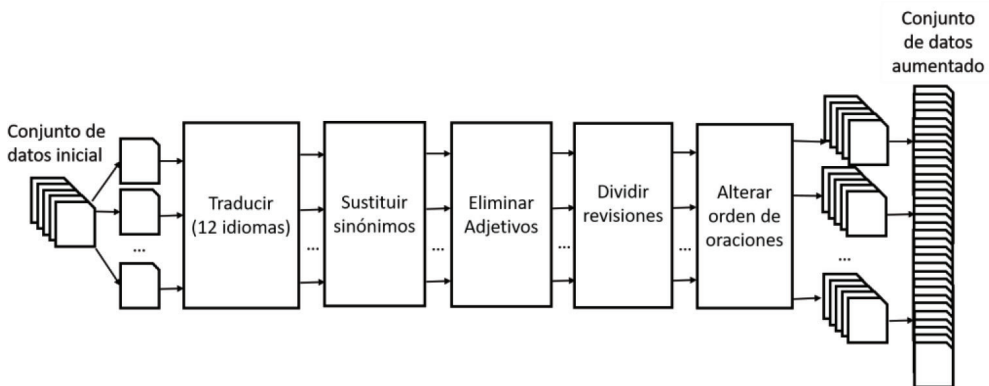


Figura 1 – Proceso de *data augmentation* realizado a los datos

2.1. Técnicas de data augmentation

Se utilizaron 5 heurísticas para aumentar los datos: 1) traducir el texto a idiomas extranjeros para luego regresarlo al español; 2) intercambiar palabras por sinónimos; 3) eliminar adjetivos; 4) alterar el orden de las oraciones; y 5) transformar revisiones extensas en varias revisiones pequeñas. Estas técnicas no son utilizadas de manera independiente, sino de forma sucesiva lo que hace que la cantidad de variaciones que se pueden aplicar al texto sea mucho mayor.

Debido a que cada etapa va aumentando la cantidad de datos, se decidió ordenar las heurísticas de más a menos complejas computacionalmente, de esta forma las técnicas más complejas trabajarán con menos datos, por lo que se logra disminuir el tiempo de generación de forma considerable. Utilizar las heurísticas en otro orden podría incrementar el costo computacional hasta el punto que dejaría de ser factible generar un conjunto de datos aumentado de manera automática. Con la presente investigación no se pretende encontrar el mejor orden para la aplicación de las técnicas, solamente se busca encontrar un orden que incremente los datos de manera consistente y que pueda mejorar el rendimiento del modelo. A continuación, se explica con más detalle cada una de estas heurísticas de manera individual.

Traducción a idiomas extranjeros

La traducción a otros idiomas es uno de los campos en que *deep learning* ha marcado la diferencia y de alguna manera se puede sacar partido a la eficacia de los algoritmos actuales. Google Translate es capaz de traducir textos a más de 120 idiomas diferentes, los cuales, al ser traducidos nuevamente al español, varían con respecto al texto inicial en muchos sentidos, desde palabras hasta estructuras gramaticales completas.

Se debe tomar en consideración que, si los lenguajes son muy similares gramaticalmente al idioma español, al ser traducido el texto de vuelta se obtiene nuevamente la frase inicial y si tienen grandes diferencias gramaticales con el castellano, el significado de la frase que se genera de vuelta puede no tener mucha relación con el texto inicial. Por esta razón, se debe encontrar un balance y elegir idiomas con los cuales se obtengan frases lo más distintas posible al texto inicial manteniendo su significado.

Se revisó manualmente la utilización de varios idiomas, fueron seleccionados aquellos que generaban las revisiones más diferentes a la original, pero que mantenían su significado. En el algoritmo propuesto se eligieron 12 idiomas con los que se obtuvieron resultados satisfactorios: afrikáans, búlgaro, inglés, francés, alemán, griego, japonés, portugués, eslovaco, sueco, turco y vietnamita.

Sustitución por sinónimos

Los sinónimos son palabras diferentes que comparten un mismo significado, por lo tanto, si alguna palabra es sustituida por su sinónimo, la frase mantendría exactamente el mismo significado y por consiguiente el sentimiento expresado. Esta técnica es utilizada en trabajos anteriores (Zhang, Zhao, & LeCun, 2015). Las partes de la oración que comúnmente tienen sinónimos asociados son los sustantivos, adjetivos y verbos, por esta razón en el presente artículo solamente se seleccionan este tipo de palabras para ser sustituidas por su sinónimo.

En español la sustitución por un sinónimo no es trivial, pues este debe coincidir en género, número y persona con la palabra inicial para mantener la concordancia de la frase. No podemos simplemente sustituir “casa” por “hogar” pues su diferencia de género crearía un error de concordancia en el texto. Para evitar este problema, una vez que se sustituye la palabra por su sinónimo el texto es traducido al inglés y luego regresado al español. Al realizar este procedimiento el texto regresa sin errores de concordancia y con el mismo significado.

En el algoritmo propuesto se les asigna una probabilidad a las partes de la oración y por cada palabra del texto, si es sustantivo, adjetivo o verbo, se sustituye, con una probabilidad igual a la asignada, por su sinónimo. Las probabilidades de sustitución establecidas por defecto son de 0.2 para cada una de las partes de la oración que intervienen (sustantivo, adjetivo y verbo). Luego de tener el texto final, se traduce al inglés y se regresa al español para eliminar errores de concordancia. Se utiliza la librería “Stanford POS Tagger” para identificar las partes de la oración y el API Google Translate de la plataforma Google Cloud para realizar las traducciones.

Eliminación de adjetivos

Los adjetivos son palabras que complementan a los sustantivos y en muchas ocasiones sucede que una frase es totalmente comprensible si se eliminan varios de estos. Al eliminar algunos de los adjetivos, por lo general, el texto mantiene el sentimiento expresado. Una vez son eliminados algunos adjetivos, la revisión se traduce al inglés y se regresa al español para corregir, en la medida de lo posible, incongruencias o faltas de concordancia.

Esta heurística está inspirada en la técnica dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) utilizada en redes neuronales para evitar el sobreajuste. El clasificador se robustece con esta eliminación, pues debe ser capaz de identificar el sentimiento de una frase, a pesar de que falten algunos adjetivos.

En el algoritmo propuesto se identifican los adjetivos utilizando la librería “Stanford POS Tagger” (Manning et al., 2014) y cada uno es eliminado con una probabilidad de 0.2. La librería “Stanford POS Tagger” es ampliamente utilizada en la literatura e identifica las partes de la oración con un rendimiento sobre el 96% de *accuracy* (Toutanova, Klein, Manning, & Singer, 2003).

Alteración del orden de las oraciones

Para cualquier revisión, alterar el orden de las oraciones mantiene el sentimiento de la opinión inicial y conserva a grandes rasgos la estructura del dominio en que se definen los datos. Esta estrategia añade robustez al modelo generado, pues el algoritmo de aprendizaje no condiciona sus decisiones al orden de las oraciones. En el presente trabajo se realiza una permutación aleatoria de las cláusulas y se crea una revisión con el nuevo orden.

Reducción de oraciones extensas

Cuando la revisión del experto es extensa, un humano puede identificar si la opinión es positiva o negativa al leer solamente un subconjunto de las oraciones del texto. Si es una crítica positiva, la mayoría de las oraciones deben ser positivas, y si es negativa pasaría lo contrario.

Por lo tanto, si se escogen subconjuntos de las oraciones, eligiéndolas de manera aleatoria, es muy probable que las nuevas revisiones se mantengan expresando el mismo sentimiento que la revisión original. Los casos en que la nueva revisión varíe de sentimiento, los cuales existirán en pequeña proporción, son un tipo de ruido beneficioso para el aprendizaje, pues lo hace más robusto debido a que el clasificador puede inferir que algunas de las oraciones del texto, a pesar de ser negativas, no afectan en gran medida el sentimiento general.

En el dominio de revisiones de artículos científicos se ha observado que comúnmente las opiniones positivas se expresan en una misma región del texto y en otra las negativas, por lo que las cláusulas que manifiestan un mismo sentimiento suelen ser cercanas, lo que hace que escoger oraciones contiguas no sea una buena idea. Esta es la razón por la cual las oraciones se eligen de manera aleatoria.

La técnica empleada en el presente trabajo, consiste en generar una cantidad de revisiones pequeñas proporcional al tamaño de la revisión original. Cada revisión pequeña se forma seleccionando de manera aleatoria 5 oraciones del texto inicial. Una revisión de 5 oraciones es lo suficientemente extensa como para expresar un sentimiento bien definido. Los párrafos básicos con los que se enseñan a desarrollar ideas en edades tempranas cuentan con 5 oraciones, esta es la longitud utilizada por varios educadores (Seo, 2007), (Bormuth, Carr, Manning, & Pearson, 1970), (van Gilst & Villalobos, 1996). Debido al planteamiento anterior, se asume que, al formar revisiones pequeñas de al menos 5 oraciones, estarán expresando una opinión concreta y bien definida. Además, al contener un número impar de oraciones se previene, en cierta medida, la situación donde el sentimiento positivo y el negativo tengan la misma relevancia.

2.2. Ejemplo del proceso

En esta sección se muestra un ejemplo para ilustrar todo el proceso que sigue una revisión desde su estado inicial, pasando por cada etapa y llegando a convertirse en un texto diferente. Como ejemplo, se utiliza una revisión artificial de 3 oraciones, debido a que es un texto corto la etapa de dividirla en párrafos de 5 oraciones no se aplica. Todo el proceso se ilustra en la Figura 2, como se puede apreciar se obtiene una revisión distinta que conservan el sentimiento de la opinión inicial.

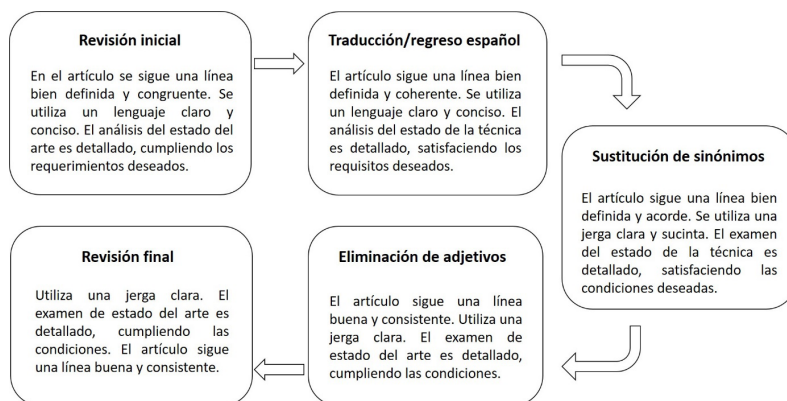


Figura 2 – Proceso de transformación que sufre una revisión al aumentar el conjunto de datos

2.3. Marco del Experimento

El método utilizado para mostrar la efectividad de las técnicas propuestas consiste en el entrenamiento de un modelo para la clasificación de sentimiento, mediante regresión logística, con el 80% de los datos del conjunto inicial y el cálculo de su *accuracy* con el 20% restante. Luego, se aplican las técnicas de *data augmentation* al 80% de entrenamiento y se entrena una vez más el modelo. El resultado esperado es mejorar el *accuracy* al clasificar el 20% que se mantuvo sin cambios. El proceso de validación utilizado se ilustra en el diagrama de la Figura 3. En esta primera etapa de la investigación se contempló generar un modelo simple a modo de prueba de concepto, por lo que se decidió entrenar un modelo de regresión logística para la tarea de análisis de sentimiento.

En esta etapa solo se desea obtener el rendimiento de la clasificación como prueba de viabilidad de los métodos de *data augmentation*, por lo que no se aplicaron técnicas más avanzadas de validación. Esto se realizará en una próxima etapa, donde se empleará la técnica de k-folds para sustentar los experimentos.

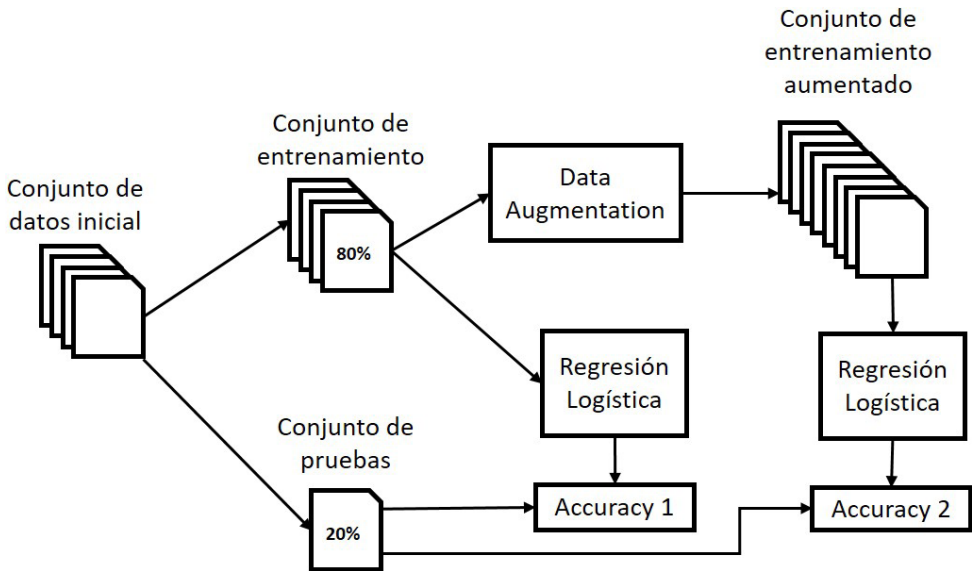


Figura 3 – Proceso de validación. Si el proceso de *data augmentation* fue beneficioso para el aprendizaje entonces el *accuracy 2* debería ser mayor que el *accuracy 1*

3. Resultados

Al aplicar las heurísticas mencionadas en la sección 2.1 y siguiendo el proceso mostrado en la Figura 3, se obtuvieron dos modelos de regresión logística, uno entrenado el 80% de los datos originales y el otro entrenado con los datos aumentados obtenidos desde el 80% de los datos originales. Ambos modelos fueron evaluados con el 20% de los datos originales. Los resultados obtenidos en términos de las métricas de rendimiento de un clasificador son resumidos en la Tabla 1.

Los resultados obtenidos con el conjunto de datos original mediante regresión logística son comparables a los reportados por Keith et al. en sus trabajos (Keith, Fuentes, & Meneses, 2017) (Keith, Fuentes, & Meneses, 2019), donde los autores obtienen como línea base 0.68 de *accuracy* y 0.64 de F1 para Naïve Bayes, 0.70 de *accuracy* y 0.69 de F1 para SVM. Por otra parte, la mejora obtenida con el conjunto de datos aumentado es comparable a la que Keith et al. obtienen al utilizar su algoritmo de Scoring y su propuesta híbrida en el caso de clasificación binaria. En particular, los autores de dicho trabajo obtuvieron 0.81 en ambas métricas para su propuesta de algoritmo de Scoring y 0.79 en ambas métricas para su propuesta híbrida. Por lo que de estos resultados se puede deducir que la aplicación de *data augmentation* provee de una mejora de rendimiento comparable, aunque marginalmente menor, a la obtenida al aplicar el método híbrido o el algoritmo de Scoring. Es posible incluso que con la combinación de *data augmentation* y estos dos métodos los rendimientos pudiesen ser mayores para este conjunto de datos en el caso de clasificación binaria.

Métricas	Conjunto de datos original		Conjunto de datos aumentado	
Cantidad de revisiones	300		14525	
Accuracy	0.67		0.74	
	Aceptado	Rechazado	Aceptado	Rechazado
Cantidad de revisiones	150	150	7235	7290
Precision	0.68	0.65	0.76	0.73
Recall	0.59	0.74	0.66	0.82
F1	0.63	0.69	0.71	0.77

Tabla 1 – Rendimiento obtenido con un modelo de regresión logística sobre el conjunto de datos original versus aumentado.

4. Discusión

Las heurísticas enunciadas lograron aumentar de manera significativa el volumen de datos, por un factor de 48 veces la cantidad de revisiones con respecto a los datos originales. Al clasificar un subconjunto de prueba del 20% de los datos originales con un modelo entrenado con los datos aumentados, se obtuvo un incremento en 7% del *accuracy* con respecto a un modelo entrenado solo con los datos originales. Esto muestra empíricamente el efecto positivo de aplicar las heurísticas de *data augmentation* en este dominio.

Con el uso de técnicas de *data augmentation* se buscaba no solo aumentar el volumen de datos y de esta forma mitigar un posible sobreajuste del modelo a los datos de entrenamiento, sino mantener la consistencia de los datos originales con respecto a los datos aumentados. Esto se logró ampliamente, dado el aumento evidenciado en *precision*, *recall* y medida F1 obtenida tanto en la clase positiva (aceptado) como en la clase negativa (rechazado).

Estos resultados permitirán en una siguiente etapa aplicar a este dominio de manera más confiable técnicas que requieren ajustar una gran cantidad de parámetros durante su aprendizaje, tal como sucede con redes neuronales profundas. De manera adicional, se planea experimentar también con técnicas de *transfer learning*, es decir, redes neuronales profundas pre-entrenadas sobre reconocimiento de patrones en texto plano.

5. Conclusiones

Respecto a las heurísticas utilizadas, algunas de ellas fueron formuladas en base a trabajos relacionados, mientras que otras son propuestas teniendo en cuenta un criterio subjetivo inspirado en diferentes dominios, considerando un sentido eminentemente práctico. A pesar de ello, los resultados de su aplicación fueron positivos, en el sentido que se cumplió el objetivo de su utilización, validando la consistencia de los datos artificialmente generados, a través de la aplicación de un clasificador simple para estimar cuantitativamente su beneficio.

El dominio de revisiones de artículos científicos, además de su relevancia científica, presenta un sinnúmero de particularidades que requieren un cuidadoso pre-procesamiento sintáctico y semántico, a fin de obtener modelos consistentes con las características del dominio (e.g., amplio desbalance de clases, amplia variabilidad del tamaño de las revisiones, limitado número de revisiones, fuerte sesgo hacia los aspectos negativos). Esto lo hace un dominio desafiante para cualquier tarea de aprendizaje automático, que requiere una cuidadosa ingeniería de datos previa a la etapa de generación de modelos.

A pesar de las características del dominio y de la subjetividad de las heurísticas, los resultados obtenidos son alentadores, dado que presentan un beneficio cuantitativo de un aumento significativo en el tamaño de los datos y a la vez un mejor rendimiento de un clasificador basado en los datos aumentados en comparación a uno basado solo en los datos originales.

Referencias

- Bormuth, JR, Carr, J, Manning, J, & Pearson, D. (1970). Children's Comprehension of Between and Within Sentence Syntactic Structures. *Journal of Educational Psychology*, 61(5), 349–357.
- Fawzi, A., Samolowitz, H., Turaga, D., & Frossard, P. (2016). Adaptive data augmentation for image classification. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Phoenix, Arizona, USA, 3688–3692. IEEE. DOI: 10.1109/ICIP.2016.7533048.
- Grosz, B.J., Jones, K.S., & Webber, B.L. (1986). *Readings in natural language processing*. Los Altos, California, USA: Morgan Kaufmann.
- Keith, B., Fuentes, E., & Meneses, C. (2019). Sentiment analysis and opinion mining applied to scientific paper reviews. *Intelligent Data Analysis*, 23(1), 191–214.

- Keith, B., Fuentes, E., & Meneses, C. (2017). A Hybrid Approach for Sentiment Analysis Applied to Paper Reviews. In *Proceedings of WISDOM 2017 at ACM SIGKDD Conference*, Halifax, Nova Scotia, Canada. Retrieved from: <https://sentic.net/wisdom2017fuentes.pdf>
- Kobayashi, S. (2018). Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2, 452–457.
- Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 55–60. DOI: 10.3115/v1/P14-5010.
- McLaughlin, N., Martinez, J., & Miller, P. (2015). Data-augmentation for reducing dataset bias in person re-identification. In *Proceedings of the 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Karlsruhe, Germany. 1–6. DOI: 10.1109/AVSS.2015.7301739.
- Seo, B.I. (2007). Defending the five-paragraph essay. *English Journal*, 97(2), 15–16.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1), 1929–1958.
- Toutanova, K., Klein, D., Manning, C.D., & Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, 1, 173–180.
- van Dyk, D., & Meng, X. (2001). The Art of Data Augmentation. *Journal of Computational and Graphical Statistics*, 10(1), 1-50.
- van Gilst, L., & Villalobos, J. (1996). *The Five-Paragraph Essay: Legacy or Liability in English Writing Classrooms outside the US*. Retrieved from: <https://eric.ed.gov/?id=ED401537>
- Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, 649–657. Retrieved from: <https://arxiv.org/abs/1509.01626>

Planificación estática para la transmisión de datos en tiempo real con WSN

Sixto Enrique Campaña Bastidas¹, Jorge Mario Londoño Peláez²

sixto.campana@unad.edu.co, jorge.londono@upb.edu.co

¹ Universidad Nacional Abierta y a Distancia - UNAD, Calle 14 No 28 – 45, C. P. 520001 - Pasto - Nariño, Colombia

² Universidad Pontificia Bolivariana, Circular 1 No 70 – 01, C.P. 050031, Medellín - Antioquia, Colombia

DOI: 10.17013/risti.34.54–67

Resumen: Este documento presenta un método (algoritmo) de planificación para la transmisión de datos en tiempo real con redes de sensores inalámbricas (WSN), basado en técnicas de planificación estática y aplicado a redes que utilicen espacios de tiempo (time-slot) para la transmisión de datos en eventos periódicos, donde se garantice un tiempo acotado de transmisión. El documento presenta en primera instancia la revisión de algunos algoritmos existentes para transmisión en tiempo real con WSN, para luego explicar el algoritmo desarrollado y los resultados encontrados que permitieron validar la solución propuesta.

Palabras-clave: Redes de sensores inalámbricas; Algoritmos de planificación; GTS; IEEE802.15.4.

Static Scheduling for Real-Time Data Transmission with WSN

Abstract: This document presents a planning method (algorithm) for the transmission of data in real time with wireless sensor networks (WSN), based on static planning, applied to networks that use time slots for the transmission of data in periodic events, guaranteeing a limited time of transmission. The document presents in the first instance the review on techniques for transmission in real time with WSN, to then explain the developed algorithm and the results found that allowed validating the proposed solution.

Keywords: Wireless Sensor Networks; Planning Algorithms; GTS; IEEE802.15.4.

1. Introducción

El problema de la transmisión de datos en redes de sensores inalámbricas (WSN), con garantías de tiempo real, es un tema que tiene referentes importantes en la comunidad científica internacional, pero que aún no dispone de una solución, que satisfaga este requerimiento completamente (Campaña & Londoño, 2016), por lo que se ha

considerado que la transmisión de datos en tiempo real con WSN es un tema abierto y con muchas posibilidades de mejora. Por lo anterior este documento presenta un nuevo procedimiento para garantizar este tipo de transmisión con WSN, particularmente a partir de la planificación de los recursos con los cuales se cuenta para la transmisión de datos en la red, tomando como referente a la norma IEEE 802.15.4.

El algoritmo de planificación que se desarrolló (PEMM, planificación estática con técnicas matemáticas MCD (Máximo común divisor) y MCM (Mínimo común múltiplo)), parte de realizar una evaluación de la factibilidad a partir de un número de nodos que participan en una WSN, el tamaño de los paquetes a transmitir, los periodos de transmisión y la cantidad de recursos disponibles en la capa MAC. El diseño de PEMM, se basa en el concepto de supertrama definido por el estándar IEEE 802.15.4 y en particular del modo de operación por intervalos de tiempo garantizados (Guaranteed Time Slots - GTS), que establece un mecanismo para la reserva de particiones de tiempo (time-slots) para la transmisión de datos.

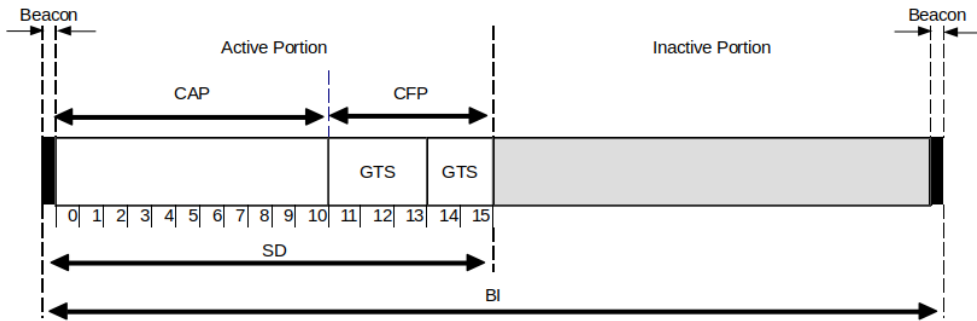


Figura 1 – Supertrama IEEE802.15.4 (Campaña, 2018)

PEMM fue comparado con otras soluciones, mencionadas más adelante, con el fin de establecer similitudes y diferencias con las mismas, además fue diseñado en pseudocódigo y programado en SCILAB (Campbell, Chancelier, & Nikoukhah, 2006), que es un programa de análisis matemático de tipo software libre, por ello los resultados presentados se basan en la simulación de casos de situaciones reales. Por otra parte, PEMM basa sus cálculos en las variables de la supertrama que maneja el estándar IEEE802.15.4, ver Figura 1. La estructura del documento parte de la comparación de algunos algoritmos previos existentes versus el algoritmo PEMM, luego se explica el funcionamiento del algoritmo a partir de un ejemplo que fue simulado y se termina con las conclusiones, resultado de la investigación.

2. Estado del Arte: algoritmos previos versus PEMM

Los algoritmos con los que se hizo el comparativo fueron los de: I-Game (Koubaa, Alves, & Tovar, 2008), EGSA (Chen, Ferreira, & Tovar, 2011), Haque (Haque, 2012), Yoo (Yoo et al., 2010), GAS (Na, 2011), ADA-MAC (Xia, et al., 2012) y FieldBus (Choi &

Kim, 2008); los cuales se seleccionaron por la similitud con el método de planificación propuesto, dado que todos trabajan con time-slot en su planificación. La comparación realizada parte definiendo las características de PEMM, ver Tabla 1.

Característica	Descripción
Modificaciones al estándar	Toma el estándar propuesto IEEE802.15.4 original sin cambios
Maneja Requerimientos de tiempo Real	Si
Obtiene los valores de SO y BO para configurar supertrama (Ver Figura 1)	Si, los calcula para aprovechar mejor los recursos de cada una de las supertramas que se generan con el manejo de estas variables. Este es uno de los elementos preponderantes del algoritmo, puesto que manejando estas variables se puede ampliar la cantidad de time-slot por supertrama, posibilitando el manejo de más dispositivos en tiempo real.
Utiliza técnicas de planificación para asignar Time-Slot	Si, utiliza la planificación estática, que refiere al uso de MCD y MCM para definir los microciclos y macrociclos del proceso (Zamorano, 1995).
Topología de funcionamiento	Estrella
Objetivo del algoritmo	Es un algoritmo que mejora el manejo de los periodos de beacon (BI) y el tamaño de la supertrama, con el fin de ajustarse a la necesidad de transmitir en tiempo real con las mismas garantías para todos los nodos que conforman la WSN. Siempre que los periodos de los sensores y las variables BO y SO lo permitan se ampliará la capacidad de time-slot por BI, ampliando la cantidad de supertramas por segundo y así permitiendo transmitir a más nodos en tiempo real.
Técnica referente utilizada para el algoritmo propuesto	Planificación estática (Zamorano, 1995).
Variables del algoritmo	N = Sensores; BI = intervalo de beacon; T = periodo de los nodos para transmitir paquetes; BO = beacon order; SO = superframe order; SD = Supertrame duration; P = Tamaño de los paquetes.
Funcionamiento	El algoritmo parte de tres variables de entrada: N, T y P, con las cuales determina el BI y SD, identificando en primera instancia los valores de BO y SO, luego analiza los microciclos y macrociclo de transmisión de la WSN, con el fin de determinar la factibilidad de transmitir en tiempo real con los nodos que conforman la red.
Tipo de solución que ofrece el algoritmo	Garantiza solución, si es factible.
Tratamiento casos no factibles	Se considera como caso no factible a una situación que luego de validar los diferentes valores de BO y SO no genera una asignación factible de time-slot a los sensores que requieren transmisión en tiempo real, ya sea por falta de time-slot para suplir el proceso o por exceso de sensores en la red, que superan la capacidad del algoritmo.

Tabla 1 – Características Generales Algoritmo de Planificación PEMM

Ahora se procede a la comparación del algoritmo PEMM (ver tabla 1), con los algoritmos que se tomaron para el estudio. A continuación, se encontrarán las tablas (2-8), donde se pueden apreciar las diferencias más importantes entre estos algoritmos.

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	I-Game si lo hace, PEEM No.
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	I-Game no tiene este aspecto como relevante. PEEM es la base de su funcionamiento
Utiliza técnicas de planificación para asignar Time-Slot	I-Game utiliza Round Robin, que es un método equitativo ordenado. PEEM utiliza planificación estática.
Objetivo del algoritmo	Los dos buscan lo mismo. La diferencia está en la forma como lo hacen. PEEM lo hace planificando el uso de los recursos y analizando el comportamiento de la WSN antes de su funcionamiento, mientras que I-Game lo hace modificando el estándar para cumplir con los requerimientos de tiempo real.
Técnicas utilizadas en la capa MAC	PEEM utiliza planificación estática (Zamorano, 1995). I-Game utiliza como referente a Network Calculus theory (Le Boudec & Thiran, 2001).
Técnica referente utilizada para el algoritmo propuesto	N = Sensores; BI = intervalo de beacon; T = periodo de los nodos para transmitir paquetes; BO = beacon order; SO = superframe order; SD = Supertrame duration; P = Tamaño de los paquetes.
Garantía de solución para transmitir en tiempo real.	El I-Game no garantiza una solución para todos los nodos de la WSN, trabaja con el mejor esfuerzo, mientras que PEEM busca ofrecer una solución con las mismas garantías para todos los nodos.
Tratamiento casos no factibles	En el I-Game cuando un nodo no puede transmitir en tiempo real está la opción de CSMA/CA, mientras que PEEM evalúa las variables BO y SO principalmente, validando valores posibles que den factibilidad.

Tabla 2 – Algoritmo PEEM Vs I-GAME (Koubaa et al., 2008)

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	EGSA amplía el número de time-slot. PEEM ninguna
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	EGSA determina el valor de SO para establecer el mayor tamaño posible, luego iguala BO a SO (lo que deja al nodo coordinador activo permanentemente).
Utiliza técnicas de planificación para asignar Time-Slot	Los dos utilizan la misma técnica en la asignación de time-slot FCFS (Primero en llegar, primero en ser atendido), pero difieren en que PEEM, basa su funcionamiento en la planificación estática para establecer el comportamiento de la WSN.
Objetivo del algoritmo	La diferencia está en que EGSA modifica el estándar para cumplir con los requerimientos de tiempo real de la WSN.
Técnica referente utilizada para el algoritmo propuesto	PEEM utiliza planificación estática (Zamorano, 1995), mientras que EGSA no utiliza ninguna técnica externa, sólo modifica el estándar para cumplir con su objetivo.
Garantía de solución para transmitir en tiempo real.	EGSA no garantiza una solución para todos los nodos de la WSN, trabaja con el mejor esfuerzo, basado en la ampliación de time-slot que propone, mientras que PEEM ofrece una solución con las mismas garantías para todos los nodos de la red.
Tratamiento casos no factibles	En EGSA cuando un nodo no puede transmitir en tiempo real está la opción de CSMA/CA, su funcionamiento se basa principalmente en el tamaño de SO, mientras que para PEEM se evalúa las variables BO y SO principalmente, validando valores posibles que den factibilidad.

Tabla 3 – Algoritmo PEEM Vs EGSA (Chen *et al.*, 2011)

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	Haque amplía el número de time-slot.
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	Con Haque estas variables se asignan de forma discreta, atendiendo sólo las recomendaciones de tomar valores entre 0 y 7 como los de mejor rendimiento.
Utiliza técnicas de planificación para asignar Time-Slot	Los dos utilizan la misma técnica en la asignación de time-slot FCFS.
Objetivo del algoritmo	Haque modifica el estándar para cumplir con los requerimientos de tiempo real de la WSN.
Técnica referente utilizada para el algoritmo propuesto	Haque no utiliza ninguna técnica externa, sólo modifica el estándar para cumplir con su objetivo.
Garantía de solución para transmitir en tiempo real.	Haque garantiza una solución de tiempo real en WSN de hasta 13 nodos, modificando el estándar con respecto a la cantidad de time-slot, mientras que PEMM ofrece una solución con las mismas garantías para todos los nodos de la red.
Tratamiento casos no factibles	Haque no trabaja de forma exclusiva con GTS, si la WSN no cumple con las condiciones para transmisión en tiempo real, se utiliza CAP y por ende CSMA/CA.

Tabla 4 – Algoritmo PEMM Vs Algoritmo de Haque (Haque, 2012)

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	En este aspecto los dos algoritmos conservan la estructura del estándar sin modificaciones.
Maneja Requerimientos de tiempo Real	Los dos algoritmos tienen el mismo enfoque, pero la diferencia radica en que PEMM establece los casos no factibles y busca facilitar una posibilidad de trabajo para los mismos, mientras que Yoo et al. no lo hace. En PEMM se busca ampliar la cantidad de nodos que pueden transmitir en tiempo real, no limitándose a 7 nodos, mientras que Yoo et al., trabaja en términos de cantidad de mensajes.
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	Los dos algoritmos buscan calcular las variables BO y SO, la diferencia radica en la forma. PEMM lo hace mediante el uso de MCM y MCD, como lo propone la técnica de planificación estática, mientras que el algoritmo de Yoo et al., lo hace partiendo del mínimo periodo de los sensores (situación que se analizó en PEMM, pero que podría generar una incompatibilidad cuando un periodo no sea múltiplo de otro para poder cumplir con la transmisión de los mensajes que envían los nodos) para obtener la variable BO, verificando que se cumpla la restricción del estándar para obtener BI, luego parte de SO=0 y va subiendo en una unidad hasta lograr soportar el ancho de banda requerido por la WSN (situación que tampoco se puede garantizar, dado que se puede llegar a agotar la cantidad de time-slot y no cumplir con el requerimiento) en la transmisión de datos en tiempo real.

Característica	Descripción
Utiliza técnicas de planificación para asignar Time-Slot	Los dos utilizan la misma técnica en la asignación de time-slot FCFS, pero difieren en que PEMM basa su funcionamiento en la planificación estática para establecer el comportamiento de la WSN y planificar mejor el uso de recursos a la hora de transmitir paquetes en tiempo real.
Objetivo del algoritmo	Buscan transmitir en tiempo real y mejorar el manejo de los recursos.
Técnica referente utilizada para el algoritmo propuesto	En PEMM se utiliza planificación estática (Zamorano, 1995), mientras que Yoo et al. utiliza técnicas que han denominado de armonización matemática.
Garantía de solución para transmitir en tiempo real.	El algoritmo de Yoo et al. y PEMM, son similares en ofrecer soluciones que garanticen tiempo real, cuando las condiciones no se cumplen se declara no factible. La diferencia está en la forma de intentar la factibilidad.
Tratamiento casos no factibles	Yoo et al., trabaja de forma que se garantice la transmisión en tiempo real de los mensajes que envían los nodos sensores, sino se cumple la red es no factible, algo similar se hace en PEMM, la diferencia está, que en este último se busca una solución factible de diversas maneras, ya sea disminuyendo el BI o evaluando el MCD de los periodos que tienen la WSN, manejando de esta manera los casos no factibles, mientras que Yoo et al., no lo hace.

Tabla 5 – Algoritmo PEMM Vs Algoritmo de Yoo et al. (Yoo et al., 2010)

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	GAS desarrolla un nuevo método de planificación basado en EDF y denominado EDCF-mGTS.
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	GAS maneja valores fijos estimados (Dentro de sus trabajos futuros esta analizar el comportamiento de estas variables), en el documento utilizan el valor de 8.
Utiliza técnicas de planificación para asignar Time-Slot	GAS, propone un nuevo modelo basado en EDF.
Objetivo del algoritmo	GAS trabaja buscando una solución de tiempo real online y trabaja con tareas esporádicas.
Técnica referente utilizada para el algoritmo propuesto	GAS basa su funcionamiento en EDF (Earliest Deadline First), diseñando un modelo de planificación ajustado a las necesidades del mismo.
Garantía de solución para transmitir en tiempo real.	GAS no garantiza transmisión en tiempo real para todos los nodos de una WSN, hace una selección de quienes cumplen con los requerimientos para poder asignarles GTS y estos serán los que transmita en tiempo real.
Tratamiento casos no factibles	Los casos no factibles de GTS se tratan con CSMA/CA y se trabaja por contención.

Tabla 6 – Algoritmo PEMM Vs Algoritmo de GAS (Na, 2011)

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	ADA-MAC cambia la estructura de la supertrama y el orden de las secciones que la componen.
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	ADA-MAC deja que los valores de BO y SO sean discretos y configurables por el usuario.
Utiliza técnicas de planificación para asignar Time-Slot	ADA-MAC propone un nuevo modelo basado en prioridades definidas en el algoritmo, descartando y no dando las mismas garantías a todos los nodos de la WSN.
Objetivo del algoritmo	ADA-MAC aumenta la cantidad de time-slot, maneja prioridades y utiliza CSMA cuando los nodos no requieran tiempo real.
Técnicas utilizadas en la capa MAC	ADA-MAC trabaja con beacon enable y también utiliza CSMA/CA cuando es necesario para paquetes que no requieren transmisión en tiempo real.
Técnica referente utilizada para el algoritmo propuesto	ADA-MAC utiliza como referente al protocolo TTP
Garantía de solución para transmitir en tiempo real.	ADA-MAC no garantiza transmisión en tiempo real para todos los nodos de una WSN, de acuerdo con la prioridad asignará GTS a quien lo necesite y rechazará los otros nodos.
Tratamiento casos no factibles	Los casos no factibles de GTS se tratan con CSMA/CA y se trabaja por contención.

Tabla 7 – Algoritmo PEMM Vs Algoritmo de ADA-MAC (Xia, et al., 2012)

Característica	Descripción
Modificaciones al estándar IEEE 802.15.4	Según la propuesta de FieldBus, se modifica el estándar para que trabaje con TDMA y EDF.
Obtiene los valores de SO y BO para configurar supertrama (Ver figura 1)	FieldBus establece periodo inactivo = 0, con lo que BO = SO. Lo que no da lugar a generar más posibilidades de cambiar el BI inicialmente planteado, dejando fija e inflexible la posibilidad de generar más supertramas y time-slot por segundo.
Utiliza técnicas de planificación para asignar Time-Slot	FieldBus trabaja con EDF y con las especificaciones del modelo Fieldbus, de donde deriva su nombre.
Objetivo del algoritmo	FieldBus maneja prioridades y utiliza CSMA cuando los nodos no requieran tiempo real y es más de tipo industrial.
Técnicas utilizadas en la capa MAC	FieldBus trabaja con beacon enabled y también utiliza CSMA/CA cuando es necesario para paquetes que no requieren transmisión en tiempo real.
Técnica referente utilizada para el algoritmo propuesto	FieldBus (Thomessse, 1998) trabaja con control distribuido, buscando la conexión punto a punto en todo momento.
Garantía de solución para transmitir en tiempo real.	FieldBus no garantiza transmisión en tiempo real para todos los nodos de una WSN, de acuerdo con la prioridad asignará GTS a quien lo necesite, utilizando EDF en la planificación y rechazará los otros nodos.
Tratamiento casos no factibles	Los casos no factibles de GTS se tratan con CSMA/CA y se trabaja por contención.

Tabla 8 – Algoritmo PEMM Vs Algoritmo FieldBus (Choi & Kim, 2008)

3. Funcionamiento Algoritmo PEMM

PEMM toma como entradas las características de una WSN con una cantidad finita de nodos y un nodo coordinador de la red, conectados en topología estrella. También asume que el coordinador de la WSN hizo el proceso de sincronización previo y cuenta con las variables que trabaja el algoritmo, que corresponden a: N, que define el número de nodos; P, que es el tamaño de los paquetes que envían los nodos y T que son los periodos en los que deben transmitir los nodos para enviar los datos al coordinador. Las variables Rts y Nts corresponden al rango de time-slot (time-slots que se asignan en la transmisión, indicando el inicio y final en la supertrama) y número máximo de timeslots, respectivamente, valores que dependerán del estándar con el que se esté trabajando. Para el caso de IEEE 802.15.4, que ha sido el ejemplo utilizado para la solución propuesta, corresponden a: Rts = [3, 4 ...16] y Nts = 13. Por otra parte, BSFD (Mínimo tamaño de un time-slot) es igual 960 símbolos y en tiempo a 0,01536 sg.

Cuando se tiene una WSN con siete o menos nodos, PEMM valida principalmente el tamaño de los time-slot y busca los valores que permitan la asignación de GTS a cada uno de los sensores que conforman la red, acorde con la planificación propuesta. Pero cuando hay más de siete sensores en la WSN, PEMM busca una planificación que garantice la transmisión en tiempo real. Para estos casos se analiza el intervalo de beacon y la relación de transmisión de datos de cada sensor, con el fin de encontrar un posible valor de BO y SO que se ajuste a los requerimientos de la aplicación. En los dos casos anteriores PEMM busca determinar si es factible o no la transmisión en tiempo real.

Por otra parte, uno de los aspectos que controla PEMM, corresponde a verificar que el número de time-slot no sobrepase el valor de 13 por supertrama y que los GTS sean máximo siete. En los casos cuando la WSN se componen de más de siete nodos, se evalúa el comportamiento de los mismos hasta llegar al macrociclo de la red, si en dicho tiempo se supera los 13 time-slot o los 7 GTS, se disminuye en 1 el valor de BO, siempre y cuando SO sea menor (ver Figura 1). Cuando se hace este proceso se disminuye el BI y se dispone de otros intervalos disponibles para asignar los periodos de los sensores de la WSN, si se encuentra una opción factible en esta asignación se dice que la WSN es factible para tiempo real, caso contrario es no factible.

4. Resultados

4.1. Aplicación paso a paso del Algoritmo PEMM

Paso 1: Lo primero que realiza PEMM es obtener los valores de BO y SO de la WSN propuesta, lo que permite conocer el BI y SD respectivamente. Para efectos de comprender el proceso se tomará un ejemplo, donde una WSN está compuesta por 10 sensores, en la Tabla 9 se pueden observar los datos iniciales.

Nodos	UmBSFD (Unidades de tiempo periódico en términos de la supertrama)	Tamaño Paquete (P) - Bits
1	3000	40
2	4000	1040

Nodos	UmBSFD (Unidades de tiempo periódico en términos de la supertrama)	Tamaño Paquete (P) - Bits
3	5000	40
4	3000	1040
5	4000	1040
6	7000	1040
7	9000	40
8	13000	1040
9	10000	40
10	21000	40

Tabla 9 – Datos Iniciales WSN Propuesta

Luego de aplicar el paso uno, se obtienen los siguientes resultados, ver Tabla 10:

Variable	Valor Obtenido
BSFD	0,01536
BO	10
SO	1
BI	15,72 sg
SD	7680 Bits
Tamaño Time-slot	480 Bits
Time-slot disponibles	13

Tabla 10 – Resultados primera simulación

Paso 2: Con los datos antes calculados y registrados en la Tabla 10, el algoritmo procede a calcular la cantidad necesaria de time-slot por nodo sensor y evalúa la factibilidad de transmitir en tiempo real con la asignación de máximo 7 GTS por supertrama, en la Tabla 11 se puede observar el cálculo realizado.

Nodos	Cantidad Time-slot requeridos por supertrama
1	1
2	3
3	1
4	3
5	3
6	3

Nodos	Cantidad Time-slot requeridos por supertrama
7	1
8	3
9	1
10	1
Total	20

Tabla 11 – Cálculo Time-slot por nodo

Paso 3: Como se puede observar, según los valores obtenidos y registrados en la Tabla 11, la WSN con 10 nodos requerirá en un determinado momento 20 time-slots, como máximo para poder transmitir en tiempo real, situación que no es factible acorde con los parámetros del estándar IEEE 802.15.4, el cual sólo permite 13 time-slot. En este caso PEMM busca hacer factible la planificación, para lo cual en primera instancia verifica si BO es mayor a SO, si es así disminuye en una unidad BO, lo que permite contar con el doble de periodos de BI, así como también se hace una reorganización de los sensores de la WSN en dos grupos, donde el primer grupo enviará los datos según le corresponda en el BI que se adapte a su periodo, mientras que el segundo grupo se desplazará en una unidad de BI con respecto al periodo inicialmente planteado. En la Tabla 12 se pueden observar los cambios que hizo PEMM en este paso.

Variable	Valor Inicialmente Obtenido	Cambio propuesto PEMM
BO	10	9
SO	1	1
BI	15,72 sg	7,68 sg
N	Un solo grupo de 10 nodos	Dos grupos de 5 nodos

Tabla 12 – Valores segunda iteración Simulación

Paso 4: Cuando se hace el cambio propuesto en el paso 3, PEMM evalúa la factibilidad de cumplir con los requerimientos del mismo para poder transmitir en tiempo real. En este caso lo que se busca es que se transmitan máximo 7 GTS y 13 time-slot en cada generación de supertrama. Para validar este requerimiento se ejecuta las iteraciones de los BI hasta que se cumpla el Macro ciclo, si en dicho proceso se encuentra que, en cada BI, las supertramas manejan hasta 7 GTS y máximo 13 time-slot, se dice que la WSN es factible para transmisión en tiempo real, caso contrario se disminuye nuevamente BO, siempre y cuando SO siga siendo menor. En el caso de la WSN de 10 sensores se hicieron las disminuciones correspondientes hasta llegar BO = SO, y no se cumplió con el requerimiento planteado, por lo anterior se realizó un nuevo paso. En la Tabla 13, se pueden observar las iteraciones que hizo PEMM.

Iteración	Valor BO	Valor SO	Resultado PEMM
Dos grupos de sensors - 1	9	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 2	8	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 3	7	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 4	6	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 5	5	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 6	4	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 7	3	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 8	2	1	La cantidad de time-slot supera el máximo permitido
Dos grupos de Sensores - 9	1	1	La cantidad de time-slot supera el máximo permitido

Tabla 13 – Validación PEMM Condiciones

Paso 5: Debido a que no se logró cumplir con el requerimiento del estándar en el paso anterior, PEMM en un nuevo paso divide el conjunto de sensores en tres grupos, donde el primer grupo trabaja con el periodo inicialmente programado, el segundo grupo trabaja con el desplazamiento en una unidad de BI y el tercer grupo con el desplazamiento de dos unidades de BI, se valida nuevamente según el macrociclo de los periodos del conjunto de sensores que constituyen la WSN y se obtuvo que la red es factible para la transmisión en tiempo real cuando BO = 9, pero trabajando con tres grupos de sensores. En la Tabla 14, se puede observar el resultado obtenido.

Itera	BI (sg)	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
0	0	1	3	1	0	0	0	0	0	0	0
1	7,86	0	0	0	3	3	3	0	0	0	0
2	15,7	0	0	0	0	0	0	1	3	1	1
3	23,5	1	0	0	0	0	0	0	0	0	0
4	31,4	0	3	0	3	0	0	0	0	0	0
5	39,3	0	0	1	0	3	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...
39997	314549,2	0	0	0	3	3	0	0	0	0	0
39998	314557,0	0	0	0	0	0	0	1	0	0	0
39999	314564,9	1	0	0	0	0	3	0	0	0	0

Tabla 14 – Resultado Final PEMM

En la Tabla 14, se puede observar que se tiene una primera columna con las iteraciones que hace PEMM en el proceso de validación de los valores de BO y SO para hacer factible la transmisión en tiempo real. En la segunda columna se tiene a BI que es el intervalo de beacon de la red y que en el caso que se ha tomado corresponde a 7,86 sg, luego están las columnas de los sensores de S1 a S10, donde se ubica la cantidad de time-slot que requieren para transmitir los datos que capturan, que a su vez se constituyen en los GTS que utiliza la WSN para transmitir en tiempo real. Como se mencionó anteriormente los sensores de la WSN de 10 nodos se organizaron en tres grupos: Grupo 1 (S1, S2, S3), Grupo 2 (S4, S5, S6) y Grupo 3 (S7, S8, S9, S10), los cuales transmiten en el BI que les corresponde, manteniendo sin cambios al Grupo 1, pero para el Grupo 2, se hizo un desplazamiento de una unidad en lo que corresponde a la iteración (itera) y para el Grupo 3 se hizo el desplazamiento de dos unidades, lo que evitó que todos los sensores transmitieran en el mismo BI con los 20 time-slot que requieren, sino por el contrario se manejará grupos que no superan los 13 time-slot, ni los 7 GTS necesarios para la transmisión en tiempo real. Al validarse esta solución se determina que la WSN debe trabajar con BO = 9 y SO = 1, en una subdivisión de tres grupos de sensores.

5. Discusión y Conclusiones

Las redes de sensores inalámbricas (WSN) brindan un soporte estructural para nuevas tecnologías, tales como el Internet de las Cosas (IoT) o los sistemas ciberfísicos (CPS), donde uno de los requerimientos importantes es la transmisión de datos en tiempo real. Los estándares para WSN se apoyan en mecanismos de acceso al medio como la multiplexación estadística (e.g. CSMA/CA), la multiplexación por división en el tiempo o por división en la frecuencia. Sin embargo, el modelo de multiplexación estadística no permite satisfacer requerimientos de tiempo real debido a la latencia no determinista introducida por el mecanismo de acceso al medio. Por tal motivo, la investigación en esta área se centra en el uso de métodos de multiplexación como mecanismos para brindar garantías de tiempo real. Sin embargo, la revisión del estado del arte arrojó que los métodos disponibles en la literatura aún son muy limitados en cuanto a proporcionar soluciones factibles utilizando los recursos limitados de la red y en cuanto a su capacidad de escalar el sistema cuando se requieren altas cantidades de sensores.

Este trabajo abordó estos problemas en el contexto específico de WSN, donde los requerimientos de los sensores se expresan en función de la periodicidad de sus transmisiones y minimizando la latencia de las transmisiones, eliminando la necesidad de mecanismos de contención para el acceso al medio a través de un esquema de planificación coordinado. Para este efecto se diseñó un algoritmo de planificación que se ha denominado PEMM, el cual toma como entradas los requerimientos de un conjunto de sensores y las restricciones de la tecnología utilizada para la implementación de la WSN, para determinar un esquema de planificación factible en caso de ser posible, o reporta la no factibilidad de la instancia del problema en cuestión. El encontrar una solución factible posibilita el despliegue de la aplicación, garantizando los requerimientos de los sensores y el cumplimiento de las restricciones de la infraestructura de red. En el caso de no ser factible el esquema, PEMM proporciona información que permite al diseñador de

la aplicación revisar sus características, de forma que se convierte en una herramienta que ayuda a ajustar el diseño del sistema para hacerlo factible.

El algoritmo de planificación fue verificado por medio de un conjunto de casos de prueba reales y sintéticos que permiten corroborar la factibilidad o no factibilidad del esquema de planificación obtenido. Este proceso se ha realizado ajustando el algoritmo a las restricciones de un caso concreto de tecnología para la implementación de redes WSN, el estándar 802.15.4. Así mismo, se implementaron y se comprobaron los beneficios del esquema de planificación en redes prototipo basadas en este estándar. Los casos sintéticos permitieron validar que el esquema permite escalar la red de sensores sin necesidad de hacer cambios al estándar, una de las limitaciones que con mayor frecuencia se identificó en el estado del arte.

Referencias

- Campaña, S. (2018). Algoritmo de planificación para redes de sensores inalámbricas con soporte para tiempo real. (Tesis de Doctorado). Universidad Pontificia Bolivariana, Medellín, Colombia.
- Campaña, S., & Londoño, J. (2016). Review: Algoritmos de planificación para la transmisión de datos en tiempo real en redes de sensores inalámbricas IEEE 802.15.4. *Revista Publicaciones e investigaciones*, (9), 177–196. DOI: <https://doi.org/10.22490/issn.2539-4088>.
- Campbell, S., Chancelier, J., & Nikoukhah, R. (2006). *Modeling and Simulation in SCILAB*. (pp. 73-105). Springer. New York.
- Chen, J., Ferreira, L., & Tovar, E. (2011). An explicit GTS allocation algorithm for IEEE 802.15. 4. In *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference*, 1-8. Portugal.
- Choi, D., & Kim, D. (2008). Wireless fieldbus for networked control systems using LR-WPAN. *International Journal of Control Automation and Systems*, 6(1), 119–125.
- Haque, S. (2012). Efficient GTS Allocation Schemes for IEEE 802.15. 4. (Master Thesis). Georgia State University.
- Koubaa, A., Alves, M., & Tovar, E. (2008). An implicit GTS allocation mechanism in IEEE 802.15.4 for time-sensitive wireless sensor networks: theory and practice. *Real-Time Systems*, 39(1-3), 169–204.
- Le Boudec, J., & Thiran, P. (2001). *Network calculus: a theory of deterministic queuing systems for the internet* (1st Ed.). New York, NY: Springer editors.
- Na, C. (2011). IEEE 802.15. 4 Wireless Sensor Networks: GTS Scheduling and Service Differentiation (Doctoral Thesis). Virginia Polytechnic Institute and State University.

- Thomesse, J. (1998). A review of the fieldbuses. *Annual reviews in Control*, (22), 35–45.
- Xia, F., Wang, L., Zhang, D., Zhang, X., & Gao, R. (2012). Ada-MAC: An adaptive MAC protocol for real-time and reliable health monitoring. In *2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)* (pp. 203-208). IEEE.
- Yoo, S., Chong, P., Kim, D., Doh, Y., Pham, M., Choi, E., & Huh, J. (2010). Guaranteeing real-time services for industrial wireless sensor networks with IEEE 802.15. 4. *IEEE Transactions on Industrial Electronics*, 57(11), 3868–76.
- Zamorano, J. (1995). Planificación Estática de Procesos en Sistemas de Tiempo Real Crítico. (Tesis Doctoral). Universidad Politécnica de Madrid, España.

Configuración de un Control de Temperatura en un Sistema Embebido de Bajo Costo, usando Herramientas de Inteligencia Artificial y el Internet de las Cosas

Carlos Mario Correa Torres, Jhon Fredy Valencia Gómez

cmcorrea@sena.edu.co, jfvalenciag@sena.edu.co

Servicio Nacional de Aprendizaje, Diagonal 104 No. 69-120, 50002, Medellín, Colombia.

DOI: 10.17013/risti.34.68–84

Resumen: El propósito de este trabajo es la implementación de un controlador difuso en línea, empleando una plataforma del internet de las cosas (IoT), para obtener un conjunto de datos que se procesan, empleando redes neuronales, con el objetivo de obtener un modelo que permita la identificación de la dinámica de un sistema térmico. Se calculan los parámetros de un controlador neuronal con base en el modelo estimado de la planta y posteriormente se transmiten a un sistema en chip (SoC), con el fin de configurar un sistema de control automático fuera de línea.

Palabras-clave: inteligencia artificial; internet de las cosas; sistemas embebidos; control automático.

Configuration of a temperature control in a low cost embedded system, using artificial intelligence and the internet of things tools

Abstract: The purpose of this paper is to implement an online fuzzy controller using an internet of things (IoT) platform to obtain a data set, that is processed using neural networks in order to build a model that permits to identify the dynamics of a thermal system. The parameters of a neural controller are calculated based on the estimated model of the plant. These parameters are transmitted to a System on a Chip (SoC) in order to configure an automatic offline control system.

Keywords: artificial intelligence; internet of things; embedded systems, automatic control.

1. Introducción

La cuarta revolución industrial ha llegado, como cada revolución, producto de la invención e introducción de herramientas tecnológicas que modifican drásticamente los métodos de producción, los empleos, las interacciones sociales y el entorno. En las primeras revoluciones, fueron la máquina de vapor, la energía eléctrica, la informática y automatización, las que impulsaron respectivamente cada una de ellas. Esta cuarta

revolución la promueve, no una tecnología específica, sino la sinergia de distintos desarrollos en áreas como biología sintética, fabricación aditiva, internet de las cosas, inteligencia artificial, *blockchain*, datos masivos, entre otras. La implementación de estas tecnologías y los productos de sus integraciones en entornos industriales, debe ser un tema de análisis prioritario para el desarrollo de la economía del país, tratar de resolver los retos que se presentan en estandarización, confiabilidad de sistemas computacionales, tráfico de datos, demanda energética, implementación de sistemas ciberseguros, costos de actualización tecnológica, entre otros, resulta imperativo si deseamos ser competitivos y tener una industria que satisfaga los requerimientos modernos. También deben considerarse las nuevas oportunidades en creación de productos y servicios soportados en estas, resulta obvio que con las revoluciones industriales desaparecen algunos tipos de trabajos y surgen otros que requieren nuevos conocimientos (Schwab, 2017), además de un alto componente de creatividad e innovación. El propósito de este trabajo es evidenciar que la aplicación de herramientas de la inteligencia artificial y el internet de las cosas en contextos industriales traerá beneficios en el desarrollo de procesos con sistemas más eficientes y con capacidad de autogestión.

2. Marco Teórico

Las técnicas de Inteligencia Artificial (IA) buscan básicamente que algoritmos informáticos emulen la observación, el entendimiento y el razonamiento humano. Por ejemplo, cuando intentamos describir muchos de los fenómenos de la naturaleza percibimos que la incertidumbre, la ambigüedad y la subjetividad hacen parte fundamental de nuestra representación y comprensión del mundo, podemos notar que algunos sistemas pueden ser definidos de manera más sencilla a partir de descripciones meramente cualitativas. Existe también, un gran conocimiento que a través de años de observación y experimentación han adquirido operadores humanos en sistemas industriales, que solo puede expresarse verbalmente, información que sería deseable ser interpretada y almacenada para su réplica. En el contexto anterior ha surgido como una solución de la representación y matematización de esta información, la lógica difusa.

2.1. Lógica Difusa

La lógica difusa tiene como base los denominados conjuntos difusos, que asumen valores de pertenencia en el intervalo $[0, 1]$. A diferencia de la lógica clásica donde la pertenencia es exclusivamente 1 o 0, pertenece o no pertenece y en donde los enunciados solo pueden ser verdaderos o falsos. Un conjunto difuso A , se caracteriza por una función de pertenencia $A: X \rightarrow [0, 1]$, donde A denota tanto al conjunto como la función de pertenencia asociada. En 1965, Lofti A. Zadeh propuso el álgebra para la operación de estos conjuntos (Zadeh, 1965), y diferentes representaciones han hecho uso de esta, como, por ejemplo, los sistemas difusos Mamdani (Mamdani & Assilian, 1975), con variables lingüísticas de entrada y salida, que poseen sistemas de inferencia basados en reglas de la forma SI antecedente, ENTONCES consecuente, cuyo fin es obtener conclusiones de las variables lingüísticas de salida, a partir de los actuales valores de las variables de entrada. En los últimos años se han desarrollado un gran número de aplicaciones principalmente en sistemas de control automático, que se construyen a partir de reglas semánticas y el conocimiento experto (Olufunke, Charles, Charles, Abraham, & Snasel, 2013) (Bazmara & Donighi, 2014).

2.2. Redes Neuronales

De la inspiración de los procesos biológicos ocurridos en el sistema nervioso central, donde una neurona, recibiendo a través de sus dendritas impulsos nerviosos, procesados luego por el cuerpo principal y transmitidos por medio de la sinapsis a otras neuronas en una estructura inmensa de conexiones e información, las redes neuronales artificiales buscan procesar información en estructuras similares, para conseguir así, un tipo de asociación y aprendizaje artificial. En la red neuronal artificial, cada neurona se alimenta de entradas que a través de una función, obtienen una salida, producto de una suma ponderada de sus entradas, pueden obtenerse también relaciones no lineales, con el uso de la llamada función de activación, a cada entrada corresponde un factor de ponderación que influye en la participación de esta en el cálculo del valor de salida, a este valor se le conoce como peso, el proceso de entrenamiento consiste en ajustar estos pesos para que la salida calculada se aproxime al valor de una salida esperada. En sus inicios las redes neuronales básicamente se usaban para clasificaciones simples, más sin embargo el desarrollo de algoritmos y estrategias de optimización (Rumelhart, Hinton, & Williams, 1986) (Rumelhart, Hinton, McClelland, 1986) han conseguido hacerlas aplicables a múltiples procesos tales como: reconocimiento de patrones, identificación de sistemas, entre otros (Bevilacqua, Cerrone-Szakal & Siegfried, 2007).

2.3. Internet de las Cosas

La miniaturización de los circuitos electrónicos y el desarrollo de sistemas embebidos, ha permitido que tecnologías de procesamiento digital y comunicación en Sistemas en Chips (SoCs) pueden integrarse prácticamente a cualquier objeto, sus grandes volúmenes de producción y la competencia entre grandes empresas que quieren participar del rentable y prometedor negocio, ha generado una reducción considerable en el costo de estos dispositivos, logrando así, que se puedan desarrollar prototipos con inversiones mínimas de dinero. Herramientas de *software* de código abierto, una comunidad enorme de desarrolladores trabajando colaborativamente también han promovido la facilidad, simplicidad en el desarrollo de aplicaciones y servicios soportados en estas tecnologías. Lo anterior ha brindado las bases para lo que conocemos como el internet de las cosas (IoT). El IoT desarrolla la idea de sensores, actuadores, máquinas, electrodomésticos etc. inmersos en redes, que permiten además de su interconexión, la presentación y el análisis de datos. Estos dispositivos encuentran aplicación en diversas áreas cuando intercambian información en servicios públicos, transporte, equipos biomédicos, inmótica, domótica, industria automovilística, entre otras (Lee & Lee, 2015) (Ning & Hu, 2012) (Gubbi, Buyya, Marusic, & Palaniswami, 2013), el verdadero potencial del IoT se evidenciará en un escenario donde máquinas y objetos a través de conexiones generen productos o servicios de forma autónoma.

3. Desarrollo de la Implementación del Sistema

La teoría del control clásico a través de ecuaciones diferenciales y el análisis en la frecuencia, ha permitido la identificación de sistemas dinámicos y el desarrollo de estrategias de control, principalmente en sistemas lineales e invariantes en el tiempo (LTI), de una entrada y una salida (SISO). La teoría de control moderna por su parte

ha permitido el análisis de sistemas más complejos; sistemas de múltiples entradas y múltiples salidas (MIMO), inclusive variantes en el tiempo. Una de las técnicas de este análisis es la representación de sistemas dinámicos por el espacio de estados, en este, un sistema dinámico puede definirse de forma matricial a través de sus entradas, las variables de estado y sus salidas (Narendra & Parthasarathy, 1990) donde:

$$\begin{aligned}\frac{dx}{dt} &\triangleq \dot{x}(t) = \mathbf{A}(t)x(t) + \mathbf{B}(t)u(t) \\ y(t) &= \mathbf{C}(t)x(t) + \mathbf{D}(t)u(t)\end{aligned}\quad (1)$$

Se conoce a $\mathbf{A}$ como la matriz de estados, $\mathbf{B}$ la matriz de entrada, $\mathbf{C}$ la matriz de salida y $\mathbf{D}$ la Matriz de transmisión directa, en general, esta última se hace cero por simplicidad, asumiendo que el sistema no tiene transmisión. Una representación por ecuaciones diferenciales sería:

$$\begin{aligned}\dot{x} &= \phi[x(t), u(t)] \quad t \in R^+ \\ y(t) &= \phi[x(t)] \\ \mathbf{x}(t) &\triangleq [x_1(t), x_2(t), \dots, x_n(t)]^T \\ \mathbf{u}(t) &\triangleq [u_1(t), u_2(t), \dots, u_p(t)]^T \\ \mathbf{y}(t) &\triangleq [y_1(t), y_2(t), \dots, y_m(t)]^T\end{aligned}\quad (2)$$

La ecuación 2 representa un sistema de orden n con p entradas y m salidas, $u_i(t)$ representa las entradas del sistema, $y_i(t)$ representa las salidas y $x_i(t)$ representa las variables de estado. El vector $x(t)$ denota el estado del sistema en el tiempo t y es determinado por el estado en un tiempo $t_0 < t$ y la entrada $u(t)$, definida en el intervalo $[t_0, t)$. La salida $y(t)$ es determinada completamente por el estado del sistema en el tiempo t . La ecuación 2 es conocida como la representación del sistema entrada-estado-salida. Para el caso de un sistema discreto lineal e invariante en el tiempo la ecuación 2 puede representarse como una ecuación en diferencias así:

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k)\end{aligned}\quad (3)$$

Donde $y(k)$, $x(k)$, $u(k)$, representan secuencias. En el caso particular de que el sistema sea un sistema SISO la ecuación 3 (Narendra & Parthasarathy, 1990) puede escribirse como:

$$y(k+1) = \sum_{i=0}^n \alpha_i y(k-i) + \sum_{j=0}^m \beta_j u(k-j) \quad (4)$$

Donde α_i, β_j son parámetros constantes desconocidos. La ecuación 4 muestra que con el cálculo de estos parámetros podríamos obtener la identificación del sistema o un modelo de la salida de la planta a partir de la información de datos de entrada-salida en instantes previos. Por la anterior, nos proponemos identificar el sistema térmico usando el método serie-paralelo (Figura 1), usando una red neuronal para obtener los valores $\hat{\alpha}_i, \hat{\beta}_j$, y así esta estimación:

$$\hat{y}_p(k+1) = \sum_{i=0}^{n-1} \hat{\alpha}_i(k) y_p(k-i) + \sum_{j=0}^{m-1} \hat{\beta}_j(k) u(k-j) \quad (5)$$

Como el propósito es implementar el sistema en tiempo real, se propone usar inicialmente un controlador difuso en línea, que en tanto se encuentra conectada la planta satisfaga las necesidades de control y a su vez pueda generar variabilidad en los datos de entrada, para que estos, en conjunto con sus salidas correspondientes, proporcionen la información necesaria para el entrenamiento de la red neuronal y con esto, la identificación del sistema. Una vez realizada la identificación del sistema térmico, se define a partir del modelo estimado de la planta, un controlador, que se implementa en el sistema embebido, para evitar la necesidad de conexión a la red. A continuación, se presenta el controlador difuso y la red neuronal para la identificación de la planta que se usaron, para luego mostrar la estructura del sistema de control.

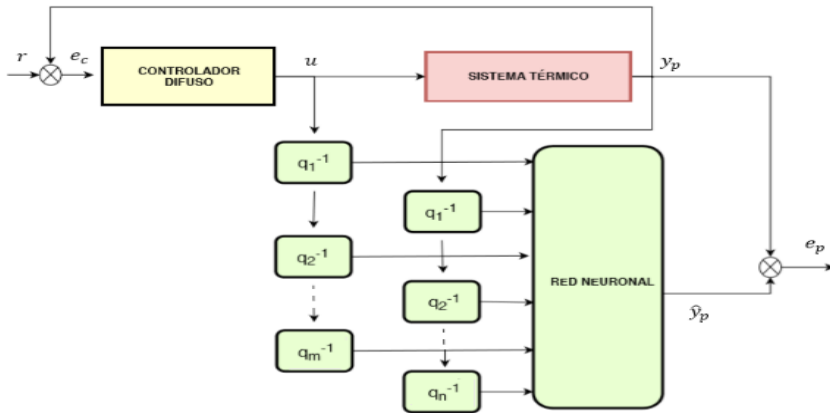


Figura 1 – Fase de identificación de la planta usando el método serie paralelo con controlador difuso en línea. q^{-1} representa las unidades de retardo de las entradas y salidas de la planta.

3.1. Controlador Difuso

Para el controlador difuso se utiliza un sistema difuso tipo Mamdani, se define el *Error* como variable lingüística de entrada, el error resulta de la diferencia entre el valor deseado o *setpoint* y el valor actual de temperatura, este se compone de cuatro conjuntos o valores difusos: *Negativo Grande*, *Negativo Mediano*, *Negativo pequeño* y *Positivo*. Se definieron funciones de pertenencia triangulares para el error, en el intervalo $[-70 \ 10]$, el rango del error se define partiendo de una temperatura ambiente

y un *setpoint* para entrenamiento de 60 grados, los rangos para cada función de pertenencia triangular se definieron subjetivamente. Como variable lingüística de salida se define la *Potencia*, con cuatro conjuntos o valores difusos; *Potencia grande*, *Potencia media*, *Potencia pequeña* y *Potencia cero*, en un intervalo [0 100], se definieron 3 funciones triangulares en intervalos subjetivos sin traslapamiento y una función *singleton* que se definió para el caso de salida de una potencia cero, se buscó que el sistema tuviese una respuesta aproximadamente proporcional o tipo rampa como puede apreciarse en la Figura 2. La simplicidad y generalidad de este sistema se plantea con el propósito de que pueda ser usado en cualquier otro sistema dinámico lineal, las variables, sus correspondientes conjuntos difusos y rangos se muestran en la Figura 2. Para la implementación de este sistema de inferencia difusa se utilizó el software *LabVIEW* y su herramienta para el desarrollo de sistemas difusos *Fuzzy System Designer* perteneciente a la versión 2018. La Figura 2 muestra la ventana de evaluación del controlador difuso propuesto. Para la operación del sistema de inferencia se definieron 4 reglas:

1. Si el Error es negativo grande entonces la potencia es potencia grande.
2. Si el Error es negativo mediano entonces la potencia es potencia media.
3. Si el Error es negativo pequeño entonces la potencia es potencia pequeña.
4. Si el Error es positivo entonces la potencia es potencia cero.

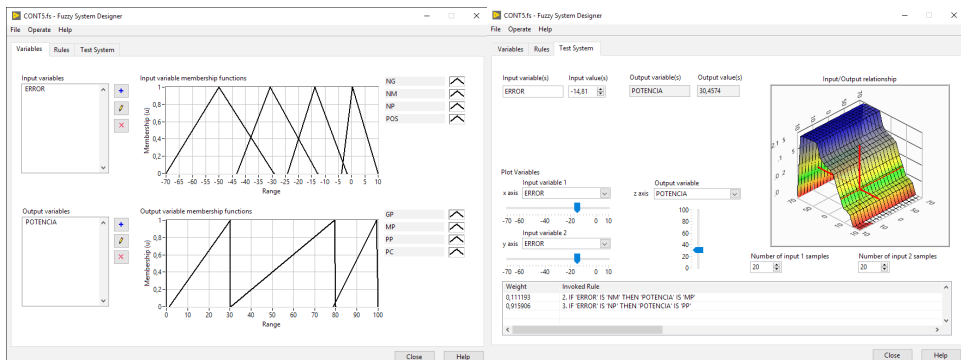


Figura 2 – Variables de entrada, salida y conjuntos difusos del controlador implementado en LABVIEW(Izq.), Respuesta del Controlador (Der.).

Cabe notar que sistemas más robustos pueden ser implementados, con mayor número de variables, conjuntos y reglas de inferencia, pueden también definirse otros apoyados en el conocimiento experto.

3.2. Red Neuronal propuesta para la identificación

Para la identificación del sistema, como se mencionó anteriormente, se propone el uso de redes neuronales, en específico el uso de un perceptrón simple (Torrubia, 2010), que pueda predecir la salida de la planta a partir de datos de entradas-salidas previos, la estructura del perceptrón propuesto se muestra en la Figura 3.

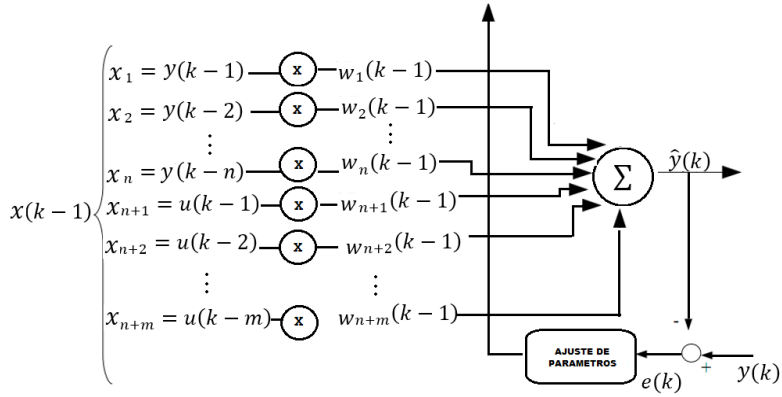


Figura 3 – Perceptrón para identificación del sistema térmico.

El vector de pesos correspondiente al perceptrón se define como:

$$\mathbf{w}(k) = [w_1, w_2, \dots, w_n, w_{n+1}, \dots, w_{n+m}]^T \quad (6)$$

El vector $\mathbf{x}(k)$ de entrada de la red se define como:

$$\mathbf{x}(k) = [x_1(k), \dots, x_n(k), x_{n+1}(k), \dots, x_{n+m}(k)]^T \quad (7)$$

cuyas componentes para la fase de entrenamiento serán las entradas y salidas correspondientes en instantes anteriores así:

$$\mathbf{x}(k-1) = [y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-m)]^T \quad (8)$$

El valor estimado de salida de la planta en el instante k se definirá entonces como:

$$\hat{y}(k) = \sum_{i=1}^{n+m} w_i(k-1) x_i(k-1) \quad (9)$$

Así un valor futuro de la planta se calculará con:

$$\hat{y}(k+1) = \sum_{i=1}^{n+m} w_i(k) x_i(k) \quad (10)$$

donde

$$\mathbf{x}(k) = [y(k), \dots, y(k-n+1), u(k), \dots, u(k-m+1)]^T \quad (11)$$

Para estimar el valor de salida de la planta en el instante k , $\hat{y}(k)$, el vector $x(k-1)$ se obtiene con el sistema *online* funcionando en lazo cerrado (Figura 1). El sistema parte midiendo el valor de la temperatura en el horno, inicialmente, temperatura ambiente, que se transmite al controlador difuso. Este calcula, basado en las reglas de inferencia, la acción de control u_i necesaria para alcanzar el *setpoint* definido en 60 grados Celsius, luego de dos segundos se toma de nuevo una muestra de temperatura y_{ip} y se repite el proceso, este termina cuando se han tomado 300 datos: 150 correspondientes a la entrada u y 150 de su respectiva respuesta y_{ip} . Se organiza el data set en una matriz donde cada elemento del vector $v_p = [y_{11} \dots y_{21} \dots y_{141}]$, se predice con los valores previos del vector columna que los precede en la Matriz *Data Set*, que se muestra en 12.

$$Data\ set: \begin{bmatrix} y_1 & y_{11} & y_{21} & \cdots & y_{141} \\ y_2 & y_{12} & y_{22} & \cdots & y_{142} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ y_{10} & y_{20} & y_{30} & \cdots & y_{150} \\ u_1 & u_{11} & u_{21} & \cdots & u_{141} \\ u_2 & u_{12} & u_{13} & \cdots & u_{142} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ u_{10} & u_{20} & u_{30} & \cdots & u_{150} \end{bmatrix} \quad (12)$$

La evaluación de la estimación del modelo del sistema térmico se realiza usando el error cuadrático medio:

$$mse = \frac{1}{l} \sum_{i=1}^l (\hat{y}_{ip} - y_{ip})^2 \quad (13)$$

Donde $l = 14$, que es el número de elementos del vector en el vector v_p extraídos del *data set*. Una vez alcanzado un valor predeterminado de este estimador el proceso de ajuste de los pesos del perceptrón termina y se considera que la estimación de la planta es aceptable y a partir de este modelo puede ya calcularse el controlador.

3.3. Controlador Neuronal

El propósito de un controlador es suministrar, a través de una variable manipulada, una entrada en la planta, que produzca que la variable controlada alcance un valor de referencia o *setpoint*. Una vez identificada la planta con el modelo que se describe en la ecuación 10, podemos calcular a partir de esta, una acción de control, despejando el valor de $u(k)$, luego dividiendo por su correspondiente coeficiente o peso w_{n+1} , y reemplazando el valor $y(k+1)$ por el valor de referencia (Torrubia, 2010):

$$u(k) = \sum_{i=1}^{n+m} w'_i(k) * z_i(k) \quad (14)$$

El vector de coeficientes del controlador w'_i puede escribirse a partir de los pesos obtenidos del modelo de identificación de la planta así:

$$w(k) = \frac{1}{w_{n+1}} [1, w_1(k), w_2(k), \dots, w_n(k), w_{n+2}(k), w_{n+m}(k)]^T \quad (15)$$

El vector z es definido a partir de los estados del modelo

$$z(k) = [y(k+1), -x(k), \dots -x_n(k), -x_{n+2}(k), \dots -x_{n+m}(k)]^T \quad (16)$$

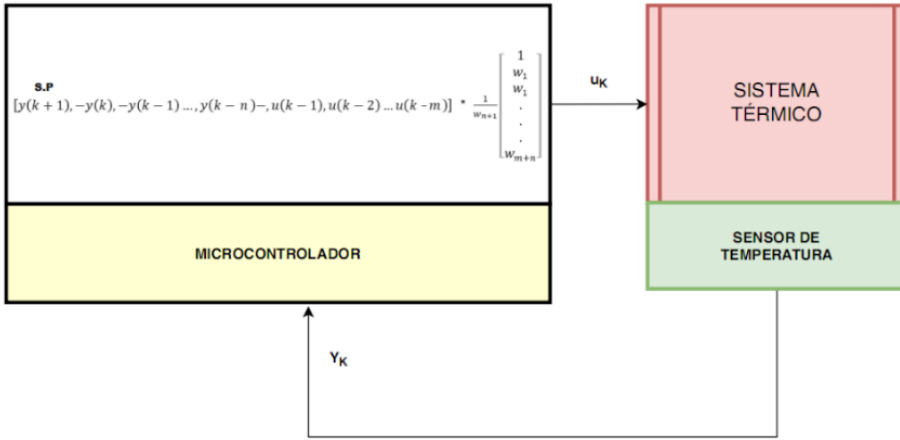


Figura 4 – Diagrama de Bloques Controlador Neuronal.

Se debe satisfacer que $w_{n+1} > 0.1$. para la convergencia del modelo. Para el entrenamiento de la red neuronal se utilizó el software Matlab versión 2018a y su herramienta para el trabajo con redes neuronales, utilizando el 80% de los datos para entrenamiento y el 20% para validación, con datos tomados en tiempo real, tanto de la salida de la planta, como del controlador, como se explicó en la sección 2.2. Durante el proceso de evaluación del controlador se entrenó la red varias veces usando sesgo o bias diferente de cero, y se observó que su valor fue siempre cercano a cero, como se evidencia en la ecuación 5, que muestra la dependencia del valor futuro principalmente de la suma ponderada de los valores previos de entrada-salida, sin termino adicional, por lo anterior en las simulaciones posteriores se consideró eliminar el sesgo.

3.4. Funcionamiento del Sistema

El funcionamiento del sistema se describe en los siguientes pasos:

1. Una vez establecida la conexión, el microcontrolador en el sistema térmico transmite el valor actual de temperatura.

2. El controlador difuso en *LabVIEW*, basado en la temperatura recibida del microcontrolador y en sus reglas de inferencia, calcula la potencia y transmite esta al microcontrolador.
3. Paralelamente a los pasos 1 y 2 los datos de temperatura y potencia son recibidos por Matlab para usarse en el entrenamiento de la red neuronal.
4. Una vez son alcanzados la cantidad de datos determinados (data set) se inicia el proceso de entrenamiento y se evalúa el desempeño usando 13, se debe satisfacer que ($mse < 1.0$) y $w_{n+1} > 0.1$, si estos criterios no se satisfacen, la acción de control sigue siendo calculada por el controlador difuso.
5. Si los criterios se cumplen, el vector w' que contiene los coeficientes calculados de la red neuronal para el controlador, es transmitido al SoC, este ignora ahora la potencia transmitida desde el controlador difuso y comienza a trabajar sin conexión.
6. El SoC calcula la potencia necesaria usando el vector w' , el vector z y la ecuación 13, el vector z es actualizado en cada instante de muestreo para el próximo cálculo de la potencia (Figura 4).
7. Los coeficientes son guardados en la memoria EEPROM del microcontrolador para que permanezcan, inclusive, el sistema se desconecta de la red eléctrica.

La transmisión y recepción de estos datos esta soportada en herramientas del IoT, específicamente, en el protocolo *Message Queuing Telemetry Transport (MQTT)* y el formato *Java Script Object Notation (JSON)*. MQTT es un protocolo *Machine To Machine*, usado ampliamente en el IoT, usa un modelo, publicar - suscribirse, donde básicamente los datos que se transmiten o publican pueden ser recibidos por cualquier

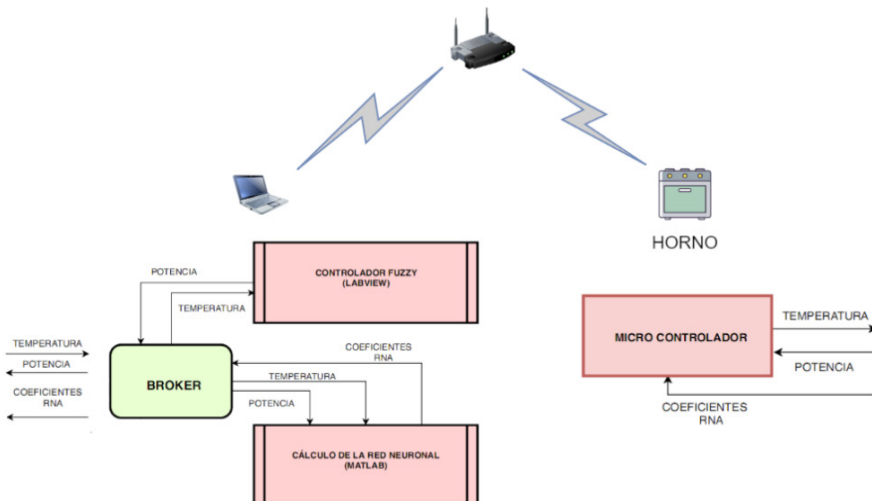


Figura 5 – Topología para prueba del sistema propuesto para configuración del controlador mientras se encuentra en línea.

dispositivo en la red que se suscriba a estos, esta característica permite compartir información paralelamente de forma eficiente. La intermediación de estos datos se consigue a través de un *Broker* que se encarga de la gestión de transporte, la topología implementada se muestra en la Figura 5.

Otra herramienta en la que se apoya este trabajo es el *Node Red*, entorno de trabajo que permite la programación basada en flujogramas, además de integrar el *Broker* para la gestión de transporte de datos, permite realizar la depuración de estos, y también su presentación usando el *dashboard*, esta herramienta se usó para visualizar las respuestas de los controladores. *Node Red* fue instalado en un computador portátil haciendo la función de servidor, con el software LabVIEW y Matlab, configurando en ellos la comunicación con el protocolo MQTT, para la implementación del sistema. El SoC que se utiliza es el Node MCU de la empresa *Espressif*, dispositivo de muy bajo costo, que integra un Procesador dual Tensilica de 32 bits, de doble núcleo a 240MHz y una interfaz WI-FI con Protocolo 802.11 b/g/n. Este microcontrolador a través del modulador de ancho de pulso (PWM) controla la potencia a través de un relevador de estado sólido que está conectado a la resistencia eléctrica del sistema térmico, la potencia calculada se encuentra limitada en el intervalo [0 100], en el sistema se integró una pantalla táctil para la visualización local de la temperatura y reconfiguración del punto de referencia. La Figura 6 muestra su implementación.

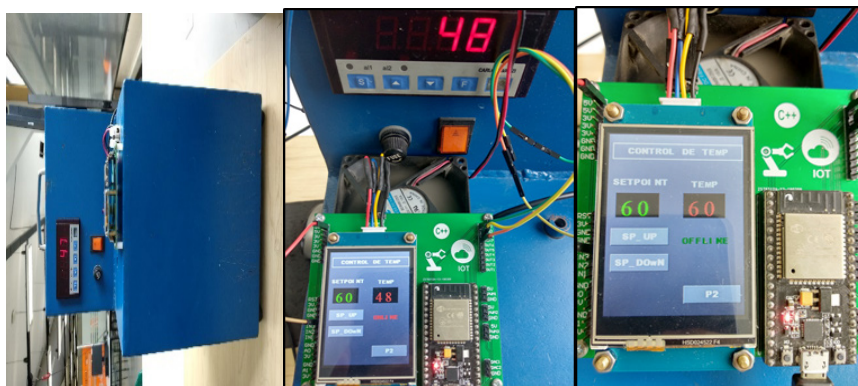


Figura 6 – (Izq.) Sistema térmico de prueba, (Centro) trabajando en línea con controlador difuso durante la identificación, (Der.) trabajando fuera de línea con el controlador neuronal.

4. Resultados

Se implementó en el sistema térmico de prueba, para tener una referencia, un controlador *Maxthermo* MC5438, usando el algoritmo Proporcional Integral Derivativo (PID), con la función de autoajuste y se estableció como referencia 58 °C, la respuesta del controlador se muestra en la Figura 7.

A continuación, en la Figura 8, se presentan los resultados obtenidos de la implementación del sistema de control difuso en línea con un *setpoint* de 60 °C, sin la aplicación de la red neuronal para la identificación.

Posteriormente se implementó el sistema completo. La Figura 9 presenta la respuesta del controlador, primero usando el controlador difuso en línea para la identificación del sistema y luego el controlador neuronal propuesto funcionando en el sistema embebido desconectado de la red. El tiempo de entrenamiento fue de 72.64 segundos, luego de haber tomado los datos para su entrenamiento.

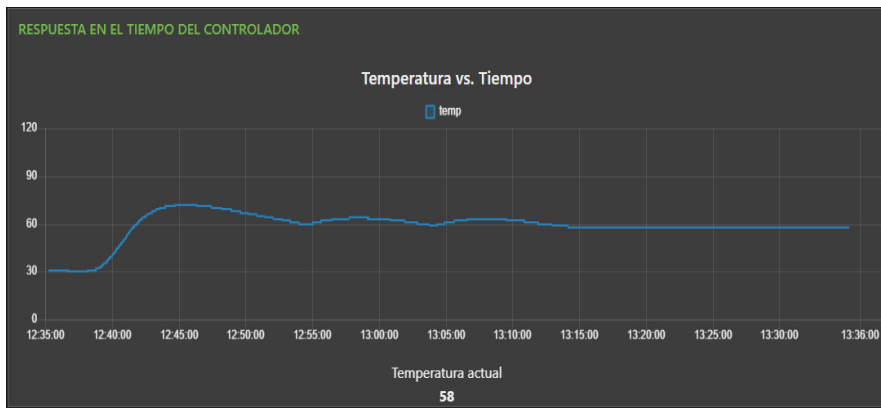


Figura 7 – Desempeño del controlador Maxthermo con algoritmo PID con autoajuste.

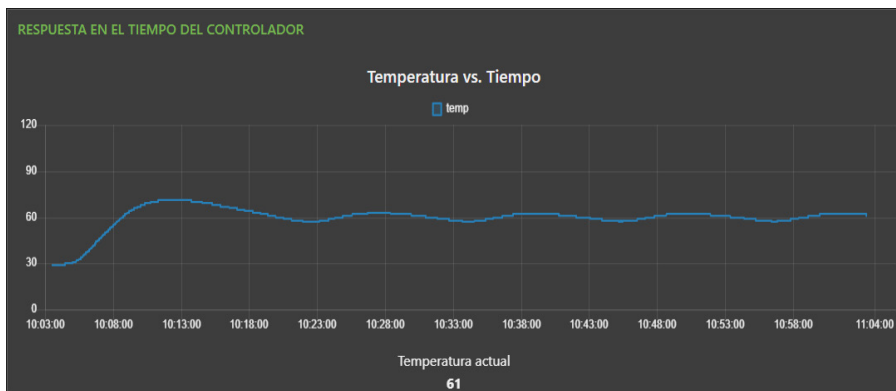


Figura 8 – Respuesta del controlador Difuso en línea implementado en la planta sin el uso del controlador neuronal.

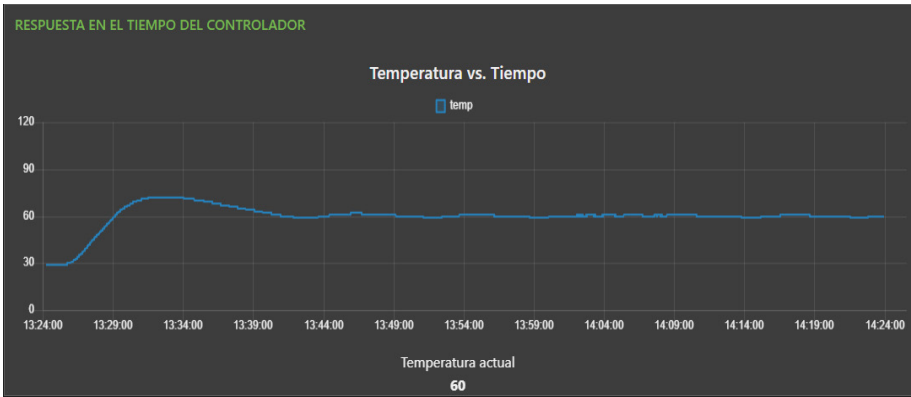


Figura 9 – Respuesta del controlador Neuronal trabajando inicialmente *online* con el controlador difuso y luego sin conexión con una referencia de 60 grados, mse= 0.6699.

Se modificó luego del establecimiento del punto de referencia de 60°C a un valor de 70°C, para verificar su funcionamiento con un valor distinto al de su entrenamiento. La Figura 10 muestra el desempeño del controlador.

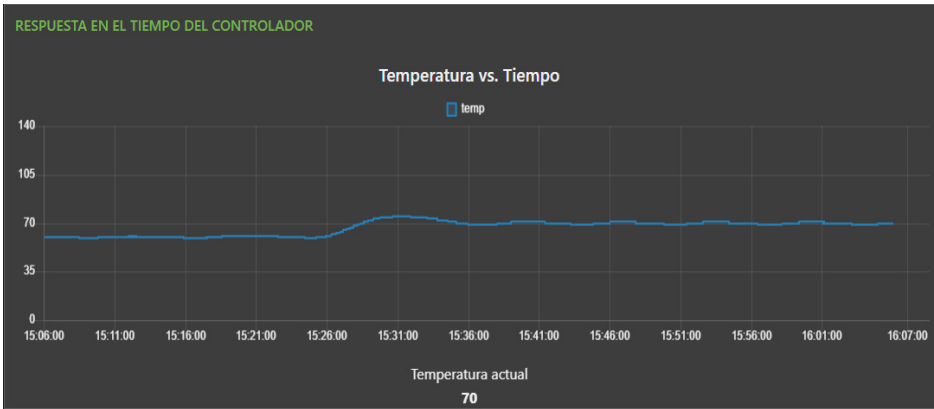


Figura 10 – Respuesta del controlador Figura 13, redefiniendo después de una referencia de 60 una setpoint de 70 grados

Las figuras 11 y 12 muestran otros resultados obtenidos de la implementación del sistema completo con sus respectivos valores de *mse* obtenidos.

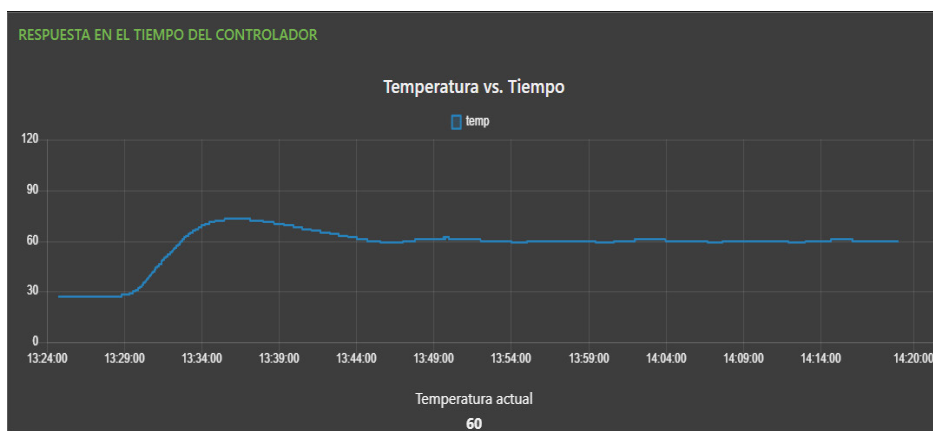


Figura 11 – Respuesta del controlador Neuronal trabajando inicialmente en línea con el controlador difuso y luego *offline* con un *Setpoint* de 60 grados, $mse= 0.8011$, 198.65 seg.

Con el objetivo de evaluar la robustez del controlador, este fue sometido a dos perturbaciones. Luego de haber llegado al punto de referencia 60° C y estabilizarse, fue introducido un flujo de aire frío constante a las 6:29 P.M, una vez el controlador consigue volver al punto de referencia 6:41 P.M, desde otra zona del horno se introduce otra corriente de aire a las 6:43 P.M como puede observarse en la Figura 13.

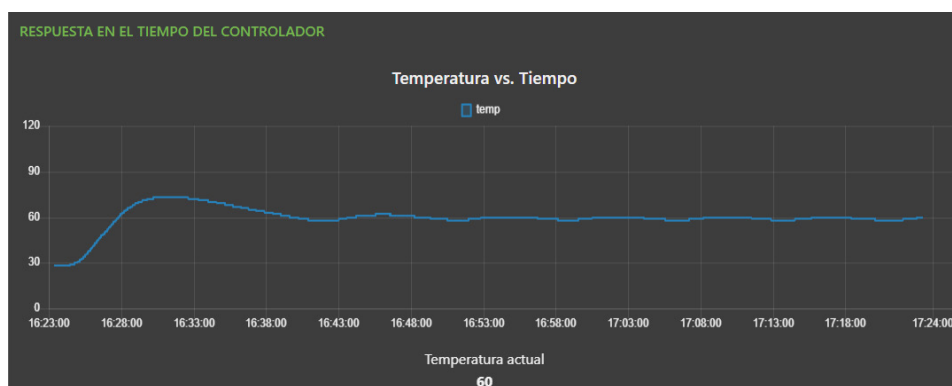


Figura 12 – Respuesta del controlador Neuronal trabajando inicialmente *online* con el controlador difuso y luego *offline* con un *Setpoint* de 60 grados, $mse= 0.5015$, 157,45 segundos.

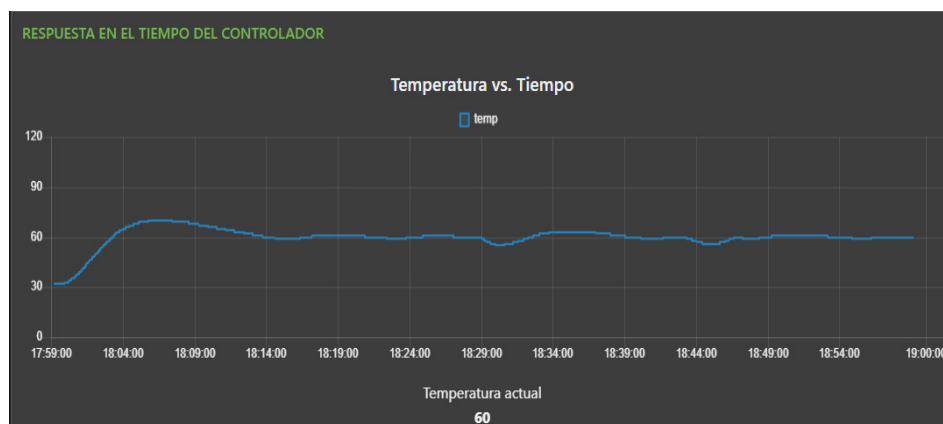


Figura 13 – Respuesta del controlador neuronal frente a dos perturbaciones de entrada

5. Conclusiones

Puede observarse que el desempeño del controlador difuso (Figura 8) es aceptable, aun siendo diseñado a partir de información verbal muy general, tiene un menor tiempo de establecimiento que el controlador *Maxthermo* (Figura 7), aunque presenta una oscilación leve y permanente alrededor del *setpoint*, también demanda una constante conexión para su operación. Cabe notar que el objetivo de este es suministrar variabilidad en los datos para alimentar el perceptrón manteniendo la planta controlada en tanto este es entrenado. Se espera que este controlador funcione solo en la etapa inicial. Por otra parte, el desempeño del controlador neuronal trabajando en desconexión con el modelo de la planta presenta un buen desempeño (Figuras 9,11,12), su tiempo de establecimiento es menor que el de controlador *Maxthermo*, esto inherente al proceso de autoajuste ejecutado por el *Maxthermo*. El controlador presenta una oscilación aún más leve que el controlador difuso, el tiempo de conexión es bajo y además no demanda de ninguna intervención en aspectos como programación o configuración, el controlador también muestra que funciona, luego de configurado para otros valores de referencia (Figura 10). La Figura 13 evidencia que se comporta bien frente a perturbaciones ya que es capaz aun siendo estas permanentes, regresar al punto de referencia.

La implementación de este sistema estaría en capacidad de configurar un controlador para cualquier sistema térmico ya que se logra a través de la información obtenida de sus entradas y salidas, este puede ser monitoreado en línea bajo demanda eliminando así el uso de interfaces hombre máquina, logrando una más efectiva gestión de la información del proceso y una menor necesidad de tiempo de conexión, todo esto soportado en un hardware de bajo costo. Implementar este sistema en la nube es

posible usando herramientas ya disponibles, bibliotecas para *machine learning*, lógica difusa y manejo de protocolos para el IoT como el MQTT que son gratuitas y de uso libre.

Referencias

- Bazmara, A., & Donighi, S. S. (2014). Bank customer credit scoring by using fuzzy expert system. *International Journal of Intelligent Systems and Applications*, 6(11), 29–35. doi: 10.5815/ijisa.2014.11.04.
- Bevilacqua, P. C., Cerrone-Szakal, A., & Siegfried, N. A. (2007). Insight into the functional versatility of RNA through model-making with applications to data fitting. *Quarterly Reviews of Biophysics*, 40(1), 55–85. doi: <https://doi.org/10.1017/S0033583507004593>
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. doi: <https://doi.org/10.1016/j.future.2013.01.010>
- Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, Investments, and Challenges for Enterprises. *Business horizon*, 58(4), 431–440. doi: <https://doi.org/10.1016/j.bushor.2015.03.008>
- Mamdani, E.H., & Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1), 1–13. doi: [https://doi.org/10.1016/S0020-7373\(75\)80002-2](https://doi.org/10.1016/S0020-7373(75)80002-2)
- Narendra, K. S., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1), 4–27. doi :10.1109/72.80202
- Ning, H., & Hu, S. (2012). Technology classification, industry, and education for Future Internet of Things. *International Journal Communication Systems*, 25(9), 1230–1241. doi: 10.1002/dac.2373
- Olufunke, O.O., Charles, U.O., Charles, A.K., Abraham, A., & Snasel, V. (2013). A fuzzy-mining approach for solving rule based expert system unwieldiness in medical domain. *Neural Network World*, 23(5), 435–450. doi: <http://www.nnw.cz/doi/2013/NNW.2013.23.027.pdf>
- Rumelhart, D.E., Hinton, G.E., & McClelland, J.L. (1986). A general framework for parallel distributed processing. In D.E. Rumelhart, J.L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition: Vol. 1. Foundations* (pp.45- 76). Cambridge, MA: MIT Press.

- Rumelhart, D.E., Hinton, G., & Williams, R. (1986). Learning Representations by Back-Propagation Errors. *Nature*, 323, 533–536. doi: <https://doi.org/10.1038/323533a0>
- Schwab, K. (2017). *The Fourth Industrial Revolution*. New York: Crown Business.
- Torrubia, S. (2010). *Redes Neuronales Multimodelo Aplicadas al Control de Sistemas*. (Tesis de maestría). Universidad Autónoma de Barcelona, España. Recuperado de https://ddd.uab.cat/pub/trerecpro/2013/hdl_2072_207064/TorrubiaCaravacaSergioR-ETISa2009-10.pdf
- Zadeh, L.A., (1965). Fuzzy sets. *Information and Control*, 8(3), 338–353. doi: [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)

Análisis de las debilidades que presentan las Entidades Muy Pequeñas al implementar el estándar ISO/IEC 29110: Una comparativa entre estado del arte y el estado de la práctica

Mirna Muñoz¹, Manuel Peralta¹, Claude Y. Laporte²

{mirna.munoz, manuel.peralta}@cimat.mx, claudelaporte@etsmtl.ca

¹ Centro de Investigación en Matemáticas A.C. – Unidad Zacatecas, Zacatecas, México, Parque Quantum, Ciudad el Conocimiento Avenida Lassec, Andador Galileo Galilei, Manzana, 3 Lote 7, 98160, Zacatecas, México.

² École de technologie supérieure, Department of Software and IT Engineering 1100, Notre-Dame Street West, Montréal, Québec, Canada, H3C 1K3.

DOI: 10.17013/risti.34.85–96

Resumen: Debido a la importancia de las Entidades Muy Pequeñas en la industria del software, hoy en día, organizaciones como la ISO, están enfocadas en el desarrollo de estándares que impulsen su competitividad, reforzando el uso de prácticas probadas de ingeniería de software y sean adecuados a las características específicas de este tipo de empresas, como lo es la serie de estándares ISO/IEC 29110. Sin embargo, las Entidades Muy Pequeñas han mostrado debilidades en la implementación de dicha serie. Este artículo realiza una comparativa entre el estado del arte y el estado de la práctica referente a la implementación del estándar ISO/IEC 29110 en entidades muy pequeñas. Los resultados muestran, que las principales debilidades están relacionadas con: la falta de ejecución de actividades de verificación y validación, y el desarrollo y/o documentación de casos y procedimientos de prueba.

Palabras-clave: Entidades Muy Pequeñas; EMPs; ISO/IEC 29110; debilidades; implementación de buenas prácticas.

Analysis of weaknesses that present the Very Small Entities in the implementation of the ISO/IEC 29110 standard: A comparison between the state of art and the state of practice

Abstract: Due to the importance of Very Small Entities in the software industry, today, organizations such as ISO, are focused on the development of standards that push their competitiveness, reinforcing the use of proven software engineering practices, and being adequate to the specific characteristics of this type of organizations, such as the ISO/IEC 29110 series of standards. However, Very Small Entities have shown weaknesses in the implementation of this series. This article makes a comparison between the state of the art and the state of practice concerning

the implementation of the ISO/IEC 29110 standard in very small entities. The results show that the main weaknesses are related to the lack of execution of verification and validation activities, and the development and/or documentation of cases and test procedures.

Keywords: Very Small Entities; VSEs; ISO/IEC 29110; weakness; best practices implementation.

1. Introducción

Hoy en día, asegurar que las Entidades Muy Pequeñas (EMPs) produzcan software con la calidad requerida es una necesidad, ya que éstas son muy importantes en la cadena de producción de software para cubrir las necesidades del mercado de software.

Se debe entender como Entidad Muy Pequeña (EMP): una empresa, organización (con o sin fines de lucro), un departamento o un proyecto que tenga hasta 25 personas (ISO/IEC, 2011).

Para este tipo de empresas, el estándar ISO/IEC 29110 proporciona un conjunto de prácticas probadas o buenas prácticas, enfocadas en reforzar los procesos de gestión del proyecto y desarrollo del software (Laporte, Muñoz & Gerançon, 2017).

Sin embargo, es importante mencionar, que un problema común que presentan las EMPs es la falta de experiencia práctica en la implementación de modelos y estándares (Software Engineering Institute, 2010).

En este contexto, el objetivo de este artículo es presentar un análisis de las principales debilidades identificadas mediante la comparación del estado del arte (obtenidas mediante la realización de una revisión sistemática), y de la práctica (obtenidas mediante la implementación del estándar ISO/IEC 29110 en 13 EMPs). Estas debilidades están relacionadas con la falta del uso de buenas prácticas en las EMPs al iniciar la implementación del estándar ISO/IEC 29110.

Este análisis, se realiza con la finalidad de establecer hallazgos en cuanto las similitudes y diferencias en las debilidades mencionadas en el estado del arte y en las identificadas en el estado de la práctica, y a partir de éstos, establecer conclusiones de las principales debilidades que presentan las EMPs al introducirse en una cultura de mejora continua.

Después de la introducción, este artículo está estructurado como sigue: en la sección 2, se muestra una visión global del estándar ISO/IEC 29110; en la sección 3, se analizan las debilidades identificadas; en la sección 4, se presenta el análisis de la comparativa entre éstas y se establecen hallazgos; y en la sección 5, se muestran las conclusiones y trabajo futuro.

2. ISO/IEC 29110

La serie de guías y directrices del estándar ISO/IEC 29110 ha sido desarrollada para brindar apoyo a las EMPs en la mejora de sus procesos de desarrollo de software (ISO/

IEC, 2011), (Laporte, Muñoz, & Gerançon, 2017). El estándar ISO/IEC 29110 proporciona un conjunto de prácticas probadas, cuya implementación permite a las EMPs obtener beneficios como: (1) un incremento de la calidad de sus productos y/o servicios; (2) una reducción en el tiempo de liberación de sus productos y; (3) una disminución de sus costos de producción (ISO/IEC, 2011), (Laporte, Muñoz, & Gerançon, 2017).

Esta serie de estándares está conformada por 4 perfiles, que pueden ser usados por las EMPs de acuerdo a sus objetivos de negocio como a continuación se lista (ISO/IEC, 2011), (Laporte, Muñoz, & Gerançon, 2017):

1. Perfil de entrada: este perfil puede ser usado por EMPs que normalmente desarrollan proyectos de 6 personas-mes o Start-ups.
2. Perfil básico: este perfil puede ser usado por EMPs que normalmente desarrollan un solo proyecto a la vez.
3. Perfil intermedio: este perfil puede ser usado por EMPs que normalmente desarrollan múltiples proyectos, dentro del contexto de una organización.
4. Perfil avanzado: este perfil puede ser usado por EMPs que desean mantener y crecer como empresas de desarrollo de software independientes y competitivas.

De las características del estándar se pueden resaltar dos: la primera es, que el estándar fue desarrollado para ser usado con cualquier ciclo de desarrollo o metodología y; la segunda es, que fue estructurado de tal forma que define un conjunto de elementos de proceso (objetivos, actividades, tareas, roles y, productos de trabajo) los cuales facilitan su implementación.

Lo que hace al estándar idóneo para ser utilizado por las EMPs es, que toma como base únicamente dos procesos: el proceso de Gestión del Proyecto y el proceso de Implementación del Software.

El proceso de Gestión del Proyecto establece y lleva a cabo de manera sistematizada, las tareas relacionadas con la gestión del proyecto, con la finalidad de lograr los objetivos del proyecto dentro de la calidad, tiempo y costos esperados. Este proceso está compuesto por 4 actividades: planificación del proyecto, ejecución del plan del proyecto, evaluación y control del proyecto y, cierre del proyecto (Muñoz, Mejía, & Laporte, 2018).

El proceso de Implementación de Software realiza de manera sistematizada, las actividades relacionadas con el desarrollo de software como: el análisis, el diseño, la construcción, la integración y, las pruebas; de acuerdo con los requisitos especificados, ya sea para el desarrollo de un nuevo producto o la modificación de un producto. Este proceso está compuesto por 6 actividades: inicio de la implementación, análisis de requisitos, arquitectura y diseño detallado, construcción, integración y pruebas y, entrega del producto (Muñoz, Mejía, & Laporte, 2018).

3. Análisis de debilidades

Esta sección muestra las debilidades identificadas en las EMPs obtenidas mediante la realización de una revisión sistemática y del estado de la práctica (resultado de un

análisis diagnóstico realizado a un conjunto de 13 EMPs que han implementado el estándar ISO/IEC 29110).

3.1. Debilidades identificadas como resultado de la ejecución de la revisión sistemática

La obtención de las debilidades que presentan las EMPs de desarrollo de software, al implementar el estándar ISO/IEC 29110, se llevó a cabo mediante el análisis de los datos obtenidos al ejecutar una revisión sistemática para la caracterización del estado del arte de la implementación del estándar ISO/IEC 29110, enfocada en los siguientes aspectos: impactos positivos, herramientas de software utilizadas para dar soporte a la implementación y debilidades que presentan las EMPs. Cabe resaltar que, para llevar a cabo esta revisión sistemática, se siguió el protocolo de Kitchenham y Charters (Kitchenham & Charters, 2007).

Como resultado, se analizaron 111 estudios de caso, los cuales fueron descritos a través de 7 estudios primarios (Laporte & O'Connor, 2016), (Larrucea & Santamaria, 2016), (Muñoz, Mejía, & Laporte, 2019), (Wongsai, Siddoo, & Wetprasit, 2015), (Laporte & O'Connor, 2017), (Muñoz, Mejía, & Lagunas, 2018), (Muñoz, Mejía & Laporte 2018) (listados en la Tabla 1). En ellos, se identificaron 5 debilidades, como se muestra en la Figura 1.

Título de estudio Primario	No. de estudios de caso
A Multi-case Study Analysis of Software Process Improvement in Very Small Companies Using ISO/IEC 29110 (Laporte & O'Connor, 2016)	3
Towards a survival Analysis of Very Small Organizations (Larrucea & Santamaria, 2016)	74
Reinforcing Very Small Entities Using Agile Methodologies with the ISO/IEC 29110 (Muñoz, Mejía & Laporte, 2019)	4
Factors of influence in software process improvement: An ISO/IEC 29110 for very-small entities (Wongsai, Siddoo & Wetprasit, 2015)	15
Implementing process improvement in very small enterprises with ISO/IEC 29110: A multiple case study analysis (Laporte & O'Connor, 2017)	6
Implementation of the ISO/IEC 29110 standard in agile environments: A systematic literature review (Muñoz, Mejía & Lagunas, 2018)	5
Implementation of ISO/IEC 29110 in Software Development Centers from Mexican Universities: An experience of the Zacatecas State (Muñoz, Mejía & Laporte, 2018)	4

Tabla 1 – Estudios primarios analizados

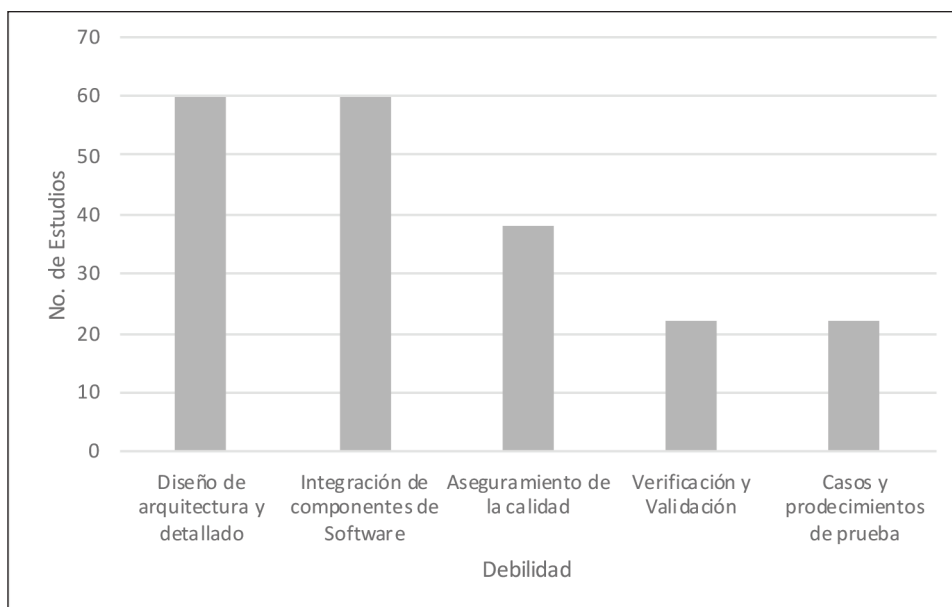


Figura 1 – Debilidades detectadas en los estudios de caso contenidos en estudios primarios.

3.2. Debilidades identificadas como resultado del análisis del estado de la práctica

La identificación de las debilidades, que presentan las EMPs en la práctica, se realizó con la aplicación de un análisis diagnóstico a un conjunto de 13 EMPs, en la que participaron empresas y centros de desarrollo de software, a los que se les proporcionó acompañamiento para la implementación del estándar ISO/IEC 29110. La lista de EMPs participantes en el estado de la práctica se listan en la Tabla 2.

# EMP	Tipo*	No. Miembros	Metodología usada
EMP1	EDS	2	Ninguna, pero usan prácticas ágiles
EMP2	CDS	5	Hibrida: TSP-Scrum
EMP3	CDS	5	Scrum
EMP4	EDS	4	Scrum
EMP5	CDS	5	Scrum
EMP6	EDS	5	Scrum
EMP7	EDS	5	Hibrida: CMMI-Scrum
EMP8	CDS	5	Cascada
EMP9	CDS	4	TSP

# EMP	Tipo*	No. Miembros	Metodología usada
EMP10	CDS	3	Cascada
EMP11	EDS	4	Hibrida: CMMI-Scrum
EMP12	CDS	5	Metodología basada en el modelo CMMI
EMP13	CDS	4	TSP

* En la columna de tipo pueden identificarse dos valores: Empresa de Desarrollo de Software (EDS) y Centro de Desarrollo de Software (CDS).

Tabla 2 – Lista de EMPs

El proceso seguido para el acompañamiento de estas EMPs está publicado en (Muñoz, Mejía, & Laporte, 2019) y (Muñoz, Mejía, & Laporte, 2019b) para empresas de desarrollo de software y, en (Muñoz, Mejía, & Laporte, 2018) para Centros de Desarrollo de Software.

El proceso seguido para el acompañamiento se muestra en la Figura 2. Como se observa en la figura, el proceso consta de 6 actividades, a través de las cuales las EMPs logran reforzar sus procesos de desarrollo de software, tanto de gestión del proyecto, como de desarrollo de software. Además, han obtenido la certificación en el estándar.

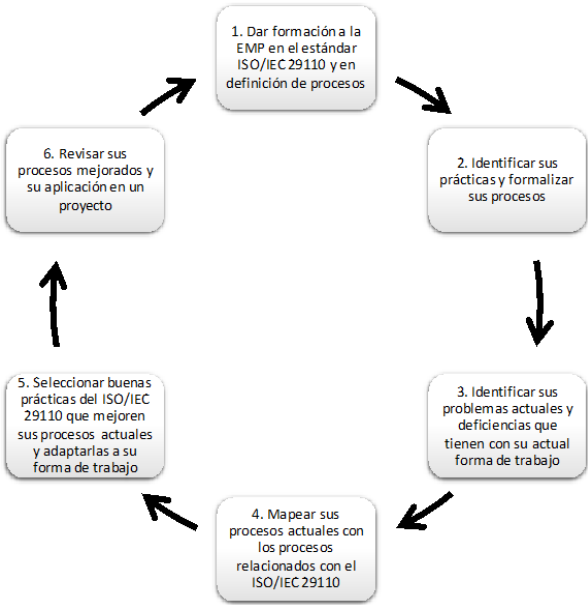


Figura 2 – Proceso implementado para el acompañamiento de las Entidades Muy Pequeñas.

Este artículo se centra en los resultados de los pasos 2 y 3 del proceso, permitiendo la identificación de las debilidades en las EMPs. Es importante mencionar, que las debilidades están asociadas a las prácticas que no son realizadas en las organizaciones.

Las principales debilidades identificadas se muestran en la figura 3. Cabe resaltar, que el principal criterio tomado en cuenta para la selección de una debilidad es, que se identificara en al menos 9 EMPs: (a) corregir los defectos encontrados hasta que la prueba unitaria sea pasada con éxito; (b) realizar actividades de verificación y validación; (c) desarrollar/actualizar casos y procedimientos de prueba; (d) realizar/actualizar registros de trazabilidad; (e) documentar la versión preliminar del manual de usuario; (f) realizar respaldo del repositorio del proyecto; (g) documentar la estrategia de control de versiones; (h) conducir reuniones con el cliente, registrando acuerdos y siguiéndolos hasta su cierre; (i) evaluar el progreso del proyecto frente al plan del proyecto; (j) realizar solicitudes de cambio; (k) incluir la descripción del producto, alcance, objetivos y entregables en el plan del proyecto; (l) establecer/actualizar entornos de desarrollo y; (m) desarrollar/actualizar pruebas unitarias.

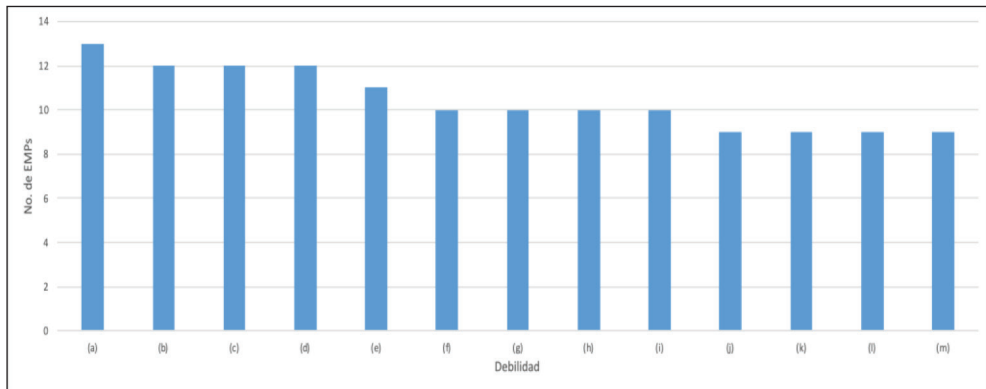


Figura 3 – Principales debilidades detectadas en la práctica.

Como se observa en la figura, en 13 EMPs analizadas se identifica a la actividad relacionada con la *corrección de defectos encontrados hasta que la prueba unitaria sea pasada con éxito*, como la debilidad que más se repite en las EMPs; en 12 EMPs analizadas, se identifican a las actividades relacionadas con la *verificación y validación*, así como el *manejo de registros de trazabilidad* como debilidades. En 11 EMPs analizadas, se identifica a la actividad relacionada con la documentación de la *versión preliminar del manual de usuario* como debilidad. En 10 EMPs analizadas, se identifican como debilidades: la *documentación de la estrategia de control de versiones*, la *realización de respaldos al repositorio del proyecto*, la *conducción de reuniones con el cliente, registrando acuerdos y, siguiéndolos hasta su cierre y*, la *evaluación del progreso del proyecto frente al plan* como debilidades. Finalmente, en 9 EMPs analizadas, se identifica la realización de *solicitudes de cambio*; la inclusión de datos relacionados con: la *descripción del producto, alcance, objetivos y entregables*; el *establecimiento o actualización de entornos de desarrollo y*; el *desarrollo o actualización de pruebas unitarias* como debilidades.

4. Comparativa entre los resultados obtenidos

En esta sección se muestra una comparativa de las debilidades que fueron identificadas con la realización de la revisión sistemática y, las debilidades que se identificaron como resultado del estado de la práctica, obtenido de un conjunto de EMPs de la región de Zacatecas, México.

Esta comparativa, tiene como finalidad, lograr un entendimiento de las principales debilidades que presentan las EMPs al inicio de la implementación del estándar ISO/IEC 29110. Las debilidades identificadas, pueden proporcionar una lista de requisitos a tener en cuenta para el desarrollo de herramientas de soporte, que apoyen a las EMPs, facilitando la implementación del estándar.

4.1. Comparativa de debilidades

Para lograr un mejor entendimiento de las debilidades presentadas, éstas se clasificarán a nivel de actividades para los dos procesos que contiene el perfil básico del estándar ISO/IEC 29110: Proceso de Gestión del Proyecto y Proceso de Implementación de Software.

La Tabla 3 muestra la clasificación de debilidades identificadas para las actividades del proceso de Gestión del Proyecto, y la Tabla 4, muestra la clasificación de debilidades identificadas para las actividades del proceso de Implementación de Software.

Actividad	Debilidad identificada en la revisión sistemática	Debilidad identificada en el estado de la práctica
Planificación del proyecto	• Aseguramiento de la calidad • Verificación y validación	• Incluir la descripción del producto, alcance, objetivos y entregables en el plan del proyecto • Documentar la estrategia de control de versiones • Verificación y validación
Ejecución del plan del proyecto		• Conducir reuniones con el cliente, registrando acuerdos y seguirlos hasta su cierre • Realizar respaldo de repositorio del proyecto
Evaluación y control del proyecto		• Evaluar el progreso del proyecto frente al plan del proyecto • Realizar solicitudes de cambio
Cierre del proyecto		

Tabla 3 – Debilidades identificadas en el Proceso de Gestión del Proyecto.

Como se observa en la Tabla 3 la debilidad en común mencionada en el estado del arte y en el estado de la práctica es: la falta de ejecución de *actividades relacionadas con la verificación y validación*. Sin embargo, se puede observar, que en el estado de la práctica han sido identificadas más debilidades en el proceso de Gestión del Proyecto, como a continuación se describen brevemente:

Actividad 1. Planificación del proyecto: las debilidades que se detectaron, están relacionadas con el plan del proyecto. La primera debilidad identificada en 9 EMPs es, la omisión de información (descripción del producto, alcance del producto, objetivos y entregables) en el plan del proyecto, esta información debe incluirse ya que el plan es validado por el cliente. La segunda debilidad, identificada en 10 EMPs es, que no se documenta la estrategia de control de versiones, dónde se especifica cómo se va almacenar el proyecto, cómo se van a realizar los respaldos, cómo se puede realizar su recuperación ante desastres, así como el estándar de nombrado de los artefactos del proyecto y el control de versiones.

Actividad 2. Ejecución del plan del proyecto: en esta actividad, se identificaron dos debilidades en 10 EMPs. La primera, está relacionada con la falta de realización de reuniones con el cliente, y/o el registro de acuerdos en actas del proyecto, tal que, se les pueda dar seguimiento a los acuerdos hasta su cierre. La segunda, está relacionada con la falta de realización de respaldos del repositorio del proyecto, tanto para el software como para la documentación, de acuerdo a lo indicado en la estrategia de control de versiones.

Actividad 3. Evaluación y control del proyecto: en esta actividad, se identificaron dos debilidades en 10 EMPs. La primera, relacionada con la falta de realización de reportes de seguimiento, en los que se analicen los datos planificados frente a los datos reales, incluyendo: tareas, objetivos, recursos, costos, tiempos y riesgos. La segunda, está relacionada con la falta de uso de solicitudes de cambio, en las que se realice una evaluación de cada cambio analizando, el impacto técnico, en tiempo y en costo; tal que, el resultado de este análisis pueda ser utilizado para la toma de decisiones adecuada sobre cada cambio solicitado.

Actividad	Debilidad identificada en la revisión sistemática	Debilidad identificada en el estado de la práctica
Inicio de la implementación		• Establecer o actualizar entornos de desarrollo
Análisis de Requisitos	• Aseguramiento de la calidad • Verificación y validación	• Verificación y validación • Documentar versión preliminar de manual de usuario
Arquitectura y diseño detallado	• Diseño de arquitectura y diseño detallado • Verificación y validación	• Verificación y validación • Desarrollar o actualizar registro de trazabilidad
Construcción		• Desarrollar o actualizar pruebas unitarias • Corregir defectos encontrados hasta que la prueba unitaria sea pasada con éxito • Desarrollar o actualizar registro de trazabilidad
Integración y pruebas de software	• Casos y procedimientos de prueba • Integración de componentes de software • Aseguramiento de la calidad	• Casos y procedimientos de prueba • Desarrollar o actualizar registro de trazabilidad
Entrega del producto		

Tabla 4 – Debilidades identificadas en el Proceso de Implementación de Software.

Como se observa en la Tabla 4, para el proceso de Implementación de Software, se tiene un número mayor de debilidades en común mencionadas en el estado del arte y en el estado de la práctica, nuevamente resaltan, la falta de ejecución de *actividades relacionadas con la verificación y validación* y, el *desarrollo y/o documentación de casos o procedimientos de prueba*. Además, se consideran relacionados: *la falta del desarrollo o actualización de pruebas unitarias* y *la corrección de defectos hasta que la prueba unitaria sea pasada con éxito*.

Nuevamente, se puede observar, que en el estado de la práctica se han identificado más debilidades en este proceso como a continuación se describe brevemente:

Actividad 1. Inicio de la implementación: la debilidad que se identifica en 9 EMPs es la falta de establecimiento o actualización del entorno de desarrollo.

Actividad 2. Análisis de requisitos: la debilidad identificada en 11 EMPs está relacionada con la falta de elaboración de una versión preliminar del manual de usuario, tal que este pueda ser validado por el cliente.

Actividad 3. Arquitectura y diseño detallado: la debilidad identificada en 12 EMPs está relacionada con la falta de la realización de registros de trazabilidad (requisito-arquitectura/diseño detallado). Es importante mencionar que, para esta actividad en el estado del arte, se identificó además la falta de realización del diseño arquitectural y diseño detallado.

Actividad 4. Construcción: la debilidad identificada en 12 EMPs está relacionada con la falta de la realización de registros de trazabilidad (requisito-arquitectura/diseño detallado-código-prueba unitaria). Es importante mencionar que, para esta actividad, en el estado de la práctica, se identificó además la falta de realización de pruebas unitarias.

Actividad 5. Integración y pruebas: la debilidad identificada en 12 EMPs está relacionada con la falta de la realización de registros de trazabilidad (requisito-arquitectura/diseño detallado-código-prueba unitaria-pruebas de integración). Es importante mencionar que, para esta actividad en el estado del arte, se identificó la falta de realización de la integración de componentes de software.

5. Discusión, conclusiones y trabajo futuro

La creciente demanda de software en diferentes dominios brinda una gran oportunidad al sector de la industria del desarrollo de software. En especial, a las Entidades Muy Pequeñas, ya que éstas producen software para: pequeñas, medianas y grandes empresas. Por lo tanto, se hace indispensable, que éstas refuercen sus procesos de desarrollo de software, tal que, sean capaces de desarrollar el software requerido por sus clientes, con la calidad y dentro de los tiempos y presupuestos establecidos.

En este contexto, la serie de estándares ISO/IEC 29110, ha sido creada para contribuir con el desarrollo de productos de calidad dentro del presupuesto y calendario, a la vez, que se optimicen esfuerzos y recursos.

Sin embargo, debido a la falta de experiencia en la implementación de modelos y estándares, se dificulta la implementación de buenas prácticas contenidas en modelos y estándares en este tipo de empresas.

Este artículo presenta una comparativa entre las debilidades mencionadas en la literatura y las identificadas en la práctica, al analizar un conjunto de 13 EMPs del Estado de Zacatecas, México.

Los resultados muestran que en estas EMPs, deben fortalecer, principalmente, la realización de prácticas enfocadas en el aseguramiento de la calidad, en las que se listan las actividades de verificación y validación, a través del desarrollo de software, resaltando: el plan de proyecto, los requisitos, el diseño, los manuales de usuario, de operación y de mantenimiento, la realización de casos y procedimientos de pruebas (ambas unitarias y de integración), y la realización de solicitudes de cambio.

Además, deben ser realizados registros de trazabilidad, que permitirán identificar el cumplimiento de cada requisito en: el diseño, la codificación y las pruebas. Finalmente, otro aspecto a fortalecer, es la realización de informes de seguimiento, que permitan analizar el progreso del proyecto y tomar decisiones adecuadas en el tiempo y forma.

La identificación de estas debilidades puede tomarse como base en el desarrollo de herramientas, que brinden el soporte requerido por las Entidades Muy Pequeñas para minimizar los esfuerzos y resistencia a la mejora de sus procesos.

Finalmente, cabe mencionar que, como trabajo futuro con base en las debilidades identificadas, se está desarrollando una herramienta, que proporcionará soporte a las Entidades Muy Pequeñas en la adopción del estándar ISO/IEC 29110.

Referencias

- ISO/IEC (2011). ISO/IEC TR 29110-5-1-2:2011, Software Engineering—Lifecycle Profiles for Very Small Entities (VSEs)—Parte 5-1-2: Management and Engineering Guide: Generic Profile Group: Basic Profile, Int'l Org. for Standardization. ISO/IEC.
- Kitchenham, B., & Charters, S. (2017). Guidelines for performing Systematic Literature Reviews in Software Engineering. EBSE Technical report. EBSE.
- Laporte, C.Y., Muñoz, M., & Gerançon, B. (2017). The education of students about ISO/IEC 29110 software engineering standards and their implementations in very small entities. In IEEE Canada-International Humanitarian Technology Conference, 94–98, Toronto, Ontario, Canadá.
- Laporte, C.Y., & O'Connor, R.V. (2016). A Multi-case Study Analysis of Software Process Improvement in Very Small Companies Using ISO/IEC 29110. In: Kreiner C., O'Connor R., Poth A., Messnarz R. (Eds) Systems, Software and Services Process Improvement. EuroSPI 2016. Communications in Computer and Information Science: Vol 633. Cham: Springer. <https://doi.org/10.1007/978-3-319-44817-6>.

- Laporte, C. Y., & O'Connor, R. V. (2017). Implementing process improvement in very small enterprises with ISO/IEC 29110: A multiple case study analysis. In *Proceedings 2016 10th International Conference on the Quality of Information and Communications Technology (QUATIC)* (pp. 125–130). <https://doi.org/10.1109/QUATIC.2016.033>.
- Larrucea, X., & Santamaria, I. (2016). Towards a survival Analysis of Very Small Organisations. In *23nd European Conference on Systems, Software and Services Process Improvement (EuroSPI 2016)* (pp. 599–609). <https://doi.org/10.1007/978-3-319-44817-6>.
- Muñoz, M., Mejía, J., & Lagunas, A. (2018). Implementation of the ISO/IEC 29110 standard in agile environments: A systematic literature review. In *Iberian Conference on Information Systems and Technologies, CISTI, Cáceres, España*. (pp. 1–6). <https://doi.org/10.23919/CISTI.2018.8399332>.
- Muñoz, M., Mejía, J., & Laporte, C. Y. (2018). Implementation of ISO/IEC 29110 in Software Development Centers from Mexican Universities: An experience of the Zacatecas State. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, (29), 43–54. <https://doi.org/10.17013/risti.29.43-54>.
- Muñoz, M., Mejía, J., & Laporte, C.Y. (2019a) Reinforcing Very Small Entities Using Agile Methodologies with the ISO/IEC 29110. In: Mejia J., Muñoz M., Rocha Á., Peña A., Pérez-Cisneros M. (Eds.) *Trends and Applications in Software Engineering. CIMPS 2018. Advances in Intelligent Systems and Computing: Vol 865*. Cham: Springer. <https://doi.org/10.1007/978-3-030-01171-0>.
- Muñoz, M., Mejía, J., & Laporte, C. Y. (2019b) Implementing ISO/IEC 29110 to Reinforce Four Very Small Entities of Mexico under an agile approach. *IET Software*, 1-11. DOI: 10.1049/iet-sen.2019.0040.
- Software Engineering Institute (2010). *CMMI for Development, Version 1.3*. Pittsburgh, PA: Carnegie Mellon University.
- Wongsai, N., Siddoo, V., & Wetprasit, R. (2015). Factors of influence in software process improvement: An ISO/IEC 29110 for Very-Small Entities. In *Proceedings 2015 7th International Conference on Information Technology and Electrical Engineering: Envisioning the Trend of Computer, Information and Engineering, ICITEE 2015*. (pp. 12–17). <https://doi.org/10.1109/ICITEED.2015.7408904>

Métricas para la Medición del Factor de Flexibilidad y el Factor de Herencia de Implementación de Sistemas de Software

Orlando Ortiz-Gutierrez, René Santaolaya-Salgado, Olivia-Graciela Fragoso-Díaz, Juan-Carlos Rojas-Pérez

orlando.ortiz18ce@cenidet.edu.mx, rene@cenidet.edu.mx, ofragoso@cenidet.edu.mx, carlos.rojas@cenidet.edu.mx

Tecnológico Nacional de México/CENIDET, Depto. Ciencias Computacionales, Interior Internado Palmira S/N, Col. Palmira, C.P. 62490, Cuernavaca, Morelos, México.

DOI: 10.17013/risti.34.97-111

Resumen: Actualmente la medición de la calidad del diseño arquitectural de software es un tópico importante para los ingenieros de software. Sin embargo, algunas métricas de calidad aún no están definidas, como lo son las métricas para medir el factor de flexibilidad y de herencia de implementación de una arquitectura de clases, enfocadas en métodos abstractos. En este trabajo de investigación se propone un conjunto de cinco métricas de calidad, tres de ellas son para medir el factor de herencia de implementación, las cuales son: Factor de Herencia de Implementación (FHI), Factor de Herencia de Implementación por Jerarquía de Clases (FHIJ) y Factor Herencia de Implementación de una Arquitectura de Clases (FHIAC); las dos restantes son para medir el factor de flexibilidad, estas son: Factor de Flexibilidad de Clase (FFC) y Factor Medio de Flexibilidad de Clases (FMFC).

Palabras-clave: Acoplamiento; flexibilidad; herencia de interfaz; herencia de implementación.

Metrics for the Measurement of the Flexibility Factor and the Inheritance Implementation Factor of Software Systems

Abstract: Currently measuring the quality of architectural software design is an important topic for software engineers. However, some quality metrics are not yet defined such as the metrics for measuring the flexibility and implementation inheritance factor for a class architecture focused on abstract methods. In this research, a set of five quality metrics is proposed; three of them are to measure the inheritance implementation factor, which are: Implementation Inheritance Factor (FHI), Implementation Inheritance Factor by Class Hierarchy (FHIJ) and Implementation Inheritance Factor of a Class Architecture (FHIAC); the remaining two are to measure the flexibility factor, these are: Class Flexibility Factor (FFC) and Average Class Flexibility Factor (FMFC).

Keywords: Coupling; flexibility; interface inheritance; implementation inheritance.

1. Introducción

Actualmente la medición de la calidad del diseño arquitectural de software es un tópico importante para los ingenieros de software. Sin embargo, algunas métricas de calidad aún no están definidas, como lo son las métricas para medir el factor de flexibilidad y de herencia de implementación de una arquitectura de clases. Entendiéndose por flexibilidad, como la facilidad con la que un sistema o componente puede modificarse para su uso en aplicaciones o entornos distintos de aquellos para los que fue específicamente diseñado (IEEE Standard Glossary of Software Engineering Terminology, 1990) y por herencia de implementación, una relación donde una subclase hereda la implementación del comportamiento de su clase base (Pinto, 2015). Esta carencia reprime la identificación de código desagradable (del Inglés *smell code*), en el software legado, que impacta en anti patrones de diseño e impide su mejora por medio de técnicas de refactorización y de reingeniería.

Para efectos de reducir el acoplamiento entre clases relacionadas por herencia de implementación en arquitecturas orientadas a objetos se necesita, una métrica que mida el acoplamiento por herencia de implementación y otra métrica para medir la flexibilidad, de un diseño arquitectural de software.

Se han revisado diferentes trabajos de investigación sobre métricas para medir algunos atributos de calidad, como lo son: acoplamiento, reusabilidad, flexibilidad, comprensibilidad, funcionalidad, extensibilidad y efectividad. Sin embargo, en la literatura revisada no se han encontrado métricas cuyo propósito sea medir el factor de flexibilidad y de herencia de implementación desde métodos abstractos, por ejemplo: en (Chawla & Nath, 2013) se mide la herencia y el acoplamiento a nivel de clase y de sistema, pero ninguna de las métricas toman en cuenta métodos abstractos; en (Chawla & Chhabra, 2013) se mide la flexibilidad utilizando una fórmula que utiliza los atributos de encapsulación, acoplamiento, composición y polimorfismo pero tampoco ninguna de las métricas utilizadas toman en cuenta los métodos abstractos. Consecuentemente, en esta investigación se definió un conjunto de métricas para la medición de la flexibilidad y el acoplamiento por herencia de implementación desde clases abstractas, utilizando la teoría de la medición para sustentarlas.

2. Marco Teórico

Recientemente se ha sugerido que la teoría de la medición debería servir de base para desarrollar, razonar y aplicar métricas de ingeniería de software (Briand, El Emam, & Morasca, 1996).

Para clasificar una métrica como de escala ordinal, se deben cumplir los requisitos del axioma de *orden débil*, los cuales son: que $\bullet \geq$ sea una relación binaria total y transitiva. Las propiedades de transitividad y completitud, son las siguientes (Suze, 1995):

1. *Transitividad*: $P \bullet \geq P', P' \bullet \geq P'' \Rightarrow P \bullet \geq P''$
2. *Completitud*: $P \bullet \geq P' \vee P' \bullet \geq P$

para todo $P', P'' \in P$, donde P es un conjunto y $\bullet \geq$ es una relación empírica binaria como “igual o más compleja que”.

Supóngase que $(P, \bullet \geq)$ es un sistema relacional empírico, donde P es un conjunto contable no vacío y $\bullet \geq$ es una relación binaria en P . Luego existe una función $\mu: P \rightarrow R$, con:

$$P' \bullet \geq P'' \leftrightarrow \mu(P') \geq \mu(P''),$$

para todo $P', P'' \in P$, si y sólo si, $\bullet \geq$ es de orden débil. Si tal homomorfismo existe, entonces, $((P, \bullet \geq), (R, \geq) \mu)$ es una escala ordinal. Un homomorfismo es una función que preserva las operaciones definidas en los objetos de la función. Este término proviene de las palabras griegas para mismo (homos) y forma (morphe). La medida μ en una escala es un homomorfismo.

3. Materiales y Métodos

3.1. Escalas o Niveles de Medición

Una asociación que se establece entre el mundo real y valores de medida, se suele denominar *escala de medición*. La teoría de la medición permite formalizar la relación entre dos escalas de medición con esta definición (Gary, 2010):

Sea (A, B, μ) una escala, donde el conjunto de objetos en B , es el conjunto de los números reales. Sea que la notación $\mu(A)$ significa que el conjunto de todos los números reales que son mediciones de algún objeto en A . Esto es que $\mu(A)$ es el conjunto de medidas de los objetos en A . (En matemáticas, a esto se la llama el rango de μ). Luego un mapeo $t: \mu(A) \rightarrow B$ se define como una *transformación admisible*, si y sólo si, $(A, B, t \circ \mu)$ también es una escala.

Se pueden describir cinco tipos de escalas que se caracterizan por sus transformaciones admisibles (Gary, 2010): nominal, ordinal, de intervalo, de proporción (ratio) y absoluta.

Específicamente, las escalas ordinales asignan números a objetos en un orden particular, pero cualquier número que mantiene el orden es igualmente bueno. Cualquier función estrictamente creciente de t , es una transformación admisible. Un ejemplo es la escala de Mohs para la dureza de los minerales. La escala original asigna, por ejemplo, 1 al talco, 7 al cuarzo y 10 al diamante. Años después, se creó una escala que asignó 1 al talco, 8 al cuarzo y 15 al diamante. Los números difieren, pero el orden sigue siendo el mismo. En este caso se cumple que aun cuando se transforme la escala no cambia el orden creciente, por lo tanto, es una transformación admisible (Gary, 2010). Esta es la escala en la que se definen las métricas propuestas, puesto que cumplen con los axiomas de la teoría de la medición para pertenecer a esta escala.

3.2. Factor de Herencia de Implementación (FHI) como escala ordinal

Se tiene el siguiente sistema relacional empírico:

1. P es el conjunto de una clase base y una clase derivada.
2. $\bullet \geq$ es una relación empírica entre clases que describe que una clase tiene mayor o igual factor de herencia de implementación que otra.
3. $\mathfrak{R}$ denota el conjunto de los números reales.
4. $\geq$ “mayor o igual que” es una relación binaria entre números.

Entonces $((P, \bullet \geq), (\mathfrak{R}, \geq), \text{FHI})$ es una escala ordinal si, y solo si, se cumplen las siguientes condiciones:

1. La relación binaria $\bullet \geq$, es de orden débil.
2. Clase1 $\bullet \geq$ Clase2 $\Leftrightarrow$ FHI(Clase1) $\geq$ FHI(Clase2)

Como parte del proceso de comprobación de las condiciones anteriores, se utiliza la ecuación (1) de la sección 4.1 para realizar el cálculo del FHI de las clases uno, dos y tres, ubicadas en la Figura 1. En donde los métodos que tienen “= 0”, significa que son abstractos, y los que no lo tienen, quiere decir que tienen una implementación en su cuerpo. Obteniéndose los siguientes resultados de la Figura 1:

- $FHI(Clase1) = \frac{\sum Mv + \sum Mnv}{\sum Tm} = 2/3 = 0.666...$
- $FHI(Clase2) = \frac{\sum Mv + \sum Mnv}{\sum Tm} = 2/4 = 0.5$
- $FHI(Clase3) = \frac{\sum Mv + \sum Mnv}{\sum Tm} = 2/5 = 0.4$

Clase1	Clase2	Clase3
funcionA() funcionB() funcionC() = 0	funcionA() funcionB() funcionC() = 0 funcionD() = 0	funcionA() = 0 funcionB() = 0 funcionC() = 0 funcionD() funcionE()

Figura 1 – Arquitectura de clases de un sistema.

Para comprobar que la relación binaria es de orden débil, tiene que cumplir con las propiedades de transitividad y completitud.

Transitividad

$$\text{Clase1} \bullet \geq \text{Clase2}, \text{Clase2} \bullet \geq \text{Clase3} \rightarrow \text{Clase1} \bullet \geq \text{Clase3}$$

$\Leftrightarrow$

$$FHI(Clase1) \bullet \geq FHI(Clase2) \ \& \ FHI(Clase2) \bullet \geq FHI(Clase3) \rightarrow FHI(Clase1) \bullet \geq FHI(Clase3)$$

Reflejado en números, se tiene que:

$$0.666... \geq 0.5 \ \& \ 0.5 \geq 0.4 \rightarrow 0.666... \geq 0.4$$

Como se puede observar, la clase1 tiene un factor de herencia de implementación mayor al de la clase2. Este a su vez tiene un factor de herencia de implementación mayor al de la clase3. Concluyendo que la relación binaria $\bullet \geq$ cumple con la propiedad de transitividad.

Relación Total

Si se tienen las clases uno y dos, se debe poder decir que la clase 1 “tiene mayor o igual factor de herencia de implementación que” la clase 2, o viceversa. Es decir:

$$\text{Clase1} \bullet \geq \text{Clase2} \text{ o } \text{Clase2} \bullet \geq \text{Clase1}$$

Calculando el FHI en las clases uno y dos de la Figura 1, siempre fue posible determinar cuando existía mayor o igual factor de herencia de implementación.

Homomorfismo

Comprobando la segunda condición quedaría de la siguiente manera:

La clase 1 “tiene mayor o igual factor de herencia de implementación que” la clase 2

$$\Leftrightarrow 0.666... \geq 0.5$$

Conclusión

La relación binaria “tiene mayor o igual factor de herencia de implementación que” de la métrica FHI, cumple con las condiciones de orden débil y además es un homomorfismo. Por lo cual, se concluye que la métrica es de escala ordinal (Suze, 1992).

Bajo el mismo sustento de la teoría de la medición se diseñaron las métricas para el Factor de Herencia de Implementación por Jerarquía de Clases (FHIJ) como Escala Ordinal y Factor de Herencia de Implementación de una Arquitectura de Clases (FHIAC) como escala ordinal.

3.3. Factor de Herencia de Implementación por Jerarquía de Clases (FHIJ) como Escala Ordinal

Bajo el mismo sustento de la teoría de la medición se diseñó la métrica para el Factor de Herencia de Implementación por Jerarquía de Clases (FHIJ) como Escala Ordinal. Obteniéndose los siguientes resultados al aplicar la ecuación (2) de la sección 4.2, a las jerarquías de clase mostradas en la Figura 2:

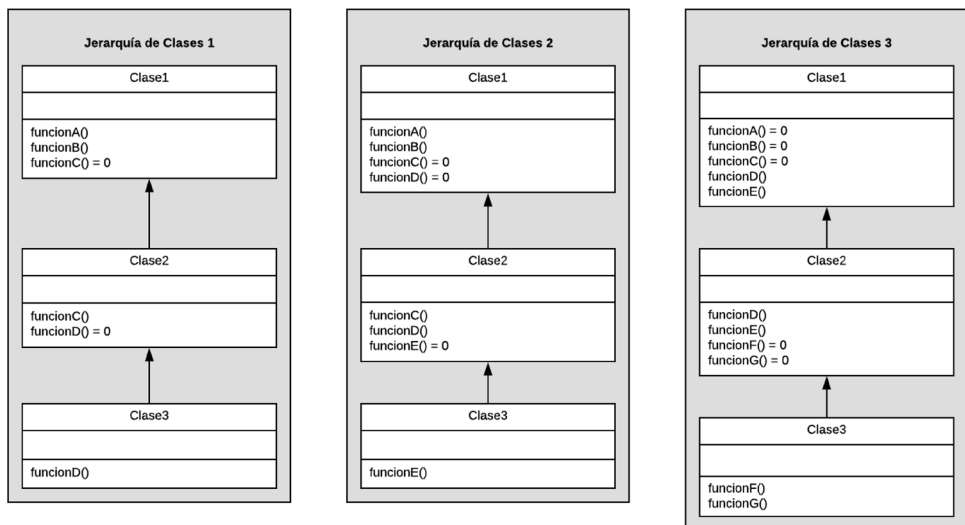


Figura 2 – Conjunto de jerarquías de clases.

- $$\text{FHIJ}(\text{JerarquiaClase1}) = \frac{\Sigma \text{FHI}}{\Sigma \text{Tc}-1} = \frac{(2/3 + 1/2)}{3-1} = \frac{1.166...}{2} = 0.583...$$
- $$\text{FHIJ}(\text{JerarquiaClase2}) = \frac{\Sigma \text{FHI}}{\Sigma \text{Tc}-1} = \frac{(2/4 + 2/3)}{3-1} = \frac{1.166...}{2} = 0.583...$$
- $$\text{FHIJ}(\text{JerarquiaClase3}) = \frac{\Sigma \text{FHI}}{\Sigma \text{Tc}-1} = \frac{(2/5 + 2/4)}{3-1} = \frac{0.9}{2} = 0.45$$

Conclusión

La relación binaria “*tiene mayor o igual factor de herencia de implementación que*” de la métrica FHIJ, cumple con las condiciones de orden débil y además es un homomorfismo. Por lo cual, se concluye que esta métrica es de escala ordinal (Suze, 1992).

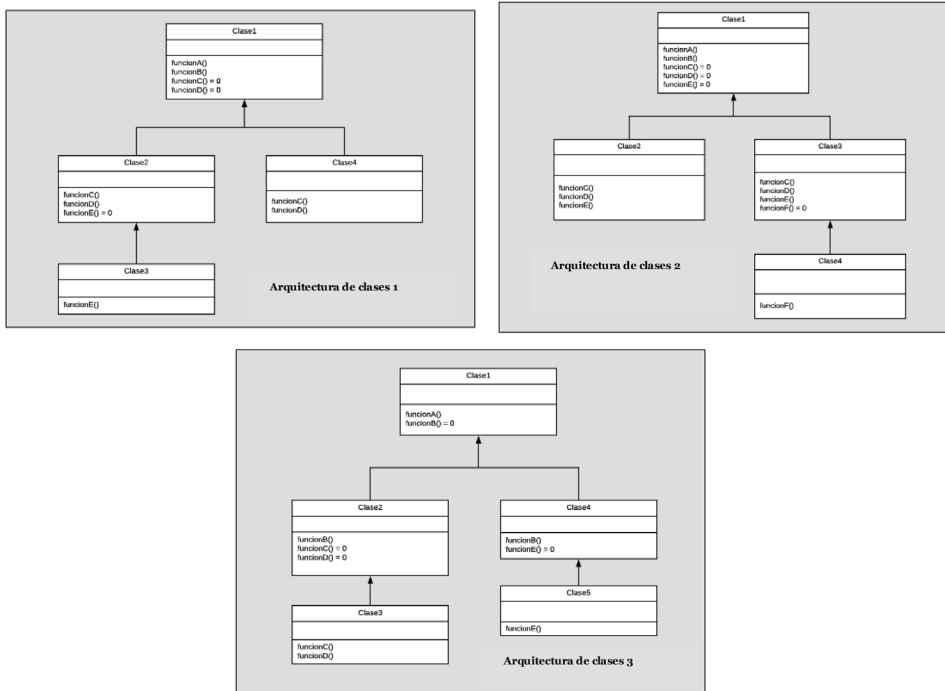


Figura 3 – Arquitecturas de clases.

3.4 Factor de Herencia de Implementación de una Arquitectura de Clases (FHIAC) como escala ordinal

Así mismo, se diseñó la métrica para el Factor de Herencia de Implementación de una Arquitectura de Clases (FHIAC) como escala ordinal. Obteniéndose los siguientes

resultados al aplicar la ecuación (3) de la sección 4.3, a las arquitecturas de clases mostradas en la Figura 3:

- $FHIAC(ArquitecturaClases1) = \frac{\Sigma FHIJ}{NOH} = \frac{(0.583...+0.5)}{2} = \frac{1.083...}{2} = 0.5416...$
- $FHIAC(ArquitecturaClases2) = \frac{\Sigma FHIJ}{NOH} = \frac{(0.4+0.575)}{2} = \frac{0.975}{2} = 0.4875$
- $FHIAC(ArquitecturaClases3) = \frac{\Sigma FHIJ}{NOH} = \frac{(0.416...+0.5)}{2} = \frac{0.916...}{2} = 0.4583...$

Conclusión

La relación binaria “*tiene mayor o igual factor de herencia de implementación que*” de la métrica FHIAC, cumple con las condiciones de orden débil y además es un homomorfismo. Por lo cual, se concluye que la métrica es de escala ordinal (Suze, 1992).

3.5. Factor de Flexibilidad de Clases (FFC) como escala ordinal

Para las métricas de flexibilidad se tiene el siguiente sistema relacional empírico:

1. P es el conjunto de clases de una arquitectura.
2. $\bullet \geq$ es una relación empírica binaria entre clases, que describe que una clase tiene mayor o igual factor de flexibilidad que otra.
3. $\mathbb{R}$ denota el conjunto de los números reales.
4. $\geq$ “*mayor o igual que*” es una relación binaria entre números.

Entonces $((P, \bullet \geq), (\mathbb{R}, \geq))$, FFC) es una escala ordinal si, y solo si, se cumplen las siguientes condiciones:

1. La relación binaria $\bullet \geq$, es de *orden débil*.
2. $Clase1 \bullet \geq Clase2 \Leftrightarrow FFC(Clase1) \geq FFC(Clase2)$

Como parte del proceso de comprobación de las condiciones anteriores, se utiliza la ecuación (4) de la sección 4.4 para realizar el cálculo del FFC de las clases uno, dos y tres, ubicadas en la Figura 4. En donde los métodos que tienen “= 0”, significa que son abstractos, y los que no lo tienen, quiere decir que tienen una implementación en su cuerpo. Obteniéndose los siguientes resultados de la Figura 4:

- $FFC(Clase1) = 3/5 = 0.6$
- $FFC(Clase2) = 2/4 = 0.5$
- $FFC(Clase3) = 1/3 = 0.333...$

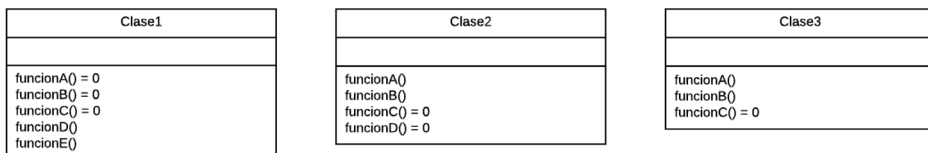


Figura 4 – Arquitectura de clases de un sistema.

Para comprobar que la relación binaria es de orden débil, tiene que cumplir con las propiedades de transitividad y completitud.

Transitividad

$$\text{Clase1} \bullet \geq \text{Clase2}, \text{Clase2} \bullet \geq \text{Clase3} \rightarrow \text{Clase1} \bullet \geq \text{Clase3}$$

$$\Leftrightarrow$$

$$\text{FFC}(\text{Clase1}) \bullet \geq \text{FFC}(\text{Clase2}) \ \& \ \text{FFC}(\text{Clase2}) \bullet \geq \text{FFC}(\text{Clase3}) \rightarrow \\ \text{FFC}(\text{Clase1}) \bullet \geq \text{FFC}(\text{Clase3})$$

Reflejado en números, se tiene que:

$$0.6 \geq 0.5 \ \& \ 0.5 \geq 0.333 \rightarrow 0.6 \geq 0.333$$

Como se puede observar, la clase 1 tiene un factor de flexibilidad de clase mayor al de la clase 2. Éste a su vez tiene un factor de flexibilidad mayor al de la clase 3, debido a la cantidad de métodos virtuales (con comportamiento polimórfico) que tienen. Concluyendo que la relación binaria $\bullet \geq$, cumple con la propiedad de transitividad.

Relación Total

Si se tienen las clases uno y dos, se debe poder decir que la clase 1 “tiene mayor o igual factor de flexibilidad de clase que” la clase 2, o viceversa. Es decir:

$$\text{Clase1} \bullet \geq \text{Clase2} \text{ o } \text{Clase2} \bullet \geq \text{Clase1}$$

Calculando el FFC en las clases uno y dos de la Figura 4 siempre fue posible determinar cuando existía mayor o igual factor de flexibilidad.

Homomorfismo

Comprobando la segunda condición quedaría de la siguiente manera:

La clase 1 “tiene mayor o igual factor de flexibilidad que” la clase 2

$$\Leftrightarrow$$

$$0.6 \geq 0.5$$

Demostración empírica de la propiedad de transitividad

$$\text{Clase1} \bullet \geq \text{Clase2}, \text{Clase2} \bullet \geq \text{Clase3} \rightarrow \text{Clase1} \bullet \geq \text{Clase3}$$

$$\Leftrightarrow$$

$$\text{FFC}(\text{Clase1}) \bullet \geq \text{FFC}(\text{Clase2}) \ \& \ \text{FFC}(\text{Clase2}) \bullet \geq \text{FFC}(\text{Clase3}) \rightarrow \\ \text{FFC}(\text{Clase1}) \bullet \geq \text{FFC}(\text{Clase3})$$

Demostración formal de la propiedad de transitividad

Reflejado en números, se tiene que:

$$0.6 \geq 0.5 \ \& \ 0.5 \geq 0.333 \rightarrow 0.6 \geq 0.333$$

Como se puede observar, la clase 1 tiene un factor de flexibilidad de clase mayor al de la clase 2. Éste a su vez tiene un factor de flexibilidad mayor al de la clase 3, debido a la cantidad de métodos virtuales con comportamiento polimórfico que tienen. Concluyendo que la relación binaria $\bullet \geq$, cumple con la propiedad de transitividad.

Demostración empírica de la propiedad de completitud

Si se tienen las clases uno y dos, se debe poder decir que la clase 1 “tiene mayor o igual factor de flexibilidad de clase que” la clase 2, o viceversa. Es decir:

$$\text{Clase1} \bullet \geq \text{Clase2} \text{ o } \text{Clase2} \bullet \geq \text{Clase1}$$

Demostración formal de la propiedad de completitud

Calculando el FFC en las clases uno y dos de la Figura 4 siempre fue posible determinar cuando existía mayor o igual factor de flexibilidad.

Demostración empírica de FFC que es un homomorfismo

Comprobando la segunda condición quedaría de la siguiente manera:

La clase 1 “tiene mayor o igual factor de flexibilidad que” la clase 2

Formalmente

$$0.6 \geq 0.5$$

Conclusión

La relación binaria “tiene mayor o igual factor de flexibilidad que” de la métrica FFC, cumple con las condiciones de orden débil y además es un homomorfismo. Por lo cual, se concluye que la métrica es de escala ordinal (Suze, 1992).

3.6. Factor Medio de Flexibilidad de Arquitecturas de Clases (FMFAC) como escala ordinal

Bajo el mismo sustento de la teoría de la medición se diseñó la métrica para Factor Medio de Flexibilidad de Arquitecturas de Clases (FMFAC) como escala ordinal.

Obteniéndose los siguientes resultados al aplicar la ecuación (5) de la sección 4.5, a las arquitecturas de clase mostradas en la Figura 5:

- $\text{FMFAC}(\text{ArquitecturaClases1}) = \frac{\Sigma \text{FFC}}{\Sigma \text{Tc}} = \frac{\frac{1}{2} + \frac{2}{3} + \frac{1}{2}}{5} = \frac{0.5 + 0.666... + 0.5}{5} = \frac{1.666...}{5} = 0.333...$
- $\text{FMFAC}(\text{ArquitecturaClases2}) = \frac{\Sigma \text{FFC}}{\Sigma \text{Tc}} = \frac{\frac{2}{4} + \frac{1}{3}}{4} = \frac{0.5 + 0.333...}{4} = \frac{0.833...}{4} = 0.2125$
- $\text{FMFAC}(\text{ArquitecturaClases3}) = \frac{\Sigma \text{FFC}}{\Sigma \text{Tc}} = \frac{\frac{2}{4} + \frac{1}{3}}{4} = \frac{0.5 + 0.333...}{4} = \frac{0.833...}{4} = 0.2083...$

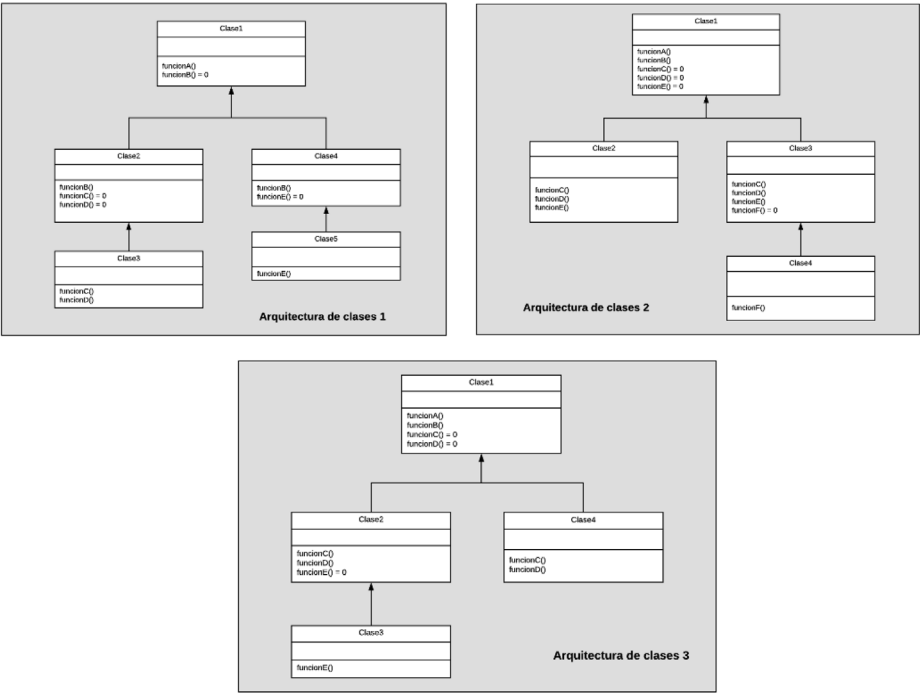


Figura 5 – Arquitectura de clases de un sistema.

Conclusión

La relación binaria “*tiene mayor o igual factor medio de flexibilidad de la arquitectura de clases*” de la métrica FMFAC, cumple con las condiciones de orden débil y además es un homomorfismo. Por lo cual, se concluye que la métrica es de escala ordinal (Suze, 1992).

4. Resultados

Se obtuvo un conjunto de métricas para medir el factor de flexibilidad (FFC y FMFAC) y de herencia de implementación (FHI, FHIJ y FHIAC), en diseños arquitecturales de clases, descritas más a detalle a continuación:

4.1. Métrica FHI (Factor de Herencia de Implementación)

Se ha definido una métrica para medir el factor de herencia de implementación que tiene una clase con respecto a otra. Dada una clase derivada, esta métrica consiste en la suma de los métodos virtuales y no virtuales implementados en su clase base, entre el número total de métodos de dicha clase base.

$$FHIAC = \frac{\Sigma FHIJ}{NOH} \quad (3)$$

$$FHI = \frac{\Sigma Mv + \Sigma Mnv}{Tm} \quad (1)$$

La expresión matemática de la métrica FHI se muestra a continuación:

En donde:

ΣMv = Sumatoria de métodos virtuales implementados en una clase base.

ΣMnv = Sumatoria de métodos no virtuales implementados en una clase base.

Tm = Total de métodos en una clase base.

Debido a la normalización de la métrica, los valores obtenidos no dependen de la cantidad de métodos que se tengan en una clase dada. El valor óptimo es 0, lo cual significa que no se está heredando la implementación de los métodos. En caso contrario, el valor menos deseado es el 1, lo que significa que todos sus métodos están heredando su implementación.

4.2. Métrica FHIJ (Factor de Herencia de Implementación de una Jerarquía de Clases)

Se ha definido una métrica para medir el factor de herencia de implementación que tiene una jerarquía de clases. Esta métrica consiste en que, para una jerarquía de clases dada, es la suma del FHI de cada subclase entre el número total de clases en esa jerarquía menos 1.

La expresión matemática de la métrica FHIJ se muestra a continuación:

$$FHIJ = \frac{\Sigma FHI}{Tc - 1} \quad (2)$$

En donde:

ΣFHI = Sumatoria de FHI de una clase base.

Tc = Total de clases en esa jerarquía.

De la misma manera, el valor óptimo es 0, lo cual significa que no se está heredando la implementación de los métodos en la jerarquía, y 1 es el valor menos deseado, lo que significa que todos sus métodos están heredando su implementación.

4.3. Métrica FHIAC (Factor de Herencia de Implementación de una Arquitectura de Clases)

Se ha definido una métrica para medir el factor de herencia de implementación que tiene una arquitectura de clases. Esta métrica consiste en la suma de los FHIJ de las ramas

de clases entre el número de jerarquías de clases (NOH). La expresión matemática de la métrica FHIAC se muestra a continuación:

En donde:

$\Sigma FHIJ$ = Sumatoria de FHIJ de una jerarquía de clases.

NOH = Es el conteo de las jerarquías de clase.

De la misma manera, el valor optimo es 0, lo cual significa que no se está heredando la implementación de los métodos en una arquitectura de clases, y 1 es el valor menos deseado, lo que significa que todos sus métodos están heredando su implementación.

4.4. Métrica FFC (Factor de Flexibilidad de Clases)

Se ha definido una métrica para medir el factor de flexibilidad que tiene una clase con respecto a otra. Dada una clase base, esta métrica consiste en la suma del NOP, entre el número total de sus métodos. La expresión matemática de la métrica FFC se muestra a continuación:

En donde:

$$FFC = \frac{\Sigma NOP}{Tm} \quad (4)$$

ΣNOP = Sumatoria de los métodos virtuales que exhiben un comportamiento polimórfico.

Tm = Total de métodos en una clase base.

Debido a la normalización de la métrica, los valores obtenidos no dependen de la cantidad de métodos que se tengan en una clase dada. El valor óptimo es 1, lo cual significa que todos los métodos de la clase son abstractos, por lo tanto, todos exhiben un comportamiento polimórfico. En caso contrario, el valor menos deseado es el 0, el cual significa que ninguno de los métodos es abstracto, por lo tanto, no exhiben un comportamiento polimórfico.

4.5. Métrica FMFAC (Factor Medio de Flexibilidad de Arquitecturas de Clases)

Se ha definido una métrica para medir el factor de flexibilidad que tiene una arquitectura de clases. Esta métrica consiste en la suma de los FFC de la arquitectura del módulo entre el número total de clases. La expresión matemática de la métrica FMFAC se muestra a continuación:

En donde:

ΣFFC = Sumatoria del Factor de Flexibilidad de Clase.

$$FMFAC = \frac{\Sigma FFC}{Tc} \quad (5)$$

Tc = Total de clases en la arquitectura.

De la misma manera el valor óptimo de FMFAC es 1, lo cual significa que todos los métodos de las clases de la arquitectura son virtuales, por lo tanto, todos exhiben un comportamiento polimórfico, y el valor menos deseado de FMFAC es el 0, el cual significa que ningún método de las clases de la arquitectura es virtual, por lo tanto, ninguno tiene un comportamiento polimórfico.

Todas estas métricas cumplieron con los requerimientos para establecerse como escalas ordinales, los cuales son: que sean de orden débil y que además se cumpla la propiedad de homomorfismo (Suze, 1992). Según la teoría de la medición, estos requerimientos sustentan a las métricas.

5. Discusión

Utilizando el lenguaje Java se programó el proceso de cálculo del conjunto de métricas presentado y se integró este código a un marco de métricas de calidad de software. Este marco automatiza varias métricas, para la medición de calidad del software orientado a objetos. Para la etapa de pruebas del marco de métricas de calidad desarrollado, se utilizaron dos casos de prueba, un “marco estadístico” y el sistema “PSPCenidet”. Se realizó el cálculo de las métricas manualmente, así como automáticamente mediante el marco de métricas de calidad de software. Comprobando que ambos valores obtenidos de manera manual y automática fueron los mismos.

El principal aporte de este desarrollo es el conjunto de métricas de calidad de software, para medir el factor de flexibilidad (FFC y FMFAC) y de herencia de implementación (FHI, FHIJ y FHIAC), en diseños arquitecturales de clases. La definición de estas métricas, servirá para medir el grado de cierto código desagradable (*smell code*) por herencia de implementación en código legado.

Agradecimientos

Tecnológico Nacional de México/CENIDET, por el proyecto 6077.19-P “Generación y evaluación de servicios Web de aprendizaje” y al CONACyT por la beca nacional de Orlando Ortiz Gutierrez.

Referencias

- Briand, L., El Emam, K., & Morasca, S. (1996). On the Application of Measurement Theory in Software Engineering. *Empirical Software Engineering*, 1(1), 61–88. doi: <https://doi.org/10.1007/BF00125812>
- Chawla, M. K., & Chhabra, I. (2013). Capturing OO Software Metrics to attain Quality Attributes – A case study. *International Journal of Scientific & Engineering Research*, 4(6), 359–363.
- Chawla, S., & Nath, R. (2013). Evaluating Inheritance and Coupling Metrics. *International Journal of Engineering Trends and Technology*, 4(7), 2903–2908.
- Gary, F. (2010). Measurement Theory for Software Engineers. Recuperado a partir de <https://courses.cs.ut.ee/2010/se/uploads/Main/measurement-theory.pdf>

- IEEE Standard Glossary of Software Engineering Terminology. (1990). *IEEE Std 610.12-1990*, 1–84. doi: <https://doi.org/10.1109/IEEESTD.1990.101064>
- Pinto, A. (2015). Interface inheritance vs Implementation inheritance. Recuperado a partir de <https://www.linkedin.com/pulse/why-prefer-interface-inheritance-implementation-adrain-pinto>
- Suze, H. (1992). Properties of software measures. *Software Quality Journal*, 1(4), 225–260. doi: <https://doi.org/10.1007/BF01885772>
- Suze, H. (1995). Properties of Object-Oriented Software Measures. Recuperado a partir de <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.27.7578&rep=rep1&type=pdf>

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 2019 <http://www.aisti.eu>

