REVECOM

REVISTA VENEZOLANA DE COMPUTACIÓN

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

Sociedad Venezolana de Computación

Vol. 5, No. 2
Diciembre 2018

ISSN: 2244-7040

SVC VENEZUELA
Sociedad Venezolana de Computación

Revista
Venezolana
de Computación

REVECOM

Revista Venezolana de Computación

Sociedad Venezolana de Computación

Editores:
Eric Gamess, Wilmer Pereira, Yudith Cardinale

ISSN: 2244-7040

Vol. 5, No. 2
Diciembre 2018

Editorial

Blockchain: La Paradójica Criptografía para la Transparencia que Inspira Confianza

La transparencia es uno de los sublimes objetivos de una sociedad abierta. Sirve al propósito trascendental de la justicia perfecta porque cuando todo se sabe no debería haber errores de juicio. Por supuesto, esa utopía tiene límites teóricos y prácticos. Entre los primeros está el derecho a la privacidad de cada persona.

Históricamente, por otro lado, la transparencia sí ha tenido límites prácticos. No sabemos todo lo que ocurre en la cosa pública. Sigue siendo prácticamente imposible, en general, registrar o recuperar, con confianza, cada detalle de cada acción en cada servicio prestado por el Estado o, de hecho, en cualquier organización. Ha sido un desafío de complejidad insuperable, hasta ahora.

Las computadoras han entrado en la escena. En 1976, el Profesor Joseph Weizenbaum, uno de los fundadores del Departamento de Ciencias de la Computación del Instituto Tecnológico de Massachusetts, publicó un voluminoso libro: El Poder de las Computadoras y la Razón Humana (*Computer Power and Human Reason, W. H. Freeman & Co, 1st edition, March 1976*) en el que criticaba con mucho cuidado este moderno afán de confiarle a las máquinas toda nuestra información y su procesamiento. Su preocupación, nos contaba, comenzó cuando una pieza de software llamada “Eliza”, que él mismo escribió y que presentó como una suerte de broma tecnológica, saltó a la fama como el representante de una nueva forma de consultoría psicológica. Eliza es lo que ahora llamamos un chatbot. Pero no fue diseñado para y realmente no entiende a la o al interlocutor. Apela a ciertas concordancias sintácticas para identificar tipos de oraciones y seleccionar respuestas que suenan apropiadas, especialmente para alguien que solo quiere “ser oído”. Weizenbaum estaba tan mal impresionado por el éxito de su Eliza que, cuando lo conocimos en el 2000, todavía insistía vehementemente en el peligro de confiar en las máquinas, porque nos pueden engañar. Con experiencias como esa, pocos esperarían que las máquinas se conviertan en el pilar de una solución informática que genera confianza a partir de la nada.

La *Blockchain, BC* o Cadena de Bloques, es una base de datos diseñada para preservar y recuperar un libro de registro contables, como el que lleva todo banco o negocio. Pero, a diferencia de los libros o los balances electrónicos tradicionales, la *BC* no se guarda en un solo computador. Se guarda una copia en cada uno de los nodos que se asocien a una particular plataforma *BC*. Y para mantener actualizada una *blockchain*, así distribuida y replicada por toda la Internet entre máquinas de desconocidos, un conjunto preestablecido de reglas, un protocolo, que ha sido convertido en software (libre, para mayor transparencia), se encarga de todo (en todas y cada una de las máquinas que lo ejecutan). La seguridad de acceso es ofrecida por mecanismos criptográficos implantados en ese software. Pero el aporte más importante de la criptografía en la *BC* no es el ocultamiento de datos o claves de sus usuarios, que está allí con sus conocidas limitaciones. El servicio más importante de la criptografía para la *BC* es la preservación de la integridad de los datos, impidiendo alteraciones fraudulentas, por medio de un uso sistemático y totalmente abierto de las llamadas funciones *hash* (o funciones codificadoras-decodificadoras).

Editorial

Las funciones *hash* le otorgan a la *BC* una propiedad extraordinaria para una pieza de software y datos: la inmutabilidad. Es prácticamente imposible cambiar el contenido de la *BC* sin que se note fácilmente. Una función hash recibe como entrada un texto cualquiera y produce un código hash, o resumen de tamaño fijo, que identifica ese texto. Si alguien alterara el texto de entrada, el nuevo código hash correspondiente sería entonces muy distinto. Es decir que el aspecto clave es que una pequeña variación en el mensaje original produce, con una altísima probabilidad, un código muy diferente.

La inmutabilidad requiere, sin embargo, no solo el concurso de funciones *hash* sensibles a cambios mínimos, sino también ciertas condiciones de operación para la *BC*. Debe haber una estrategia clara para resolver conflictos entre los muchos nodos que replican la *BC* al momento de actualizarla y, además, que una porción adecuada de esos nodos no responda a una misma autoridad administrativa. Si eso no se cumple, sería todavía posible cambiar los valores sin dejar rastro. Nos pueden engañar.

En el prefacio del su libro *Blockchain, Blueprint for a New Economy* (*O'Reilly Media, 1 edition, February 2015*), Melanie Swan dice: "Debemos pensar en la cadena de bloques (blockchain) como algo similar a la Internet – tecnología de la información extendida con niveles organizados por capas y múltiples clases de aplicaciones para todo tipo de registro de bienes, inventario, intercambio, incluyendo cada área de las finanzas, la economía y el dinero; bienes duros (propiedad física, casas, carros); y bienes intangibles (votos, ideas, reputación, intención, datos de salud, información, etc). Pero el concepto de la cadena de bloques es todavía más; es un nuevo paradigma organizativo para el descubrimiento, evaluación y transferencia de todo tipo de "quanta" (unidades discretas) de cualquier cosa, y potencialmente para la coordinación de toda actividad humana a una mucho mayor escala de lo que ha sido posible hasta ahora".

Dr. Jacinto Dávila
Profesor del Centro de Modelado y Simulación de la Facultad de Ingeniería
Universidad de Los Andes

Revista Venezolana de Computación

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

En noviembre de 2018, se celebraron conjuntamente la Sexta Conferencia Nacional de Informática, Computación y Sistemas (CoNCISa 2018) y la Sexta Escuela Venezolana de Informática (EVI 2018), en la Universidad de Los Andes, Mérida, Venezuela.

La edición de este noveno número de ReVeCom está dedicada a los mejores trabajos presentados en CoNCISa 2018. Esta edición consolida un esfuerzo grande que se ha venido haciendo en el seno de la SVC, para promover la investigación en el campo de la computación a nivel nacional, e impulsar una nueva generación académica y profesional en nuestra área de saber para el desarrollo del país.

ReVeCom es una revista abierta para una mayor difusión de los resultados de investigación. Cuenta con una página web (http://www.svc.net.ve/revecom), donde se encuentran los trabajos publicados e información sobre la revista. La revista promueve la pluralidad de intereses, dando cabida a la divulgación de trabajos de todos los campos del conocimiento inherentes a la computación.

Además de selecciones de los mejores artículos de conferencias, ReVeCom también publica artículos de investigación en el campo de la computación, a través de un arbitraje por expertos del área. Por ende, se hace una invitación amplia a la comunidad informática nacional, latinoamericana e internacional, a someter sus propios trabajos para los números de ReVeCom por venir.

Directorio de la Sociedad Venezolana de Computación

Presidente:
Dr. Leonid Tineo (Universidad Simón Bolívar – Venezuela)

Vicepresidente:
Dr. Eric Gamess (Universidad Central de Venezuela – Venezuela)

Secretario:
Dr. Wilmer Pereira (Universidad Católica Andrés Bello – Venezuela)

Tesorero:
Dr. David Coronado (Universidad Simón Bolívar – Venezuela)

Coordinadora de Educación e Investigación:
MSc. Mildred Luces (Universidad Nacional Experimental de la Gran Caracas – Venezuela)

Coordinadora de Publicaciones:
Dra. Yudith Cardinale (Universidad Simón Bolívar – Venezuela)

Coordinadora de Eventos:
MSc. Soraya Carrasquel (Universidad Simón Bolívar – Venezuela)

Edición

Comité Editorial

Miembros del Comité Editorial

Tabla de Contenido

Revista Venezolana de Computación
ISSN: 2244-7040
http://www.svc.net.ve/revecom
Vol. 5, No. 2, pp. 1-11, Diciembre 2018
Selección de los Mejores Artículos de CoNCISa 2018

Ciclo Autonómico de Análisis de Datos para el Diseño de Descriptores para Algoritmos de Aprendizaje Automático

Ricardo Vargas[1], Jose Aguilar[2], Eduard Puerto[3]
ricardo.servitechs@gmail.com, aguilar@ula.ve, eduardpuerto@ufps.edu.co

[1] Posgrado en Computación-CEMISID, Universidad de Los Andes, Mérida, Venezuela
[2] CEMISID, Universidad de Los Andes, Mérida, Venezuela
[3] Grupo de Investigación GIDIS, Universidad Francisco de Paula Santander, Cúcuta, Colombia

Resumen: Varios trabajos en la literatura han determinado que, para obtener buenos resultados con Algoritmos de Aprendizaje Automático, se requieren excelentes descriptores del fenómeno estudiado. En particular, para el proceso de reconocimiento de patrones es importante tener buenos descriptores. En ese sentido, en este trabajo se propone un ciclo autonómico de tareas de Analítica de Datos (AdD) que faciliten la obtención de descriptores para el proceso de reconocimiento de patrones. El ciclo autonómico provee las características /descriptores ideales a ser usado por los Algoritmos de Aprendizaje Automático en sus tareas de construcción de modelos de conocimiento (clasificadores, predictores, etc.). Los experimentos iniciales con el ciclo autonómico han mostrado resultados alentadores.

Palabras Clave: Ciencias de Datos; Ingeniería de Características; Analítica de Datos; Aprendizaje Automático.

Abstract: Several works in the literature have determined that, in order to obtain good results with Automatic Learning Algorithms, excellent descriptors of the phenomenon studied are required. In particular, for the pattern recognition process it is important to have good descriptors. In this sense, this paper proposes an Autonomic cycle of Data Analytics (AdD) tasks that facilitate the obtaining of descriptors for the pattern recognition process. The autonomic cycle provides the ideal characteristics / descriptors to be used by the Algorithms of Automatic Learning in their tasks of construction of knowledge models (classifiers, predictors, etc.). The initial experiments with the autonomic cycle have shown encouraging results.

Keywords: Data Sciences; Characteristics Engineering; Data Analytics; Machine Learning.

I. Introducción

El área de Ingeniería de Características/Descriptores consiste en el proceso de búsqueda de descriptores en un conjunto de datos, y está compuesto por los procesos de extracción, construcción, reducción y/o selección de descriptores/características [1][2][3][4]. La Ingeniería de Características es muy importante para los algoritmos de Aprendizaje Automático, también conocidos como Aprendizaje de Máquinas (o Machine Learning, en inglés), ya que la precisión de los mismos depende de la calidad de los descriptores. Existen un importante número de trabajos que han estudiado estos problemas individualmente, proponiendo diferentes técnicas específicas para cada uno de esos procesos [2][5][6][7], y, además, normalmente enfocándose en específicas áreas de aplicación, como procesamiento de imágenes [2][8], análisis de tráfico de redes [5], o análisis de identidad urbana y puntos de interés en ciudades [6][7].

Por otro lado, la analítica de datos (AoD o Data analytics, en inglés) es un área que comienza a tener cierto grado de madurez, con un importante número de aplicaciones en diferentes ámbitos [7][9]. Ahora bien, la utilización de la AdD se ha entendido como un proceso de diseño aislado (una tarea para un problema específico), y los trabajos actuales no consideran la integración de un conjunto de tareas de analítica de datos para resolver problemas complejos.

Recientemente, se ha propuesto el concepto de Ciclo Automático de Tareas de AdD para el ámbito de aprendizaje [9][10][11][12], para organizar los diferentes tipos de tareas de Análisis de Datos que se integran en ese entorno, para alcanzar diferentes objetivos de aprendizaje (por ejemplo, para optimizar las condiciones ambientales, o mejorar el proceso de enseñanza-aprendizaje). Un ciclo autónomo es un ciclo cerrado de tareas de análisis de datos, que supervisa constantemente el proceso bajo estudio, tal que las tareas de análisis de datos tienen diferentes roles: observar el proceso, analizarlo, y tomar decisiones.

En este trabajo se propone la construcción de un ciclo autonómico de tareas de AdD para el proceso de ingeniería de características, basándonos en la metodología MIDANO [13][14]. MIDANO es una metodología para desarrollar tareas

de AdD, que parte por el análisis exhaustivo del ámbito de estudio, para determinar en qué procesos es posible extraer conocimiento desde los datos.

En particular, por medio del ciclo autonómico se busca simplificar el proceso de ingeniería de características, subdividiéndolo en tareas de AdD más simples, las cuales son fáciles de implementar y evaluar. Cada tarea es definida por un grupo de técnicas de minería, requeridas para el proceso de ingeniería de características, según el problema específico que se esté considerando en un momento determinado.

Así, este trabajo es del ámbito de las ciencias de los datos, y en particular, de la Ingeniería de Características, por lo cual se organiza de la siguiente manera. En la Sección II se ampliará el contexto teórico, y se presentan los trabajos de referencias para nuestra propuesta. La Sección III introduce el ciclo autonómico (CA, por sus siglas en español) propuesto, sus tareas y técnicas que lo componen, y el modelo de datos que lo acompaña. La Sección IV detalla la implementación de este CA y se presenta el caso de estudio para probarlo. En la Sección V se realizan experimentos, y en la siguiente Sección se presentan las conclusiones y trabajos futuros.

II. Contexto Teórico

A continuación, se presenta que se entiende por *ingeniería de características*, y la metodología utilizada en el proceso de AdD.

A. Ingeniería de Caracteristicas

Al realizar procesos de reconocimiento de patrones, ya sea mediante métodos supervisados, semi-supervisados o no supervisados, uno de los elementos más importantes a tener en cuenta son los descriptores o características que se usan para representar el fenómeno a estudiar. Estos descriptores son de suma importancia, ya que la precisión del modelo depende de que los descriptores representen lo mejor posible los objetos a reconocer, además de que también repercuten en los recursos, tiempo, entre otras cosas, requeridos por los modelos.

Es por esto que existen diferentes tipos de técnicas usadas para procesar los datos del experimento, y obtener los descriptores que se usarán en el proceso de reconocimiento. A esta fase se le denomina *Ingeniería de Características*, y consiste en la extracción, construcción, reducción y/o selección, de descriptores/características desde los datos. Cada una usa diferentes técnicas de minería y tiene diferentes objetivos [1][2][3][4][15]. A continuación, se describen los procesos principales de la Ingeniería de Características considerados en este trabajo.

Extracción de Características (Feature Extraction, FE): Estas técnicas buscan identificar los descriptores que mejor describan el fenómeno estudiado desde los datos disponibles, que, a su vez, sean de mayor utilidad para construir los modelos de conocimiento. Para ello, se aplican transformaciones sobre los datos usando funciones específicas. Estas funciones varían dependiendo de distintos criterios, como el tipo de variable/característica sobre la cual se implementará, el tipo de algoritmo de aprendizaje a usar, etc. Ahora bien, los criterios más comunes a usar para escoger las funciones de transformación son: a) maximizar varianza o variación, es decir, buscar características que tomen valores diferentes en cada instancia, ya que características que puedan tomar los mismos valores en diferentes instancias no aportarían ningún valor como discriminantes, b) reducir correlaciones o evitar características redundantes, usualmente lográndolo por medio de funciones que reducen la dimensionalidad entre los datos. Existen diferentes tipos de funciones que se pueden usar, dependiendo del tipo de datos al cual se aplicará. En este tipo de técnicas se puede incluir el Análisis de Componentes Principales (PCA, por sus siglas en inglés), el cual es una de las más comunes [2][3][4][6], y busca reducir la dimensionalidad al proyectar valores en componentes que mejor reflejen la información de los datos originales, con mayor independencia entre ellos.

Los tipos de características/descriptores pueden ser:

a) Características Estadísticas [5][8]*:* Son características que se extraen de datos de tipo numéricos, mediante valores estadísticos como la media, mediana, moda, desviación estándar, etc.

b) Basados en Grafos [5][8]*:* Son características que usan algún tipo de representación en grafos, de tal manera de poder aplicar la teoría de grafos. En general, los grafos pueden representar muchos problemas importantes, como redes sociales, interacciones entre bacterias, redes de computadoras, etc. Sobre estas representaciones se pueden aplicar diferentes técnicas de la teoría de grafos, para determinar cuáles son los mejores descriptores. Por ejemplo, la técnica de detección de comunidades, en la cual se busca agrupar nodos fuertemente conectados en clusters, mientras que los nodos que están en diferentes clusters son los que están más débilmente conectados. Estos tipos de técnicas varían según si se aplican a grafos dirigidos, no dirigidos, con pesos, sin pesos, etc. En general, se pueden usar métricas como la distancia, centralidad, densidad, además de la ya mencionada, entre otras.

c) Espectrales o Basadas en Series de Tiempo: Estos son tipos de características que se derivan de datos secuenciales o continuos, como por ejemplo una serie de eventos que ocurren unos detrás de otros [14]. Se puede tomar en cuenta la periodicidad, magnitud del espectro, variaciones temporales, etc., y se usan técnicas como la transformada de Fourier [15].

En cuanto a las las técnicas usadas en "Extracción de Características", se pueden clasificar en:

a) Métodos Lineales: Usualmente, los problemas de clasificación pueden resolverse mediante separaciones lineales en un plano o hiperplano, es decir, las clases pueden delimitarse fácilmente "trazando" líneas que las separen. Entre estas técnicas se encuentran la ya mencionada PCA, y también el Análisis de Discriminantes Lineales (LDA, por sus siglas en inglés), el cual es un método que también busca reducir dimensionalidad, pero a diferencia de PCA que es no supervisado (no toma en cuenta clases), LDA es supervisado y funciona buscando proyecciones óptimas que maximicen las distancias entre las clases y minimicen distancias de los datos dentro de esas clases.

b) Métodos no Lineales: Cuando la clasificación es más compleja y las clases no son linealmente separables, es decir, los límites entre ellas son discontinuos, se utilizan métodos no

lineales, para realizar esta separación de clases. Entre estos métodos pueden usarse técnicas como realizar una transformación de los datos a dimensiones más altas, donde sí sea posible realizar una separación lineal para aplicar un modelo lineal. Veamos el ejemplo de la Figura 1 de datos no linealmente separados.

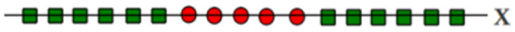

Figura 1: Método Lineal

En la Figura 1 se aprecian dos clases (representadas por los cuadros verdes y círculos rojos), los cuales no están linealmente separados. Se puede realizar una transformación a dos dimensiones mediante la función $X \rightarrow \{X, X2\}$, resultando en un doble valor por cada elemento del ejemplo original, que al representarlo en un plano se puede apreciar que sí es linealmente separable en esa nueva representación (ver Figura 2). Entre las técnicas que realizan esto están las redes neuronales multicapa, k-NN, máquinas de soporte vectoriales no lineales (SVM por sus siglas en inglés), etc.

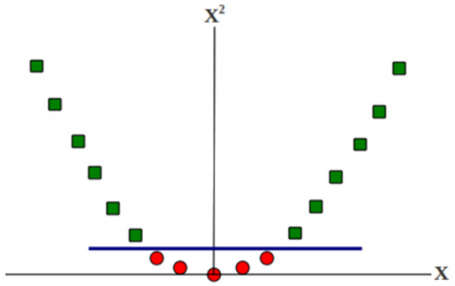

Figura 2: Partición de Datos

Existen algunos trabajos recientes en el área de extracción de características, como [16][17]. En [16] se propone un enfoque para reconocimiento facial bajo condiciones de variación de luz. Este enfoque consta de tres fases, la primera en la cual se realiza una normalización de los componentes de iluminación. En segundo lugar, se realiza la extracción de descriptores. Para esto, los autores usan dos grupos de técnicas: holísticas y locales. Entre las técnicas holísticas se encuentran PCA y LDA, y entre las locales los métodos de patrones binarios locales (LBP, por sus siglas en inglés) o patrones direccionales locales (LDP, por sus siglas en inglés). Finalmente, en la tercera fase usan SVM para realizar la clasificación. En [17] se propone un proceso de extracción de características en textos, el cual se divide a nivel sintáctico y a nivel semántico. A nivel sintáctico proponen una versión de $\chi 2$ (Chi-squared statistics) mejorada (ICHI), aplicada sobre una matriz de palabras, y a nivel semántico una técnica llamada asignación latente de Dirichlet, sobre una representación de documentos por tópicos. Finalmente, se realiza una fusión entre ambos resultados, para obtener el conjunto de características final.

Construcción de Características (Feature Construction, FC): Estos tipos de técnicas buscan generar nuevas características [1][4]. Su uso principal es cuando se tienen datos incompletos, y se desea buscar la información faltante que complemente la descripción del objeto, o la relación que existe entre las características que ya se tienen. Esta nueva información se puede obtener por medio de inferencia o creación de nuevos datos, pero siempre usando los datos ya existentes para derivarla [4], buscando incrementar el poder de expresión de los datos ya existentes. Algunas de las técnicas que se podrían usar para generar estos nuevos datos son las que aplican algún operador sobre estos datos, como por ejemplo operadores algebraicos en datos numéricos.

En cuanto a propuestas de construcción de características, el trabajo [18] propone un método basado en programación genética (GP por sus siglas en inglés), para obtener características faltantes en datos incompletos. En esta técnica se utilizan funciones de intervalos como parte del conjunto de funciones del GP, resultando en una sustitución del valor faltante por un intervalo de posibles valores, o si es un valor no faltante, se sustituye por un intervalo que incluye el valor original. En [19] se propone un proceso de construcción de características basadas en entropía, para la detección de ataques en redes de comunicación. Este proceso consta de tres partes: primero, se extraen características de cada cabecera de los paquetes de red. Datos como direcciones IP origen y destino, puertos origen y destino, y protocolos, son los más comúnmente usados. En segundo lugar, se calcula la entropía de Shannon, usando un intervalo de tiempo específico, y, por último, se construyen nuevas características con variaciones de entropía, usando combinaciones de características generadas en los pasos previos.

Selección de Características (Feature Selection, FS): Con estos tipos de técnicas, al contrario de las descritas antes, se busca reducir el conjunto de datos a utilizar. Como lo definen Blum y Langley en [1], "se diferencia de las transformaciones de características en que no se generan nuevas, si no que se selecciona un subconjunto de ellas". Esta reducción puede ser realizada ya sea mediante una selección entre las características disponibles, o mediante la agregación/fusión de ellas. La selección se puede realizar [4] generando subconjuntos aleatoriamente, incrementalmente, etc. En el caso incremental, podría iniciarse con un subconjunto vacío, y se van agregando características una a una; o empezar con el conjunto completo, e ir eliminando características. Se suelen usar diferentes criterios para evaluar si un subconjunto generado es óptimo o no, estos criterios se pueden agrupar dentro de dos grandes modelos: Filtros (en adelante Filtering) y Envoltorios (en adelante Wrapping) [15].

a) Los Filtering constan de criterios que son independientes del modelo de aprendizaje que se usará luego. Es una aproximación eficiente, pero debido a esta separación del proceso de aprendizaje posterior, puede descartar información que hubiera sido útil para el mejor rendimiento del algoritmo de aprendizaje usado. El proceso usado suele realizarse en dos pasos; primero la aplicación del criterio de selección, entre las cuales pueden estar: dependencia de una característica de su clase, correlaciones de característica-característica, característica-clase, medidas de distancia, medidas de consistencia, etc. En el segundo paso, simplemente se selecciona el subconjunto con mejor ranking.

b) En el modelo de Wrapping sí se toma en cuenta el algoritmo de aprendizaje a usar, es decir, se mide el rendimiento

obtenido por ese algoritmo, usando el subconjunto de características seleccionadas. Ese proceso se suele realizar usando una estrategia de búsqueda entre las características o subconjunto de características, las cuales se pasan al algoritmo para medir su rendimiento, y este resultado se vuelve a pasar al componente de búsqueda en un proceso iterativo, hasta que se selecciona el conjunto de características con el mejor rendimiento. Para evitar una búsqueda exhaustiva se pueden usar distintas estrategias como algoritmos genéticos, Best-first, Hill-climbing, etc.

c) Existe un tercer modelo que es una especie de unión entre Filtering y Wrapping, para obtener lo mejor de ambos mundos [20][21]. Esto porque Filtering no toma en cuenta el algoritmo de aprendizaje a usar, pudiendo descartar características útiles para ese algoritmo, y Wrapping debe evaluar el rendimiento de los subconjuntos de característica en el modelo, haciéndolo bastante costoso computacionalmente. Este tercer modelo, llamado Embedding, busca incrustar el proceso de selección en la construcción del modelo de aprendizaje, logrando mejorar el costo computacional, ya que no es necesario correr el proceso de aprendizaje repetidas veces.

Finalmente, en el área de selección de características se pueden mencionar algunos trabajos, tales como [22], que propone un enfoque de selección de características para previsión de precios y cargas en sistemas de suministro eléctrico. Este enfoque mezcla características de modelos de Filtering y Wrapping. De este modo, el método propuesto selecciona un subconjunto de características tomando en cuenta criterios como relevancia, redundancia e interacción entre características, considerando el algoritmo de aprendizaje que se usará. En [23] se proponen mejorar la identificación de patologías del pecho mediante el análisis de radiografías. En este enfoque se extraen características de estas imágenes radiológicas usando redes neuronales convolucionales, luego, establecen una fase de reducción de características realizando una combinación de ellas, por medio de una técnica de fusión lineal ponderada basada en los pesos de probabilidad de cada clase. Finalmente, realizan una fase de selección de características, en la cual usando pruebas de Kruskal-Wallis sobre la varianza de los datos, determinan las características más informativas.

B. MIDANO

MIDANO es una metodología para el Desarrollo de Aplicaciones de Minería de datos (MD) basada en el Análisis Organizacional. MIDANO es diseñada para el desarrollo de aplicaciones de Minería de Datos para un proceso de cualquier empresa o institución, sin embargo, esta puede ser utilizada en procesos de AdD [13][14]. MIDANO está compuesta por tres fases (ver Figura 3).

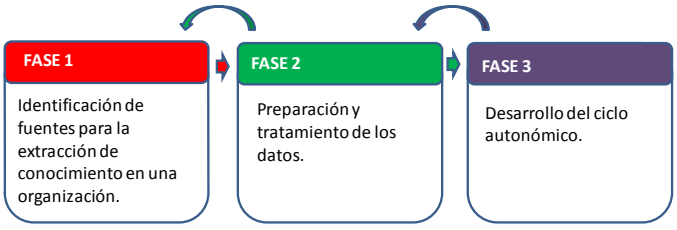

Figura 3: MIDANO

Fase 1. Identificación de Fuentes para la Extracción de Conocimiento en una Organización: Esta fase tiene como finalidad realizar un proceso de ingeniería de conocimiento, orientado a organizaciones/empresas, de las cuales no se conoce o se tiene poca información del (de los) problema(s) o los procesos a estudiar. El principal objetivo de esta fase es conocer la organización, sus procesos, sus expertos, entre otros aspectos, para definir el objetivo de la aplicación de AdD en la organización.

Fase 2. Preparación y Tratamiento de los Datos: Para aplicar AdD sobre un problema en específico, es necesario contar con un historial de datos asociado al problema de estudio. Esto conlleva realizar distintas operaciones con los datos, con la finalidad de prepararlos. Ese proceso se basa en el paradigma ETL (por sus siglas en inglés): extracción de los datos desde sus fuentes, transformación de los datos, y carga de los mismos en el almacén de datos del CA. Para realizar este proceso se crea una vista minable, que básicamente contiene información sobre las variables y sus históricos. En específico, se crea una vista minable conceptual (VMC), que detalla cada una de las variables a ser tomadas en cuenta para las tareas de AdD. La misma está compuesta por la descripción de todas las variables de interés, y algunos campos adicionales de importancia para realizar el proceso de tratamiento de datos (por ejemplo: dependencias con otras variables, transformaciones a realizar, entre otras características). Con esta VMC se crea el modelo de datos a ser usado por el medio de almacenamiento (almacén de datos). Finalmente, el medio de almacenamiento es cargado con los datos. Al medio de almacenamiento cargado con los datos, lo llamaremos vista minable operativa. Así, en esta fase se construye el modelo de datos requerido por el ciclo autonómico de tareas de AdD. Por lo tanto, en esta fase se realiza la preparación y tratamiento adecuado de los datos, que serán utilizados por el ciclo autonómico de AdD.

Fase 3. Desarrollo del Ciclo Autonómico de Tareas de AdD: En esta fase se implementan las tareas de AdD. Así, esta fase tiene como objetivo implementar las diferentes tareas de AdD del ciclo autonómico, que generan los modelos de conocimientos requeridos (por ejemplo, modelos predictivos, modelos descriptivos, etc.). Esta etapa culmina con la implementación de un prototipo del ciclo autonómico. Para el desarrollo de las tareas de AdD, se puede usar cualquiera de las metodologías existentes de desarrollo de tareas de MD. Además, durante esta fase se realizan experimentos para validar los modelos de conocimiento generados por las tareas de AdD. La utilización de esta metodología permite el desarrollo sistemático de aplicaciones de software especializado basada en técnicas inteligentes, para la extracción de conocimiento a partir de los datos almacenados en las bases de datos de cualquier industria o proceso

III. Especificación de las Tareas de Análisis de Datos

En esta sección se describe el ciclo autonómico a ser utilizado, con sus respectivas tareas de AdD. El objetivo de este ciclo autonómico es buscar la obtención de las características óptimas

que mejor describan el objeto estudiado, para su uso en un proceso de reconocimiento de patrones.

A. Ciclo Autonómico

El ciclo autonómico propuesto para este trabajo consta de dos etapas:

- Monitoreo: En esta etapa del ciclo autonómico se realiza la recolección de los datos desde sus fuentes.
- Análisis y toma de decisiones: Se ejecutan las tareas de Análisis de Datos sobre los datos obtenidos en la etapa anterior, para determinar los descriptores que se usarán en la clasificación.

Las tareas de AdD que se aplicarán en cada etapa se describen en la Tabla I.

Tabla I: Especificación de Tareas de AdD para el Ciclo Autonómico

Tarea	Nombre	Fuentes generales de datos requeridas	Indicadores generados	Efectos esperados sobre el objetivo estratégico
Monitoreo	Captura de datos	Tablas VMC, ETL (Extracción, Tratamiento y Carga) y CCA (colección, curetaje y agregación)	Datos del Experimento	Se obtienen los datos recogidos en etapas anteriores sobre los cuales se quiere realizar la clasificación
Análisis y Toma de decisiones	Construcción de características	Datos obtenidos del paso anterior	Datos tratados y depurados	Se emplean los primeros métodos y técnicas de preparación y tratamiento de datos
	Extracción de características	Datos depurados	Medias, medianas, modas, mínimos, máximos, entre otros valores numéricos necesarios	Conjunto de técnicas para extraer valores numéricos, métricas, etc. que mejor representen los datos
	Selección y reducción de características	Representación numérica de los datos	Conjunto final de características	Se terminan de depurar las características extraídas, reduciendo descriptores redundantes, o descartando algunas carac excedentes

En las siguientes tablas (Tablas II al V), se especifican en detalle cada una de estas tareas:

Tabla II: Detalles de la Tarea de Captura de Datos

Nombre de la tarea:	Captura de datos
Descripción	Obtener los datos del caso de estudio sobre el que se desea realizar la clasificación.
Fuente de datos	Tablas de VMC, ETL, CCA
Tipo de tarea de analítica de datos	Descubrimiento
Tipo de modelo de conocimiento	Descriptivo
Tareas relacionadas de analítica de datos	Construcción de características
Tipo de tarea del ciclo (rol)	Monitoreo

Tabla III: Detalles de la Tarea de FC

Nombre de la tarea:	Construcción de características
Descripción	Preparar los datos obtenidos
Fuente de datos	Datos obtenidos en la etapa anterior
Tipo de tarea de analítica de datos	Clasificación, Agrupamiento, Asociación
Técnicas de analítica de datos	Normalización, estandarización, transformación, reducción, discretización, etc.
Tipo de modelo de conocimiento	Descriptivo, Predictivo
Tareas relacionadas de analítica de datos	Captura de datos Extracción de características
Tipo de tarea del ciclo (rol)	Análisis / Toma de decisiones

Tabla IV: Detalles de las Tareas de FE

Nombre de la tarea:	Extracción de características
Descripción	Obtener representaciones numéricas de los datos procesados para usarlas luego en el proceso de clasificación
Fuente de datos	Datos depurados
Tipo de tarea de analítica de datos	Clasificación, Agrupamiento, Asociación, Regresión
Técnicas de analítica de datos	Cálculos estadísticos, KNN, random forest, regresión, grafos, etc.
Tipo de modelo de conocimiento	Descriptivo, Predictivo
Tareas relacionadas de analítica de datos	Construcción de características Selección y reducción
Tipo de tarea del ciclo (rol)	Análisis / Toma de decisiones

B. Modelo de Datos

a) Vista Minable Conceptual para el CA: Los datos se obtienen desde las Tablas generadas por el proceso que se está examinando, principalmente de su propia VMC y Tablas ETL y CCA. Además de estos datos, también se requieren otras variables derivadas de cada tarea del ciclo autonómico. Una posible VMC podría ser la Tabla VI.

Tabla V: Detalles de las Tareas de FS

Nombre de la tarea:	**Selección y reducción de características**
Descripción	Asegurar que se obtienen solo las características más relevantes para realizar la clasificación
Fuente de datos	Características extraídas en la etapa anterior
Tipo de tarea de analítica de datos	Clasificación, Agrupamiento, Asociación
Técnicas de analítica de datos	Filtering, Wrapping, análisis bivariantes, etc.
Tipo de modelo de conocimiento	Descriptivo, Predictivo
Tareas relacionadas de analítica de datos	Extracción de características
Tipo de tarea del ciclo (rol)	Análisis / Toma de decisiones

Tabla VI: Posible Tabla VMC para el Ciclo Autonómico

Variable	**Descripción**	**Procedencia**	**Descripción**
variable_vmc	Variables de la VMC del proceso estudiado	Vmc	Estos son los datos pertenecientes a la tabla VMC sobre la cual se realiza el análisis la cual posee los datos de entrada del caso de uso.
descripcion_vmc	Descripción de las variables de la VMC	Vmc	
procedencia_vmc	Procedencia de las variables de la VMC	Vmc	
observaciones_vmc	Observaciones de las variables de la VMC	Vmc	
variable_etl	Variables de la tabla ETL del proceso estudiado	Etl	Estos son los datos de la tabla etl la cual posee los datos con un primer pre-procesamiento sobre los datos de entrada.
extraccion_etl	Fuente de datos de donde fueron extraídas	Etl	
transformacion_etl	Procesos de pre-procesamiento realizados en ellas	Etl	
carga_etl	Dimensión del modelo donde irán	etl	
variable_cca	Variables de la tabla CCA del proceso estudiado	cca	Datos curados sobre fuentes de datos externas.
coleccion_cca	Fuente externas de datos de donde fueron extraídas	cca	
curacion_cca	Procesos de pre-procesamiento realizados en ellas	Cca	
analisis_cca	Criterios de calidad y dimensión donde irán	cca	
medias	Cálculos de la media sobre datos de entrada	Datos de entrada	Datos estadísticos calculados a partir de los datos curados de entrada necesarios para aplicar tareas de AdD para extracción de características.
medianas	Cálculos de la mediana sobre datos de entrada	Datos de entrada	
modas	Cálculos de la moda sobre datos de entrada	Datos de entrada	
desviacion_estandar	Cálculos de la moda sobre datos de entrada	Datos de entrada	

b) Modelo de Datos Multidimensional: De la VMC propuesta, se obtiene el siguiente modelo multidimensional con las variables agrupadas por temas (ver Figura 4).

Cada dimensión de la tabla multidimensional corresponde a los datos de las diferentes tablas usadas para el análisis. Una dimensión tiene la información de la tabla ETL, otra de las tablas VCM, CCA, y finalmente, una contiene los datos estadísticos generados a partir de los datos del caso de estudio. El contenido de las Tablas ETL y CCA contienen los datos de los procesos de preparación de datos, y la de VCM los detalles de las variables que conforman la fuente de datos (ver [14][15] para más detalles).

IV. Implementación del CA

En esta sección, vamos a dar un ejemplo de instanciación de nuestro CA.

A. Flujo Asociado al CA

Para la implementación del CA, en primer lugar, se define el conjunto de etapas del flujo asociado al CA, a ser usadas al instanciar el modelo propuesto con datos reales del caso de estudio seleccionado. Estas etapas se muestran en la Tabla VII.

Tabla VII: Etapas para el Ciclo Autonómico

Etapa	**Detalle**	**Tipo de Tarea**	**Producto**
1	Captura de datos	Descubrimiento	Datos del Experimento
2	Aplicar técnica de construcción de características	Cálculos, inferencias, procedimientos que producen nuevos datos	Conjunto de características extendido
3	Aplicar transformaciones básicas en los datos	Transformación	Primer conjunto de Características con valores numéricos usables en técnicas de FE
4	Aplicar técnicas de extracción de características (FE)	Transformación, Clasificación, Clustering, etc.	Conjunto de características procesadas
5	Aplicar técnicas de selección	Filtrado, Agrupamiento, Unión, etc.	Conjunto final de características

Estas etapas están basadas en las tareas propuestas para el CA, por lo que en la Tabla se puede ver que cada paso corresponde a una de estas tareas. Además, se especifica el(los) tipo(s) de tarea(s) de análisis de datos que se podrían aplicar, y los datos que se producen en cada paso, siendo el resultado final del flujo, el conjunto de características óptimas seleccionadas.

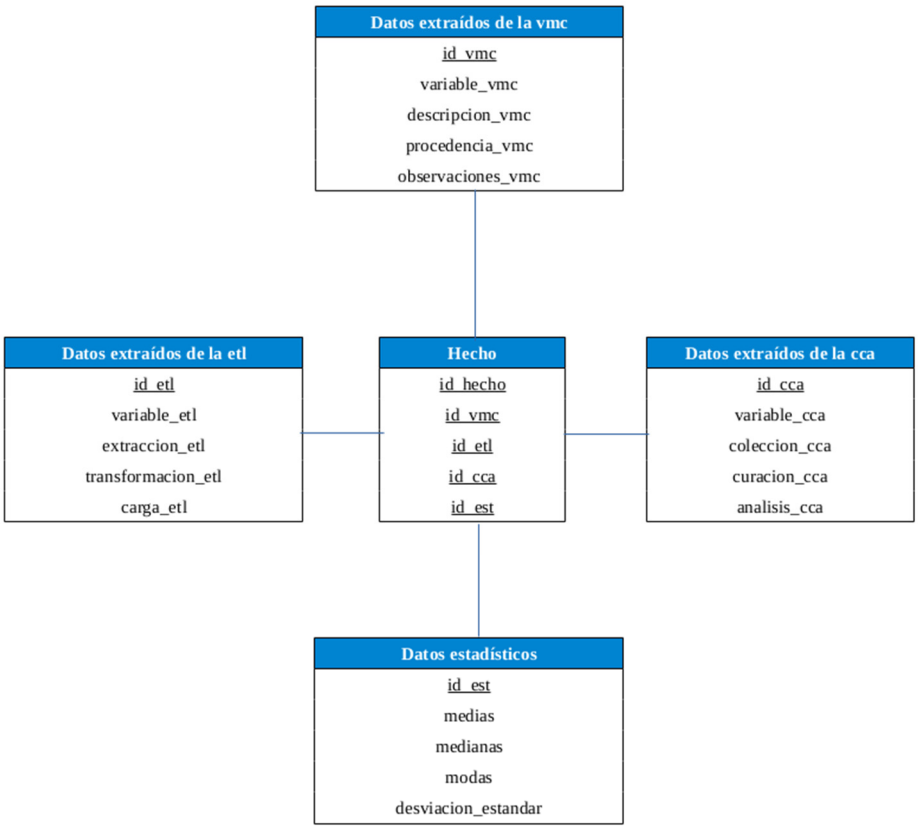

Figura 4: Modelo Multidimencional

B. *Caso de Estudio*

El caso de estudio seleccionado para esta implementación consiste en una base de datos [24] de gestos de letras escritos a mano en una tableta digital, por niños que se encuentran aprendiendo a escribir. Esta base de datos consiste en 6118 registros con diferentes características, obtenidas al momento en el que el niño traza la letra. Entre los datos obtenidos se encuentran [25]:

- Repeats: número de veces que el niño ha intentado escribir la letra
- Class: la letra escrita
- ID: id del niño
- Gender: genero del niño (masculino/femenino)
- Age: edad del niño
- Laterality: lateralidad del niño (zurdo/derecho)
- Duration: el tiempo en ms que el niño tardó en escribir la letra
- letraNbStrokes: número de trazos usados para escribir la letra
- AveragePressure: presión promedio aplicada por el niño al realizar el gesto
- VariancePressure: varianza de la presión aplicada por el niño al realizar el gesto

Se usará esta base de datos para instanciar el CA, aplicando las técnicas correspondientes a cada tarea, para determinar las características que mejor representen las letras dibujadas.

V. Experimentos

Para realizar los experimentos se usará la librería del lenguaje de programación Python, *scikit-learn*, que cuenta con un gran número de implementaciones de algoritmos para ciencia de los datos, lo cual facilita la experimentación, permitiendo concentrarse solo en los posibles resultados del CA.

A. *Especificación de los Pasos*

a) Captura de Datos: En este paso, simplemente se cargan los datos iniciales necesarios para comenzar el proceso de ingeniería de características. En este caso, son los datos obtenidos de INTUIDOC, como se muestra en la Figura 5 [24][25]

b) Aplicar Técnicas de Construcción de Características: Una vez obtenidos estos datos, se aplican las primeras técnicas de construcción de características. Estos nuevos datos con los que se expandirá el espectro de características disponibles, son datos estadísticos generados desde los datos del experimento.

	A	B	C	D	E	F	G	H	I	J	K
1	Repeats	Class	ID	Gender	Age	Laterality	Duration	NbStrokes	AveragePressure	VariancePressure	SavedScoreGlobal
2	1	d	anonstudent1475048982665	BOY	ND	RIGHT	1300	1	0,580593467	0,127687503	0,619901053
3	1	n	anonstudent1475048868521	BOY	ND	RIGHT	1514	1	0,932148755	0,165912816	1
4	2	n	anonstudent1475048868521	BOY	ND	RIGHT	1299	1	0,897993743	0,182596353	1
5	1	u	anonstudent1475048868521	BOY	ND	RIGHT	1567	1	0,955063939	0,125530824	1
6	1	d	anonstudent1475049326525	GIRL	ND	RIGHT	773	1	0,620547235	0,09371386	1
7	1	u	anonstudent1475049326525	GIRL	ND	RIGHT	840	1	0,568899095	0,083730014	0,83106493
8	1	n	anonstudent1475049326525	GIRL	ND	RIGHT	2690	1	0,7090469	0,112996955	1
9	1	u	anonstudent1475048868521	BOY	ND	RIGHT	2609	1	0,933333337	0,109634476	1
10	1	j	anonstudent1478764526427	BOY	6	RIGHT	4130	2	0,898688734	0,116078498	1
11	2	j	anonstudent1478764526427	BOY	6	RIGHT	3293	2	0,886679053	0,104820331	1
12	3	j	anonstudent1478764526427	BOY	6	RIGHT	3285	2	0,881134212	0,132600282	1
13	1	m	anonstudent1478763974290	BOY	ND	RIGHT	1157	1	0,666666687	0,106140371	1
14	2	m	anonstudent1478763974290	BOY	ND	RIGHT	1488	1	0,772387564	0,095200617	0,74480688
15	1	v	anonstudent1478764526427	BOY	6	RIGHT	2238	1	0,542212009	0,063159453	0,840620845
16	2	v	anonstudent1478764526427	BOY	6	RIGHT	2826	1	0,501985133	0,061357192	0,921001711
17	3	v	anonstudent1478764526427	BOY	6	RIGHT	2276	1	0,535986364	0,061060099	1

Figura 5: Datos del Experimento Obtenidos de INTUIDOC

Para este experimento se calcula la media del tiempo de duración que tomó el niño al dibujar la letra, y el número de veces promedio que los niños escribieron cada letra, además de sus desviaciones estándares, agregando así nuevas variables al conjunto total de ellas.

c) Aplicar Transformaciones Básicas en los Datos: Como primer paso de las tareas de extracción de características, se realizan algunas transformaciones básicas necesarias para obtener datos más relevantes. En este experimento se puede notar que, de los datos disponibles, los datos de Class, Gender y Laterality no son óptimos, ya que se encuentran en formato de cadenas de texto, por lo que se realiza una transformación sobre ellos, para obtener un formato numérico, más útil para usar otras técnicas. Los resultados de esta transformación se muestran en la Figura 6, en donde ya todos los valores son numéricos.

	Repeats	Class	Gender	Laterality	Duration	NbStrokes	\
0	1	4	0	1	1300	1	
1	1	14	0	1	1514	1	
2	2	14	0	1	1299	1	
3	1	21	0	1	1567	1	
4	1	4	1	1	773	1	
5	1	21	1	1	840	1	
6	1	14	1	1	2690	1	
7	1	21	0	1	2609	1	
8	1	10	0	1	4130	2	
9	2	10	0	1	3293	2	
10	3	10	0	1	3285	2	
11	1	13	0	1	1157	1	

Figura 6: Transformación de los Datos a Valores Numéricos

d) Aplicar Técnicas de Extracción de Características: En este caso, se aplican dos técnicas de extracción de características. La primera es Random Forest (RF), con la cual se determinan los pesos de las características disponibles, y así la relevancia de cada una de ellas. Esta técnica consiste en generar un conjunto de árboles de decisión, cada uno usando aleatoriamente un subconjunto de las características disponibles como nodos, y se van asignando pesos a cada característica de acuerdo a que tan relevante fue en el resultado del árbol al cual pertenecía [5]. La segunda técnica a aplicar es PCA. Así, se definieron tres escenarios, uno donde solo se usa RF, en otro solo PCA, y en el tercero se aplica una mezcla de ambos, aplicando RF a los datos obtenidos al aplicar PCA.

e) Aplicar Técnicas de Selección de Descriptores: Finalmente, se aplican técnicas de selección y reducción sobre este conjunto de datos final, para descartar las características menos relevantes. Para este experimento se usan diferentes técnicas proporcionadas por scikit learn. La primera es Recursive Feature Elimination (RFE), la cual realiza la selección considerando recursivamente conjuntos de características cada vez más pequeños, seleccionadas basadas en sus importancias calculadas en el paso anterior, hasta conseguir el número de características deseadas (ver Figura 7). En segundo lugar, se usa la técnica VaraianceThreshold, la cual elimina las características cuya varianza no alcancen un valor de umbral propuesto, para lo cual se prueba con distintos valores de umbrales calculados con diferentes porcentajes de varianza aceptable para las características.

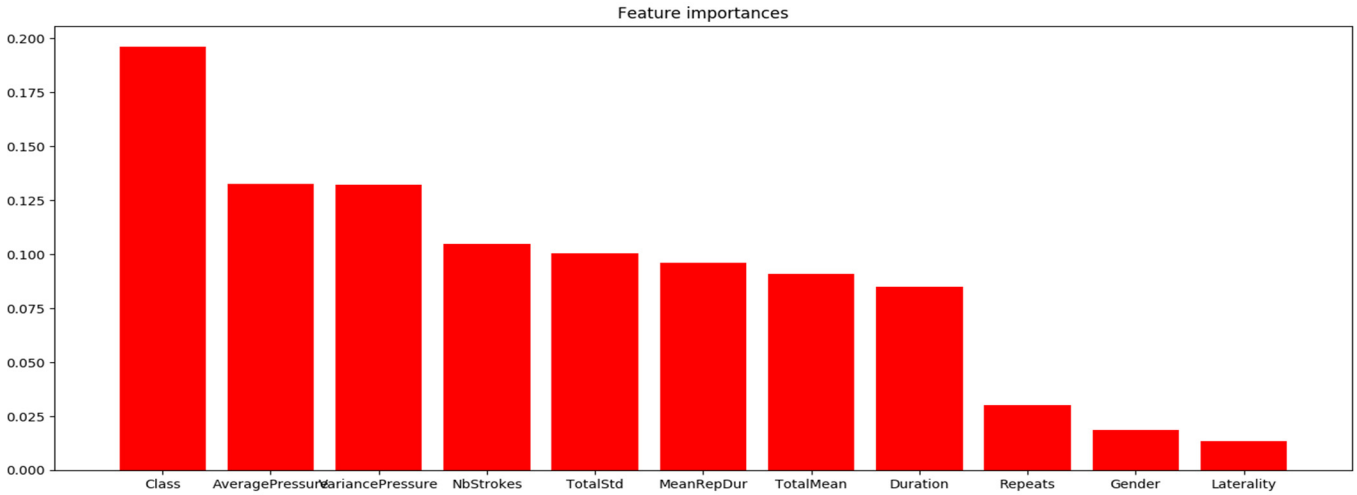

Figura 7: Importancia de Descriptores con Random Forest

B. Resultados

En esta sección se mostrarán los resultados obtenidos al aplicar las técnicas descritas previamente, y se usarán las características finales obtenidas en un algoritmo de clasificación, para estudiar y comparar sus comportamientos.

La Figura 7, muestra el resultado de aplicar RF para extracción de características, en la cual se muestran las características ordenadas por su importancia. Luego de calcular estas importancias, se utilizan para realizar la selección a RFE. Este proceso se realiza un par de veces, para calcular con esta técnica las características óptimas, obteniéndose dos listas: la primera con 8 características: "Class, Duration, NbStrokes, AveragePressure, VariancePressure, MeanRepDur, TotalMean, TotalStd"; y la segunda con 6 características "Class, NbStrokes, AveragePressure, VariancePressure, TotalMean, TotalStd".

Por otro lado, se aplica la técnica de VarianceThreshold, seleccionando las características con varianzas de 50%, 80% y 90%.

Para validar los resultados obtenidos, se usa nuevamente la técnica de RF, en este caso como clasificador, para verificar la precisión de la clasificación de las letras usando los diferentes grupos de características obtenidas por las diferentes técnicas y estrategias definidas en el CA. Para calcular la métrica de precisión se usó la técnica de validación "k-fold cross", con k = 10, tal que el 90% de los datos son usados para entrenamiento y 10% para probar. Estos resultados se muestran en la Tabla VIII, donde los nombres para indicar las técnicas de selección de descriptores son:

- DNP: Datos no procesados, no se aplicó ninguna selección, se usaron todas las características obtenidas hasta el paso de FE.
- RFE_6: Aplicar RFE obteniendo 6 características óptimas.
- RFE_8: Aplicar RFE obteniendo 8 características óptimas.
- RFE_6_V_50: Aplicar RFE obteniendo 6 características óptimas, y luego VarianceThreshold con umbral de 50% de varianza.
- RFE_6_V_80: Aplicar RFE obteniendo 6 características óptimas y luego VarianceThreshold con un umbral de 80% de varianza.
- RFE_6_V_90: Aplicar RFE obteniendo 6 características óptimas y luego VarianceThreshold con un umbral de 90% de varianza.
- RFE_8_V_50: Aplicar RFE obteniendo 8 características óptimas y luego VarianceThreshold con un umbral de 50% de varianza.
- RFE_8_V_80: Aplicar RFE obteniendo 8 características óptimas y luego VarianceThreshold con un umbral de 80% de varianza.

Tabla VIII: Comparación de Resultados

Técnica FS vs FE	RF	PCA	RF + PCA
DNP	0.8155	0.8109	0.8008
RFE_6	0.8097	0.6471	0.8187
RFE_8	0.8090	0.8017	0.8090
RFE_6_V_50	0.8173	0.6023	0.8109
RFE_6_V_80	0.8133	0.6068	0.8068
RFE_6_V_90	0.8203	0.6221	0.8078
RFE_8_V_50	0.8155	0.7848	0.8080
RFE_8_V_80	0.8157	0.7887	0.8042
RFE_8_V_90	0.8090	0.7936	0.8094

La Tabla VIII muestra los valores obtenidos después de aplicar las diferentes técnicas de extracción de características y selección de descriptores usando las métricas de validación descritas anteriormente. Los valores más altos indican un mejor rendimiento en la clasificación. Como se puede observar en la Tabla VIII, los casos donde se usa la técnica de RF en la fase de extracción de características, es donde se obtienen los mejores resultados. Además, los mejores resultados se obtuvieron cuando se seleccionaron 6 características por medio de RFE.

VI. Conclusiones y Trabajos Futuros

En general, este trabajo demuestra la importancia de conseguir buenos descriptores, con la finalidad de mejorar el comportamiento de los algoritmos de Aprendizaje Automático. En particular, la calidad de las métricas se ve influenciado por los descriptores usados (en este caso, se usó la métrica de *precisión* en tareas de clasificación).

Por otro lado, también el trabajo muestra como las diferentes etapas de la Ingeniería de Características están vinculadas. Según las técnicas que se usen en las fases de extracción de características y selección de descriptores, los resultados varían. Lo anterior también indica la necesidad de ver a la Ingeniería de Características desde un proceso de CA, donde las tareas de AdD se engranen entre ellas. En ese sentido, nuestro CA es pertinente para el descubrimiento de óptimos descriptores para procesos de reconocimiento de patrones.

Así, el CA mostró cómo al aplicar distintas técnicas de FC, FE y FS, y combinaciones de ellas, se puede obtener una gran variedad de resultados con calidades distintas, lo cual significa que debe seleccionarse correctamente la técnica que mejor ayude a determinar los descriptores relevantes de acuerdo al área de aplicación que se esté analizando en cada caso.

Este trabajo realiza la prueba de concepto sobre el CA, pero deriva en un importante número de trabajos futuros. Uno de los trabajos futuros es realizar muchas más pruebas con el CA (para diferentes contextos de aplicación), de tal manera de definir el perfil de técnicas adecuadas para el CA según el contexto de aplicación. Eso implica utilizar un mayor número de técnicas en cada una de las fases del CA (FC, FE y FS), no solamente RF, PCA, y RFE, ya que se demostró que el CA es muy sensible a las técnicas usadas en cada tarea de AdD. Otro trabajo futuro es usar más métricas de calidad, además de la precisión (*precision*), tales como la exactitud (*accuracy*) y la memorización (*recall*), en el caso de tareas de clasificación, para determinar si el comportamiento de calidad se mantiene con las diferentes métricas. También, otro trabajo es hacer el estudio del comportamiento del CA para problemas con datos con diferentes características: desbalance entre clases, datos etiquetados y no etiquetados, datos con ruidos, o muchas variables con comportamiento que se solapan entre ellas. Finalmente, en este trabajo solo se usó para el problema de clasificación a RF, pero hay que hacer pruebas con otras técnicas de clasificación, para determinar si se mantiene el comportamiento de las métricas de calidad en los descriptores seleccionados.

Referencias

[1] H. Liu and H. Motoda, *Feature Extraction, Construction and Selection: A Data Mining Perspective*. Springer Science & Business Media, vol. 453, 1998.

[2] S. Khalid, T. Khalil, and S. Nasreen, *A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning*, in proceedings of the Science and Information Conference (SAI), pp. 372-378, London, England, August 2014.

[3] B. Yoshua, O. Delalleau, N. L. Roux, J. Paiement, P. Vincent, and M. Ouimet. *Spectral Dimensionality Reduction*. In Feature Extraction: Foundations and Applications (I. Guyon, S. Gunn, M. Nikravesh, L. Zadeh, Eds.), Springer, pp. 519-549, 2003.

[4] H. Motoda and H. Liu *Feature Selection, Extraction and Construction*. Communication of IICM (Institute of Information and Computing Machinery), vol 5, pp. 67-72, 2002.

[5] F. Pacheco, E. Exposito, M. Gineste, C. Budoin, and J. Aguilar, *Towards the Deployment of Machine Learning Solutions in Traffic Network Classification: A Systematic Survey*, IEEE Communications Surveys and Tutorials, 2018.

[6] M. Chang, P. Buš, and G. Schmitt, *Feature Extraction and K-means Clustering Approach to Explore Important Features of Urban Identity*. in proceedings of the 16th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1139-1144. Cancun, Mexico, December 2017.

[7] M. Sánchez, J. Aguilar, J. Cordero, and P. Valdiviezo, *Basic Features of a Reflective Middleware for Intelligent Learning Environment in the Cloud (IECL)*, in proceeding of the Asia-Pacific Conference on Computer Aided System Engineering (APCASE). Quito, Ecuador, June 2015.

[8] G. Kumar and P. Bhatia, *A Detailed Review of Feature Extraction in Image Processing Systems*, In proceedings of the Fourth International Conference on Advanced Computing & Communication Technologies (ACCT), pp. 5-12. Rohtak, India, February 2014.

[9] J. Aguilar, O. Buendia, K. Moreno, and D. Mosquera. *Autonomous Cycle of Data Analysis Tasks for Learning Processes*, In Technologies and Innovation (R. Valencia-García, et al., Eds.), Communications Computer and Information Science Series, vol. 658, Springer, pp. 187-202, 2016.

[10] J. Aguilar, J. Cordero L, Barba, M. Sanchez, P. Valdiviezo, and L. Chamba, *Learning Analytics Tasks as Services in Smart Classroom*, Universal Access in the Information Society Journal, vol. 17, no. 4, pp. 693–709, 2018.

[11] J. Aguilar, J. Cordero, and O. Buendia, *Specification of the Autonomic Cycles of Learning Analytic Tasks for a Smart Classroom*, Journal of Educational Computing Research, vol 56, no. 6, pp. 866-891, 2018.

[12] M. Sánchez, J. Aguilar, J. Cordero, P. Valdiviezo-Díaz, L. Barba-Guamán, and L. Chamba-Eras, *Cloud Computing In Smart Educational Environments: Application in Learning Analytics as Service*. In New Advances in Information Systems and Technologies (A. Rocha, M., Correia, H., Adeli, P. Reis, M. Mendonca, Eds.), Springer, pp 993-1002, 2016.

[13] C. Rangel, F. Pacheco, J. Aguilar, M. Cerrada, and J. Altamiranda, *Methodology for Detecting the Feasibility of Using Data Mining in an Organization*, in proceedings of the XXXIX Conferencia Latinoamericana en Informática (CLEI), vol. 1, pp. 502-513, Naiguata, Venezuela, Octubre 2013.

[14] F. Pacheco, J. Aguilar, C. Rangel, M. Cerrada, and J. Altamiranda, *Methodological Framework for Data Processing Base on the Data Science Paradigm*, in proceedings of the XL Conferencia Latinoamericana en Informática (CLEI), Montevideo, Uruguay, Septiembre, 2014.

[15] K. Igor, and M. Kukar. *Machine Learning and Data Mining*. Horwood Publishing, 2007.

[16] C. Tran, C. Tseng, P. Chao, C. Shieh, L. Chan, and T. Lee, *Face Recognition under Varying Lighting Conditions: A Combination of Weber-face and Local Directional Pattern for Feature Extraction and Support Vector Machines for Classification*, Journal of Information Hiding and Multimedia Signal Processing, vol. 8, no 5, pp. 1009-1019, 2017.

[17] F. Wang, T. Xu, T. Tang, M. Zhou, and H. Wang, *Bilevel Feature Extraction-Based Text Mining for Fault Diagnosis of Railway Systems*, IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 1, pp. 49-58, 2017.

[18] C. Tran, M. Zhang, P. Andreae, and B. Xue, *Genetic Programming based Feature Construction for Classification with Incomplete Data*, in proceedings of the Genetic and Evolutionary Computation Conference, pp. 1033-1040, Berlin, Germany, July 2017.

[19] A. Koay, A. Chen, I. Welch, and W. K. G. Seah, *A New Multi Classifier System using Entropy-based Features in DDoS Attack Detection*, in proceedings of the International Conference on Information Networking (ICOIN), Chiang Mai, Thailand, January, 2018.

[20] L. Talavera, *An Evaluation of Filter and Wrapper Methods for Feature Selection in Categorical Clustering*. in proceedings of the International Symposium on Intelligent Data Analysis. pp. 440-451, Madrid Spain, September 2005.

[21] F. Peres and F. Fogliatto, *Variable Selection Methods in Multivariate Statistical Process Control: A Systematic Literature Review*. Computers & Industrial Engineering, vol. 115, pp. 603-619, 2018.

[22] O. Abedinia, N. Amjady, and H. Zareipour, *A New Feature Selection Technique for Load and Price Forecast of Electrical Power Systems*, IEEE Transactions on Power Systems, vol. 32, no. 1, 2017.

[23] Y. Bar, I. Diamant, L. Wolf, S. Lieberman, E. Konen, and H. Greenspan, *Chest Pathology Identification Using Deep Feature Selection with Non-Medical Training*, Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, vol. 6, no. 3, pp. 259-263, 2016.

[24] *Children Handwriting DataBase.* https://goo.gl/R1d6ZK.

[25] Project-Team IntuiDoc, *IntuiScript Project: Handwriting Quality Analysis, in Intuitive User Interaction for Document*, Technical Report, pp. 19–23, 2016. https://goo.gl/KsC9Xo.

Revista Venezolana de Computación
ISSN: 2244-7040
http://www.svc.net.ve/revecom
Vol. 5, No. 2, pp. 12-22, Diciembre 2018
Selección de los Mejores Artículos de CoNCISa 2018

NANO-Communication Management System for Smart Environments

Alberto Lopez Pacheco[1,3], Jose Aguilar Castro[2,3]
albertolopez@ula.ve, aguilar@ula.ve

[1] Doctorado en Ciencias Aplicadas, Universidad de Los Andes, Mérida, Venezuela
[2] Departamento de Computación, Universidad de Los Andes, Mérida, Venezuela
[3] CEMISID, Universidad de Los Andes, Mérida, Venezuela

Abstract: This paper describes a communication platform for the middleware nanodata processing. This communication platform is based on a hierarchized protocol, called the Communication Management Nanolayer (CmNA), which is a layer of the ARMNANO middleware. ARMNANO is an autonomic architecture, which defines mechanisms to deliver proper services from nanostructures in smart environments, using nanodevices such as nanosensors and nanoactuators. In this paper is described the CmNA layer, which is composed of 3 classes that are, the Protocol Management System, the Ontology of the Physical Communication Management and the Abnormal Situation Detection Mechanism. Our proposal can be used in contexts like smart cities, Internet of Things, and Internet of NanoThings. In this sense, we demonstrated the applicability of CmNA in two case studies in the context of medical care.

Keywords: Nanocommunication Platform; Electromagnetic and Molecular Communication; Molecular Nanocommunication; Nanosensors and Nanoactuators; Reflexive Middleware.

I. Introduction

Nanotechnology is evolving rapidly nowadays. It involves the synergy of different fields such as informatics, nanomaterials, and communication systems, among others [1]. The nanotechnology has enabled the introduction of nanodevices, with sensing, action, processing and communication capabilities. The network of communication of nanodevices is a nanonetwork that can accomplish complex tasks, such as health monitoring.

Data processing and transmission at the nanoscale pose big challenges. On one hand, it is a difficult task to adapt the complex circuitry with transistors, logic circuits, capacitors, memories, etc. to nanodevices, due to the size and possible signal overlapping [2]. By the other hand, data transmission responds to different rules at the nanosize, due to difficulties in resonation. It has generated innovative communication protocols, such as the Electromagnetic Communication (EMC) or the Molecular Communication (MC) [3]. Herein, it will be explained these aspects in the ARMNANO architecture, to provide services in a nanoscaled smart ambient.

This paper proposes a nanocommunication management system, in order to manage the different communication protocols for data transmission and logical treatment of the sent data through the nanodevices deployed in a smart environment (AmI). Previously, we have proposed a reflective middleware for the management of nanodevices in AmI, called ARMNANO, in which several elements were defined. The Communication Management Layer (called CmNA) is one of them, which must provide the management of the communication platform, in order to collaborate with the autonomy, connectivity, and self-adaptive properties of ARMNANO [4]. In this sense, CmNA must essentially filter, route, and transfer the data between the nano-devices in the AmI, and mainly provide communication services for the Data Analysis Tasks of ARMNANO, carried out by its Data Analysis Smart System (DASS).

Particularly, we use the abstraction that is provided by the agent theory (which allows defining specialized structures, such as nanodevices), for the processes of monitoring at the nanoscale, sending information at the molecular level, synchronization of nanodevices, self-detecting of nano-failures, among other aspects, in order to define the CmNA, as a crucial part of the ARMNANO architecture.

In previous works have been proposed communication platforms for the nanodevices. Section II carries out a description of some of these works, based on their nanonetwork characteristics and their elements. CmNA is a crucial layer of the ARMNANO middleware. It aims to transmit the data between the nanodevices deployed in an AmI, for which is required the data processing, the information routing, the data authentication, among other services. This a novelty respect to

previous publications since it is based on autonomic properties, in order to connect the nanodevices.

The organization of this paper includes related works. At the Section III is presented the theoretical communication framework in a typical nano-context architecture. Also, this Section presents ARMNANO, the reflective middleware for the management of nanodevices in an AmI. In Section IV is described the physical management system of CmNA and its different components. Section V focuses into the logical management system of CmNA. Last Section states the definition of the case studies and how can be used CmNA in them.

II. Related Works

Nanosensors and Nanoactuators are devices that need to cooperate and interchange information for the synchronization of their tasks at the nanolevel. This is even more crucial when these nanostructured nodes are inside the body, at such cases, they mainly communicate horizontally (peer-to-peer) and with the exterior (information travelling in-to-out the body). It requires to set nanonetworks that properly connect a nanoscale mesh with the Internet capabilities to send and receive data [3], or to invoke the terahertz band [5].

The terahertz band communication is mainly based in the Electromagnetic Communication within nanodevices. It presents advantages such as a broader bandwidth transmission, non-physical constraints interference, less-noised data channeling and more efficient connection within materials assembled at the nanolevel [6]. The terahertz band permits to overcome challenges presented in the classical communication, such as propagation modeling, modulation problems, etc. [7]. The CmNA architecture replicates the aforementioned advantages. Through CmNA it is decentralized the nanodata handling, improving the communication capabilities among the ARMNANO levels.

Gubbi J. et al. [8] state that Wireless Sensor Networks (WSN) are the elegant and relevant element to the assembly of nanostructured platforms. In specific, the Internet of Things (IoT) represents an outstanding technology changing connection between persons and objects. The IoT represents the extension of the ubiquitous computing, which is a field that incise directly in the nanotechnology field to generate an on time proper detection, as well as the command execution with independence and accurateness.

Akyildiz et al. [9] have introduced the Internet of Nanothings (IoNT) concept, as a general architecture for EMC device. They describe the components more suitable for nanoscale communication, focusing on graphene-based nanoantennas, which are most efficient in the terahertz band. Also, they define how to do the channel modeling, information encoding, and novel routing protocols, as well as services that would be required for EM-based nanocommunication. This approach had, however, issues related to path loss and noise, resulting from molecular absorption.

Elsewhere, Akyildiz et al. [10] present the description of the nano-communication protocols, ranging from the electromagnetic to the molecular approaches, showing the inherent advantages of each one. Among the possible nanonetwork architectures, they analyze the intrabody networks and their links with the IoT. Inside of the nanodevices-based architectures, the authors include nano-nodes, nano-routers, nano-microinterface devices, and gateways, which in conjunction establish the integral functionality at the architecture. This model constitutes a reference for the CmNA definition.

In [11] is analyzed the interconnection of multimedia nano-devices with existing communication networks, and defines a communication paradigm, called the Internet of Multimedia Nano-Things (IoMNT). The paper presents the state of the art and major research challenges in the IoMNT, in terms of multimedia data and signal processing, physical layer solutions for terahertz band communication and protocols for the IoMNT, propagation modeling, etc. They propose a novel QoS-aware cross-layer communication module, access control techniques, neighbor discovery and routing mechanisms, addressing schemes, and security solutions, for the IoMNT.

In [12], the authors analyze two major challenges to implement the IoNT paradigm, the definition of the data collection and routing mechanism in nanonetworks, and the design of a middleware that connects to nanonetworks with conventional microsensors. Also, they define the requirements to extend current communication management systems to support the IoNT, as well as some IoNT applications.

Chude-Okonkwo et al. [13] present a survey about targeted drug delivery (TDD) within the Domain of MC. They describe how MC-based TDD concepts differ from traditional TDD in the field of medical science. They present a taxonomy of the different aspects. Also, they present models and requirements for developing MC-based TDD systems. The clinical implementation is highlighted and its software tools, as well as the standards and regulatory policies in their practical contexts.

In [14], they propose the design of a mobile ad hoc molecular nanonetwork (MAMNET) with electrochemical communication. MAMNET consists of mobile nanodevices and info-stations that share nanoscale information using electrochemical communication. Additionally, they propose an analytical model to examine the effect of mobility into the performance of electrochemical-based communication.

In the field of MC, in [15] is proposed an approach of communication between nanodevices using bacteria communication nanonetworks. This is possible due to the bacteria properties: 1. Biased motility toward the destination through chemotaxis process, and 2. The ability to transfer genetic information using bacterial conjugation. In this paper, they propose an opportunistic routing process in bacteria communication network based on these two properties. The paper scrutinizes classical metrics used in communication networks, such as the average delay and the number of messages.

In [16], the authors analyze the integration of the communication and networking functionalities in the MC, in order to utilize these natural systems to create an artificial biocompatible communication network that can interconnect nanodevices in different parts of the body with the cloud, which they have called the Internet of BioNanothings (IoBNT). In this

sense, they present a nanonetwork approach for cellular tissue based on the Ca^{2+} signaling process. They highlight the performance of the Ca^{2+} signaling-based molecular communication system for cellular tissue, in different contexts.

In [17], the authors propose a health monitoring ubiquitous approach in situ, where the nanodevices can be connected through the Internet, or either, the terahertz band. Elkheir et al. [17] propose a nanodevices-based middleware comprising essentially the *application layer*; in which occur the measurement, the *transport layer*; which control the noise in the channeled data, the *network layer*; where all the connections and communication protocols are selected, and the *MAC/PHY layer*; where the coding and data analysis are join.

In general, engineering biological nanostructures can connect biocompatible organs inside the human body [10], and allow information carriers to transport themselves. They are DNA, RNA, aminoacids, proteins, cells, or nucleotides. Nano-adapted devices can be embedded into the body and communicate the changes between them, or to external units (outbound communication) that process the signal, and generate orders in return.

The communication in this context is an open domain of research that our CmNA approach takes into consideration. The communication at the nanostructure level is a crucial area to develop, in order to fuse it with informatics platforms, such as ARMNANO. The pertinence of this publication relies on the nano-communication platform developed in the CmNA layer.

III. Theoretical Framework

A. Bases of the Nanocommunication

Nanocommunication refers specifically to the information transmission among nano-devices or nano-objects. For the interconnection of nanodevices is required a nanonetwork or nanoscale-based network, which extend the capabilities of a single nanodevice, both in terms of complexity and range of operation, by allowing them to coordinate, share, and fuse information. Nanonetworks are necessary for the application of the nanotechnology in different domains, such as biomedical and industrial applications, among others. The classical communication paradigms for communication in the nanoscale are:

1) Electromagnetic *Communication (EMC)*: it is based on the transmission and reception of electromagnetic radiation from the nanodevices. In general, this kind of communication occurs based on the transmission of a wave, which find a proper transceiver in the Terahertz band. Some examples of nanoscale electronics are nanobatteries, nano-memories, nano-antennas, and nanoscale energy harvesting systems. From the communication perspective, the properties of the nanomaterials define the specific bandwidths for emission of electromagnetic radiation, the time lag of the emission, among other things. Mainly, there are two alternatives for electromagnetic communication in the nanoscale, according to how is generated the wave in the Terahertz band to resonate at this level:

- *Nanoradio*, which is an electromechanical carbon nanotube that can decode an amplitude or frequency modulated wave. In this way, it is possible to receive and demodulate an electromagnetic wave.
- *Nanoantennas*, graphene-based nanodevice that act as transceivers in the Terahertz band.

In this sense, the information is transmitted in packages similar to the principles of the photoelectric effect.

2) Molecular Communication (MC): it is a communication paradigm that uses biochemical signaling to achieve information exchange among naturally and artificially synthesized nanosystems. In general, MC carries the transmission and reception of information by means of molecules. There are different molecular communication techniques according to the type of molecule propagation.

- In flow-based molecular communication, the molecules propagate through diffusion in a fluidic medium. The hormonal communication through blood is an example of this type of propagation.
- In walkway-based molecular communication, the molecules propagate through pre-defined pathways by using carrier substances, such as molecular motors (they are biological molecular machines that are the agents of movement in living organisms. Some examples are the myosins, the dynamism, and the RNA polymerase) and bacteria.
- In diffusion-based molecular communication, the molecules propagate through spontaneous diffusion in a fluidic medium. Some examples of diffusion-based architectures are the calcium signaling between cells, the quorum sensing among bacteria, or the pheromone communication in a fluidic medium, such as water, or air.

In general, for the nanodata transmission, the communication platforms necessitate to fit the following conditions:

- *Straightforward Functionality:* the structure must accomplish the simple task of transmission, without the intention of affording a secondary task. In order to do that, the platform requires of nanorouters that are data transmitter channels (no saving information or treating it in any form), leaving the nanosensors and nanoactuators to carry out their crucial tasks. They represent devices that route the information between nanodevices in the AmI. Additionally, it requires of microgateway units, which route and authenticate the information in an appropriate fashion.
- *Overcoming Physical Barriers of the AmI:* data have to travel against these physical constraints, avoiding noise and decreasing uncertainty. For example, in the context of human beings, typically possesses organs, tissue, cell walls, dynamic fluids, such as blood, fatty, water, or different gradient solutions, which must be considered by a nanocommunication platform.
- *Synchronicity:* nanodevices always act in a group, this is, it means that they proceed in a synchronization fashion. Thus, the nanorouter that selects the channel to transmit the information to the appropriate nanodevice, must guarantee it.

B. ARMNANO

The ARMNANO architecture is a multilayer architecture that provides services for nanodevices in an AmI, hence offering the ubiquity, real monitoring, interconnecting, self-learning, as key properties that enrich its performance [4].

ARMNANO architecture is organized in two levels [4] (see Figure 1): the base level and the meta level. In addition, ARMNANO has a transversal structure, called the Data Analysis Smart System (DASS), which is in charge of performing data analysis tasks using nano-data, delivering appropriate services to the real place in real time, for the different agents in the AmI. Nevertheless, the platform as a whole is decentralized and autonomic, in order to allow the self-configuration for the data transmission and commands delivery.

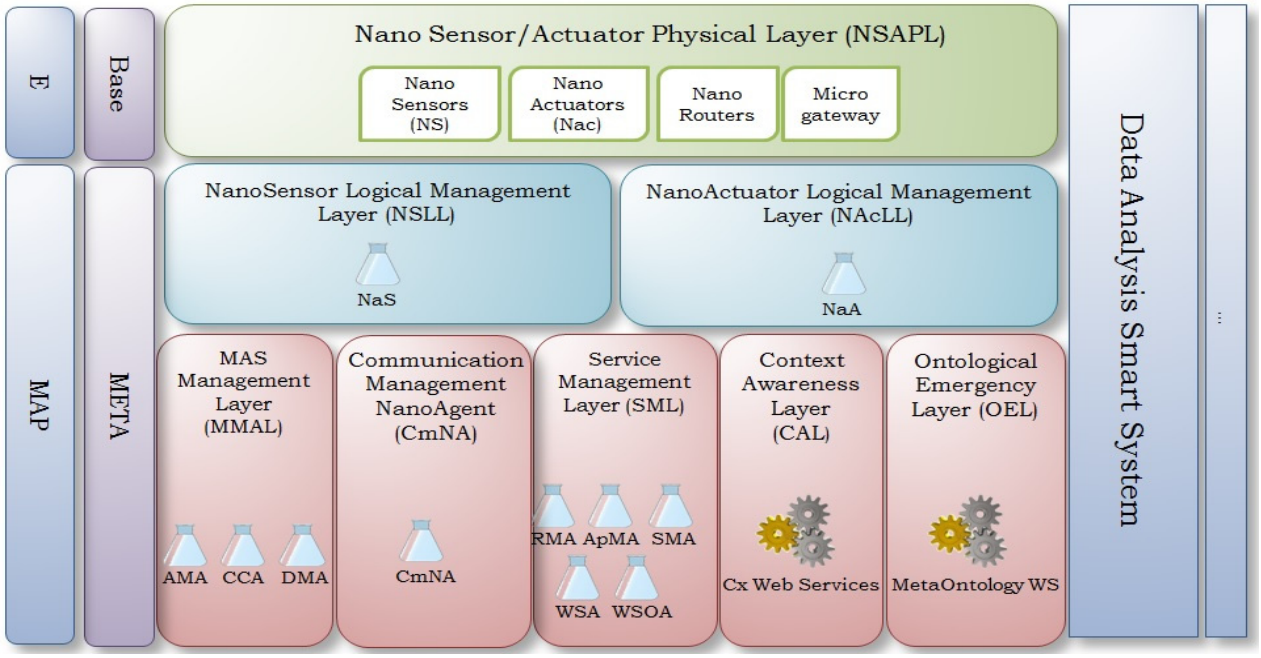

Figure 1: ARMNANO Architecture [4]

The base-level, named NSAPL, contains the physical devices in the AmI, such as the nanodevices and the microgateway. Then, the abstract views of the nanodevices in the NSAPL layer are deployed as logical agents, called NaS and NaA. Additionally, at the meta-level ARMNANO has 5 layers, which are MMAL, CmNA, SML, CAL, and OEL, that provide services respect to context characterization, cloud connection, among other things.

We describe briefly each layer in ARMNANO as follows.

1) Nanosensor and Nanoactuator Physical Layer (NSAPL): in this layer are deployed the nanosensors (NS), nanoactuators (Nac), nanorouters, and microgateway.

- *Nanosensor (NS)*: it refers to the nanodevices that perform the measurement.
- *Nanoactuator (Nac)*: it refers the nanodevices that receive the command to execute in situ, at the structure in that it deploys.
- *Nanorouters (Ro)*: these are the nanodevices that control the information routing and organize the data transmission chain.
- *Microgateway (Mc)*: this is the unit to authenticate the information from nanosensores. Thus, it decides if the data pass to the upper layer in the architecture, or on the contrary the nanosensor node has to perform a new measurement. At this level the iterations will continue till a certified value is approved.

2) Nanosensor Logical Management Layer (NSLL): it is composed of the logical view (it is an abstraction) of each NS agent in NSAPL.

3) Nanoactuator Logical Management Layer (NAcLL): it holds the abstractions of each Nac agent defined in NSAPL.

4) MAS Management Layer (MMAL): it is a multi-agent architecture with the agents AMA, CCA and DMA defined previously [18][19][20], which manage the community of agents.

5) Communication Management Nanoagent (CmNA): it addresses the control the communication protocols and the authentication protocols to the nanoscale. This layer represents the core of this publication.

6) Service Management Layer (SML): it connects the Multi-agents (MAS) and service-oriented application (SOA) paradigms. It is a crucial layer to deploy web services (for more details about this layer, see [21]).

7) Context Awareness Layer (CAL): it deploys context-based services pointing to context discovery, modeling, and reasoning (for more details about this layer, see [21]).

8) Ontological Emergence Layer (OEL): it defines a set of services to allow the emergence of ontologies (for more details about this layer, see [21]).

As aforementioned, this paper defines the CmNA layer. CmNA covers all the aspects related to the data transmission involving the nanodevices, such as the NS, NA and Ro, coupled among them, to deploy the capabilities of the NSAPL layer at ARMNANO. In addition, *microgateway* is the processing unit located upon the nanorouter, which assures the logical significance of data measured in the nanosensors. Thus, CmNA is a crucial and a novel layer in a middleware involving the component nano. Figure 2 illustrates the different elements of the CmNA layer. In the following Sections, we describe these components of CmNA.

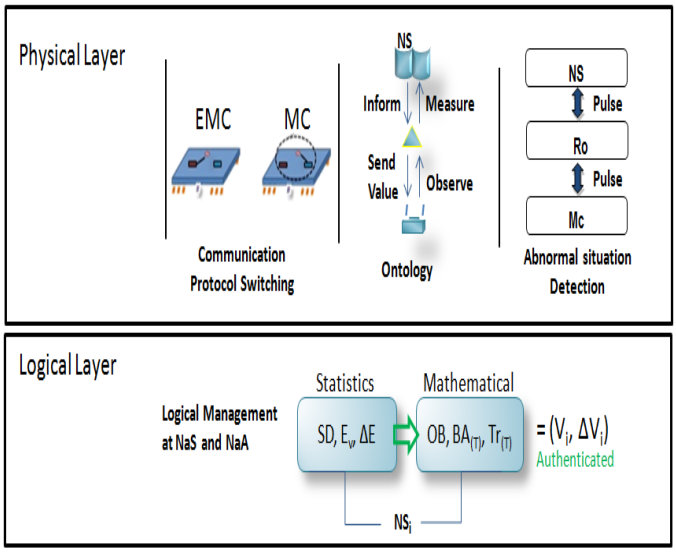

Figure 2: Components of the CmNA

IV. Physical Communication Management

In this Section, we describe the different components of the physical layer of CmNA.

A. Communication Protocol Management System

CmNA layer is in charge of defining the communication protocol, according to the detection mechanisms required in a given moment in the AmI. The nanosensors work in a hybrid paradigm, sometimes to detect a focused target (accumulation) or a dispersed target (relaxation), and the protocol must change to support this ability. Thus, the Communication Protocol Management System must have the capability to switch the protocol according to the detection mode required in situ. To do this, it is based on the next elements:

- *Nanosensor and* Nanoactuator *Behavior:* these devices carry out a single operation. They are in charge of measuring or execution. To this end, the nanosensors perform a single operation to make the detection. In the case of nanoactuators, they have a mechanism to destroy (fatty or coagula), to order (cells, nucleotides or proteins) or to command (bacteria or microorganism migration, in-and-out motion to a cell).
- *Nanorouter:* graphene-based devices ranging the nanoscale in charge of selecting the less trafficked microgateway unit.
- *A Protocol Selection Mechanism:* it is based on a switching procedure that detects the in-situ context of the place in which is performed the measurement, in order to select the Communication Protocol: EMC or MC. It selects the protocol according to a spectrophotometry method launched in the context of the target, the source of the target, or either the organelles at which the target is located. To determine this, it defines a spatial standard deviation of the above structures, which define two states:
  - *Accumulated Context (Focused System):* for this case the spatial standard deviation is below the 0,1 threshold. Then, it selects the EMC protocol.
  - *Relaxed Context (Dispersed System):* for this case the spatial standard deviation is above the 0,1 threshold. Then, it selects the MC protocol.
- *A Communication Protocol Monitor*: it starts after the communication protocol is selected, either EMC or MC. It can change the protocol in runtime using a similar rule to the used by the selection mechanism.

Note that in the case of the EMC protocol, the nanosensors measure, but in the case of the MC protocol, the nanosensor is a gateway (see Figure 3). That is, a typical NS node will possess a hybrid function. In autonomic platforms, this task is crucial since this switching must be performed with no human intervention. Thus, a NS node must be adaptable to the context, which contributes to the complete autonomy and decentralization of the architecture.

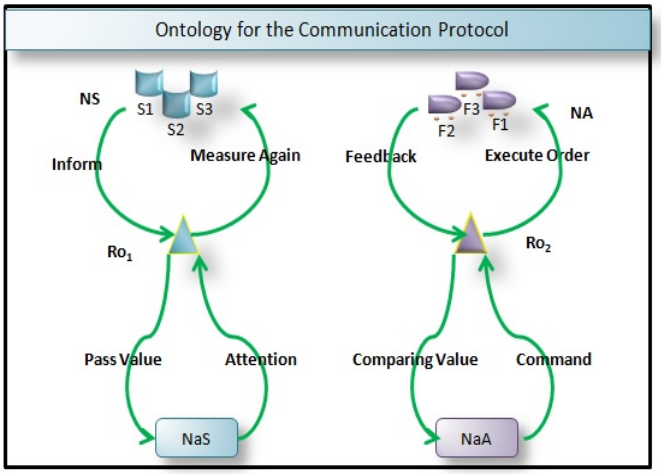

Figure 3: Hybrid Communication Protocol. Switching among EMC vs MC

It is possible to distinguish several steps of the Communication Protocol Management System:

- *Observation*: the nanosensors acting locally are enabled to implement the spectrophotometry method, to describe the target respect to the dispersion level.
- *Adjustment*: since the target distribution is not steady, nanosensors can determine a new dispersion level locally to cover the target, which must define a new communication protocol adjusted in real time.
- *Execution*: the measurement is made. The EMC model typically involves electronical transitions at the terahertz frequency range, which are not interfered by physical barriers. The MC model considers nanosensors as gateways, so it allows the target passage.

B. Ontology of the Physical Communication Management Level

The CmNA layer requires an ontology, in order to allow the interchange of messages for communication tasks among the nanorouters, nanosensors, and nanoactuators. This ontology is based on the ARMNANO ontology, which is being described in other work, and is shown in the Figure 4. The CmNA layer instances this ontology, in order to allow the communication.

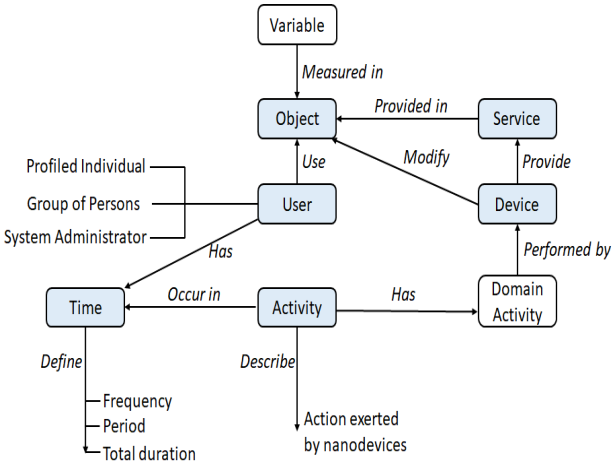

Figure 4: ARMNANO Ontology

The main concepts of this ontology used by this layer are: the devices that describe the different components of the CmNA layer (like the sensors, routers, etc.), the "activity" and "domain Activity" concepts that describe the process of nano-communication, and the "variable" and "users" concepts that describe the nano-environment (nanodata, etc.).

C. Information Organization System

In general, NS and NA send discrete packages of information in the form of:

NS		NA
$[Ro_1 ; V_1 ; T_1]$		$[Ro_2 ; F_1 ; T_2]$
$[Ro_1 ; V_2 ; T_1]$	and	$[Ro_2 ; F_2 ; T_2]$
$[Ro_1 ; V_3 ; T_1]$		$[Ro_2 ; F_3 ; T_2]$
...		...
$[Ro_i ; V_i ; T_1]$		$[Ro_j ; F_j ; T_2]$

The packages of information are multiple triplets $[R_{oi}, V_i, T_k]$, where $\mathbf{R_{oi}}$ is *informed* by NS; $\mathbf{V_i}$ is the sensed variable, and $\mathbf{T_k}$ is the instant of time. Several successive instants **T** describe a continuous fact. In the case of NA node, the triplet is formed by $[R_{oj}, F_j, T_h]$, where $\mathbf{F_j}$, refers to the *feedback* within the connection NA-R_{oj}. Each package represents the reporting of a variable **V**, to the same $\mathbf{R_o}$, in a given instant of a single nanodevice.

Data values are typically sent in this format to the Mc unit. Once all the NSs have reported, then the authentication is applied to the data in this specific node, as it will be explained in Section V. Thus, each node reports a single authenticated value at an instant T. In this sense, it is possible to see a contextual reality from different nodes at different instants T, which for sure generate a higher volume of data, but with a decentralized data management as ARMNANO, make affordable this multi-perspective approach.

In this way, the NaS and NaA agents, which host the statistical tools to apply in the authentication of the information, have the information well organized, such that the agents can exert their capabilities in a proper fashion. Each statistical function is represented as f(Vn or Fn), and the idea is to determine if the data correspond statistically to the studied context. If it corresponds, then the data is sent to DASS (see Figure 5).

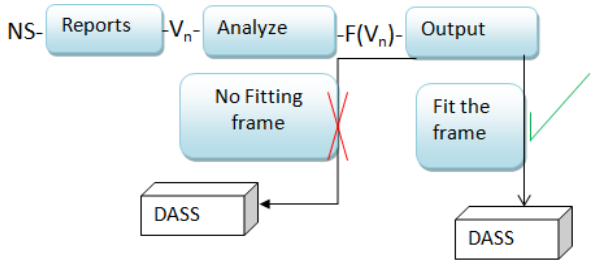

Figure 5: Contextualization of the Data

D. Abnormal Situation Detection Mechanism

Due to that the middleware is autonomous, it requires to evaluate the connection of the different components in the NSAPL layer. For that, CmNA uses a reference signal that permits testing, comparing an entry signal VS an outgoing signal. This assures that every component is indeed available and connected (wired or non-wired). The entry signal is transmitted by pulses. A *pulsed signal* corresponds to the features of a numerical value or wave shaped signal at a certain frequency of time. The sent pulse allows the following:

- Test that the intensity and magnitude of the entry signal to match the output signal.
- Assure that the range of working for the components is operationally correct.
- Confirm if the nanodevices are connected or not.

The test of the operational state of the CmNA components starts at the Mc unit. A pulse-after-pulse is sent up to NS or NA. If these nanodevices recognize the pulse signal, then they answer as is expected. This is translated as Mc, Ro, NA and NS are properly connected and working. In order to track and detect a

failure, it is required to define a task assignation for the components of CmNA (see Figure 6).

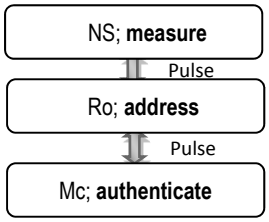

Figure 6: Task Assignation of the Components of CmNA for a Communication Failure Detection

Each of the components at Figure 5 possess a pulse-sending mechanism, so they push the signal into Mc, which act here, as a receiver. Therefore, Mc act here as transceiver (transmitter and receiver) and analyzer of the signal, the node NA or NS act uniquely as transmitter, as well as Ro. The Abnormal Situation Detection Mechanism (ASDM) uses the following rules to determine if there is a failure:

1. If NS sends a pulse and Mc receives, then it is supposed that every component is ok.
2. If Mc does not receive the pulse from NS. Additionally, if Ro sends a pulse to Mc and Mc receives it, then it is supposed that NS has a fail but the rest of the components are ok.
3. If Mc does not receive the pulse of Ro, then nor NS not Ro are operative. So, Mc orders to the NaS agent migrates to another NS node with its Ro, and the verification process is executed again.

V. Logical Communication Management

The final objective is a CmNA layer that decentralizes the middleware execution. CmNA layer must facilitate two aspects: **1.** A decentralized *Data Transmission* and **2.** A decentralized *Data Authentication*. The physical communication management level allows the decentralization of the data transmission (see Section IV), and in this Section, we present the decentralized of the data authentication in CmNA.

The logical management of the data collected from the nanodevices is performed at the CmNA in order to authenticate them, because it has to send truthful information to DASS. In this sense, the autonomic platform assures the numerical value is indeed characterizing the reality. In this Section, we aim to clarify the recognition and authentication of the data that have been generated at the NS and NA nodes. DASS resides in a physical unit or in the cloud, in order to carry out data analytics tasks (DASS will be treated in an upcoming publication). Herein is specified the logical behavior in Mc.

A. Data Authentication at the Statistical Domain

Statistical assessment is crucial to the data certification [22]. As ARMNANO is an autonomic system, it will validate the data to assure it corresponds to a description of the observed object. The system recognizes the values to be statistically accurate.

Statistical treatment includes the Standard Deviation (SD) and Relative error (Ev). Particularly, these functions are necessary because a NS node will iteratively measure V_i, up to the authentication of it.

The statistical analysis will ensure *precision* and *exactitude* in a dataset. The more the amount of measurements of V_i, the greater the confidence in the nanodata assessment. Statistics are constantly applied to V_i in the NaS or NaA agents (see Figure 4). So, at the time when SD and Ev have a valid value, then V_i is statistically validated. For this purpose, the statistical frame of authentication of V_i, $\mathbf{[V_i \pm \Delta V]}$ have to accomplish the 2 conditions below [23]:

$$Standard\ Deviation, \quad SD < 0.1 \tag{1}$$

$$Percentage\ Error, \quad Ev < 5\% \tag{2}$$

A SD below 0.1 means that the nanosensor is observing the same value (statistically, it is assured the precision is high). Furthermore, if Ev is below 5%, then V_i is near to its typical measured values in the observed target, which indicates a high exactitude. In this context, the value V_i is authenticated when precision and exactitude are high.

In particular, at this stage will exist an iterative cycle in which the dataset is proved to be descriptive of the observed reality.

B. Data Authentication at the Mathematical Domain

At CmNA, the mathematical domain is carried out to complement the statistical assessment aforementioned. CmNA layer must permit the formulation of the following questions: *Does the data guarantee the description of real organs? Are the values truthful?* For a correct response to these questions, it is necessary a new concept, called Object Logical Recognition (OLR). The OLR defines a set of mathematical operations applied to the data collected in a Mc. These mathematical operations are executed at the layer NSLL and NAcLL through their agents. They are [22][24][25][26][27]:

- *Outranged Behavior*: the magnitude for each variable have a normal oscillation, described by the measurements performed during a long-time range. Going beyond the range, is known as outranged behavior, and is translated as a failure. Outranged Behavior (*OB*) can be defined mathematically as,

$$OB_{(x)} =: {}^{X_F}_{X_o} \tag{3}$$

  This equation establishes the normal behavior (oscillation) of the variable $V_{(x)}$, within the limits X_o and X_F. $OB_{(x)}$ indicates authentication when the observed value V_i oscillates between the range X_0 and X_f.

- *Banded Activities:* mathematically talking, the performance is parameterized within a down and an upper limit. It can be used from an outsider to oblige to a given structure to behave accordingly to a standard, or simply to adapt. This is important in a human system that possess flexibility and adaptability capabilities. Both features can be used for instance, to an unpredicted change or simply, to induce an

organ to behave properly when facing novel surrounding conditions. Banded Activities (*BA*) can be defined mathematically as

$$BA_{(x)} =: \overset{X_F}{X_i} \quad (4)$$

This equation establishes the limit values of the variable $V_{(x)}$. $BA_{(x)}$ indicates authentication when the observed value is within the range X_i and X_f. Note that, $OB_{(x)}$ and $BA_{(x)}$ describe different things. They are essentially differentiated based on the amount of data collected. $OB_{(x)}$ is a short-term memory determined by the current context, meanwhile $BA_{(x)}$ is the description of the variable in a long term, referred to a period between 3 to 6 months monitoring.

- *Trending*: The output data sent for authentication can hold patterns or repeated values, so it can form an object awareness at the logical level. This object awareness can be employed to define a target, to know the evolution of the system, etc. Trending (*Tr*), can be defined mathematically as

$$\lim_{x \to t} Tr_i(x) = V_i(T) \quad (5)$$

It represents the tendency of V_i at the time. For instance, people reach the maximum height at 21 years old, thus V_i represents the people's height when **X** tends to 21, and *Tr(T)* describes the tendency of the human height. *Tr(T)* indicates authentication when the observed value V_i is below $V_i(T)$, where $V_i(T)$ is the maximum possible value for this variable in the time line.

The functions in the mathematical domain aim to formulate the typical behaviors in a given context.

VI. Case Studies

Herein will be analyzed 2 cases of study to demonstrate the capabilities of CmNA in the health area. In ARMNANO, the monitoring is carried out in an autonomic fashion, with self-corrective actions and a smart data treatment. These features conjugate the advantages of non-invasive nanostructures diagnosis and actions (through nanosensores and nanoactuators), and the smart data authentication deployed in the Mc unit. The potential of ARMNANO can be observed in the health area, to contribute in personalized medicine, automatic diagnosis, remote patient's treatment, and low-cost medicine.

A. Context

The general scenario deals with a person involved in a traffic accident that holds multiple injuries and traumatisms along the body. ARMNANO is deployed in order to diagnose properly the condition of the individual. Due to injuries are of different magnitude and danger, is necessary based on the flexibility of our middleware, to adapt at the context and provide solutions autonomously. To this end, is required to inject the NS nodes that sensible to detect inflammation and infection level.

There are 2 groups of nanosensores injected, identified as NS1 and NS2, that will be in charge of monitoring the gravity at each leg. In this group is added a third node, identified as NS3, in charge of monitoring at the upper Section of the individual, such as neck and head, to analyze lesions in there. In this sense, it is assembled a network of 3 nodes which are not conscious among them, but that provide complementary information to diagnose the individual status.

An external observation is also used, to monitor fixed variables in this injured patient. Externally, it is measured the body temperature, the body heat distribution and the blood pressure.

- *Body Temperature* signals if there is an infection process at the individual, due to the multiple wounds [28].
- *Body Heat Distribution* signals the evolution of the inflammation and the distribution of the coagula alongside the body. It determines the recovery of the skin and internal tissue affected by the blood migration [29].
- *Blood Pressure* regulates the normal performance of the body, respect to the lost amount of blood. This variable represents the oxygen bomb to keep active the body that work in a regular fashion in each individual [30].

B. 1st Scenario

There are 3 NS nodes describing the evolution in this scenario, which essentially monitor in it. Respect to the condition mentioned above, the main injuries to be monitored will be tracked with NS_1, NS_2 (at the legs) and NS_3 (at the head). The form to capture data is represented as,

NS_1 ; Ro
NS_2 ; Ro
NS_3 ; Ro

The sensors monitoring at each instant **T**, reports to Ro. In general, NS agent node **informs** Ro agent, and it **sends the value** to NaS agent. This last agent applies the statistical and mathematical analysis, to authenticate the data. If the data are authenticated, then it is sent to DASS, on the contrary, NaS sends **observe** again to Ro agent, and Ro agent sends **measure again** to the NS agent (see Figure 7).

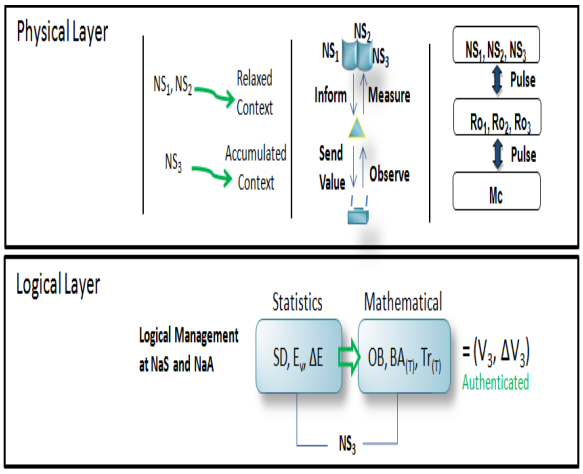

Figure 7: CmNA Deployment at the 1st Scenario

Due to the length of the leg, the selected communication protocol should correspond to the MC mode. Therefore, nanosensors NS_1 and NS_2 arrange in a dispersed fashion to act as gateways, and detect the gradient variation in the target (see Figure 7).

Note that NS_1 is assigned to the left leg, and NS_2 to the right leg, and they are distributed alongside each leg to communicate the gradient of histamine. The deployment of NS_1 and NS_2 obey to the dispersion of the wounds along the legs. NS_1 and NS_2 will report the gradient of histamine circulating in veins and arteries in the legs. The greater this level, the more advanced the inflammation. Thus, the information coming from NS_1 and NS_2 will be classified by DASS as *low*, *moderate*, *overloaded*. In this case study, it is supposed as moderate at both NSs. This means the existence of certain alarm, given the inflammation.

For the purpose of NS_3, it must determine the situation locally, thus the communication protocol is EMC at the terahertz band (*accumulated context*). NS_3 reports the head's condition. The information at NS_1, NS_2, NS_3 can be collected in parallel. NS_3 can report different measures of different instants of time,

NS_3: **[***0.606*, 0.696, 0.645, 0.689, 0.701, 0.652**] ppb ; Ro_3**

The typical value of the histamine level is 0.617 ppb (part per billion) [31][32][33][34]. CmNA can carry out the statistical treatment of this information (see Table I). At the Table II is established the characterization of the measurement for NS_3 for an interval of time. Given that **SD**<0.1 and **Ev**<5%, then **V_i=[0.665 ; 0.064]** is **authenticated.** So, this value is sent to DASS (see Figure 5).

Table I: Statistical Characterization of NS3

Statistics to NS_3	**Magnitude**
Average (A)	0.665 ppb
Absolute Error (ΔV_n)	0.064 ppb
Relative Error (E_v)	4.7 %
Standard Deviation (SD)	0.037 ppb
OLRs to NS_3	
Outranged Behavior (OB)	$OB_{(V)} =: {}^{0.648}_{0.586}$
Banded Activity (BA)	$BA_{(V)} =: {}^{0.698}_{0.501}$
Trending (Tr)	$\lim_{Vi \to T} Tr(T) = 0.671$

Also, Table I signals that the values for *BA*, that is ranged similar to *OB*, includes the value V_i in it. And *Tr(T)* was determined as 0.671, which is above the V_i. So, the measurement at NS_3 can be approved by the OLR assessment, that is, V_i is *authenticated*.

According to the above mentioned, it has been tested the properties of the CmNA platform to provide the communication management of nanostructures in a flexible and adaptable health monitoring application. Similar achievements can be performed in contexts like smart cities, military sector or transportation.

CmNA can detect a failure when is transmitted the information. For that, it uses the rules defined in its detection mechanism (see Section IV). The detection mechanism carries out an analysis of the available resources analyzing the [Device, task] correlation. In our case, we suppose that the NS_1 does not send signals to Mc, but it receives pulses from Ro_1. In this case, the second rule is activated to determine that there is a failure in the NS_1 nanosensor. In this way, it can determine if the failure is in one of the NS nodes, of the nanorouter or microgateway, in which case, it migrates to a parallel device in an autonomous fashion, or request external human intervention.

C. 2nd Scenario

In this scenario is added an outside monitoring at the individual, using nanosensors at a smart room to monitor variables such as, the *Body Temperature*, the *Body Heat Distribution* and the *Blood Pressure* (see Figure 7). Table II shows the values measured at the instant T with the outside nanosensors.

Table II: Typical Measurements Performed at the Instant t with Outside Nanosensors

Statistics to NS_3	**Magnitude**
Body Temperature	(37 ± 1)°C
Body heat Distribution	Extremities (heat distributed) and Head (heat localized) have become abnormally high.
Blood Pressure	70 (diastolic) - 110 (systolic)

The outside nanosensors measure in parallel to NS_1, NS_2, NS_3, making a complementary information, in order to represent a proper characterization.

The Figure 8 shows the deployment of CmNA in this case. There is not a switch of the communication protocol, because the outside nanosensor uses an external communication protocol to communicate its information. Additionally, the data authentication is carried out using both approaches, statistical and mathematical, in the agent that represents this new nanosensor. In this way, the platform couples the *internal sensing* (1st case) with the external sensing (2nd case). Thus, CmNA proposes a decentralized approach to the deployment of the overall architecture.

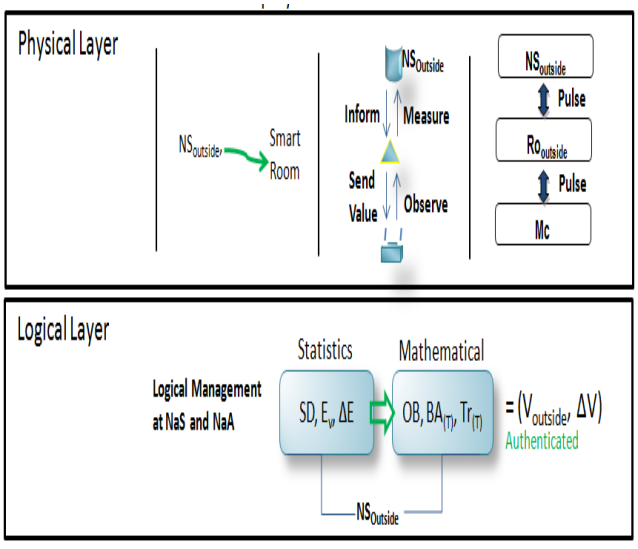

Figure 8: CmNA Deployment at the 2nd Scenario

VII. Conclusion

At the present paper, it has been detailed the deployment of the CmNA of the ARMNANO middleware. This layer selects the proper communication protocol according to the characteristics of the dispersion level of the context. Thus, the communication protocol can be based on molecular labels, or simply be supported by the electromagnetic spectrum, sending pulses at either infrared or UV band. Particularly, CmNA allows that the sensors act as Nanosensors (Accumulated Context) or gateway (Relaxed Context), as a consequence of the Communication Protocol Switching, according to the EMC or MC model used.

At CmNA, the logic agents NaS and NaA can authenticate the data based on statistical and mathematical approaches. They are complementary, which can improve the quality of the data. Additionally, the 1st and 2nd scenarios showed a complementarity among both, the 1st scenario aimed to highlight in situ detection, meanwhile 2nd scenario aimed the out-sensing. CmNA can manage both worlds in a transparent way.

CmNA layer can act in a decentralized fashion, can detect failures, and provides just truthful data to the DASS unit, ensuring the autonomy of the architecture. In this sense, further works will include design the ontologies and the general services deployed in the DASS unit for the appropriate deployment in ARMNANO. As well, next publications will consider to develop the specific components of CmNA (Communication Protocol Management System, Abnormal Situation Detection Mechanism, Data Authentication mechanism), in order to define and to test different alternative of design in each case.

References

[1] I. Akyildiz, J. Jornet and M. Pierobon, *Nanonetwoks: A New Frontier in Communications*, Communications of the ACM, vol. 54, pp. 84-89, 2011.

[2] A. Rae, *Real Life Applications of Nanotechnology in Electronics*, OnBoard Technology, vol. 1, pp. 36-39, 2005.

[3] I. Akyildiz, F. Brunetti, and C. Balsquez, *Nanonetworks: A New Communication Paradigm*, Computer Network, vol. 52, pp. 2260-2279, 2008.

[4] A. Lopez-Pacheco and J. Aguilar, *Autonomic Reflective Middleware for the Management of NANO Devices in a Smart Environment (ARMNANO)*, sent to publication, 2018.

[5] F. Moshir and S. Singh, *Modulation and Rate Adaptation Algorithms for Terahertz Channels*, Nano Communication Networks, vol. 1, pp. 1-38, 2016.

[6] I. Akyildiz, J. Jornet, and C. Han, *Terahertz Band: Next Frontier for Wireless Communication*, Physical Communication, vol. 12, pp. 16-32, 2014.

[7] K. Witrisal, G. Leus, G. Janssen, M. Pausini, F. Trosch, T. Zasowski, and J. Romme, *Nanocoherent Ultra-wideband Systems*, IEEE Signal Process. Mag., vol. 4, pp. 48-66, 2009.

[8] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, *Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions*, Fututre Generation Computer Systems, vol. 29, pp. 1645-1660, 2013.

[9] A. Akyildiz and J. Jornet, *The Internet of Nanothings*, European Conference on Wireless Sensor Networks, vol. 1, pp. 58-63, 2010.

[10] I. Akyildiz, M. Pierobon, S. Balasubramaniam, and Y. Koucheryavy, *The Internet of Bio-Nanothings*, IEEE Communications Magazine. Communications Standard Supplement, vol. 1, pp. 32-40, 2015.

[11] I. Akyildiz and J. Jornet, *The Internet of Multimedia Nanothings*, Nanocommunication Networks, vol. 3, pp. 242-251, 2012.

[12] A. Balasubramaniam and J. Kangasharju, *Realizing the Internet of Nanothings: Challenges, Solutions and Applications*, Computer, vol. 46, pp. 62-68, 2013.

[13] U. Chude-Okonkwo, R. Malekian, B. Maharaj, and A. Vasilakos, *Molecular Communication and Nanonetwork for Targeted Drug Delivery: A Survey*, IEEE Communications Surveys and Tutorials , vol. 19, pp. 3046-3096, 2017.

[14] A. Guney, B. Atakan, and O. Akan, *Mobile Ad hoc Nanonetworks with Collision based Molecular Communication*, IEEE Transactions Mobile Computing, vol. 11, pp. 353-366, 2012.

[15] P. Lio and S. Balasubramaniam, *Opportunistic Routing through Conjugation in Bacteria Communication Nanonetwork*, Nano Communication Networks, vol. 3, pp. 36-45, 2012.

[16] M. Barros, *Ca2+, based Molecular Communication Systems: Design and Future Research Directions*, Nano Communication Networks, vol. 11, pp. 103-113, 2017.

[17] M. Elkheir and N. Ali, *Internet of Nano-Things Healthcare Applications. Requirements, Opportunities and Challenges*, The First International Workshop on Advances in Body-Centric Wireless Communication and Networks and their Applications, vol. 1, pp. 9-14, 2015.

[18] J. Aguilar, M. Cerrada , G. Mousalli, F. Rivas, and F. Hidrobo, *A Multiagent Model for Intelligent Distributed Control Systems*, Lecture Notes in Artificial Intelligence, vol. 3681, pp. 191-197, 2005.

[19] V. Bravo, J. Aguilar, F. Rivas, M. Cerrada, Diseño de un Medio de Gestión de Servicios para Sistemas Multiagentes, in proceedings of the XXX Conferencia Latinoamericana de Informática, pp. 431-439, Arequipa, Perú, October, 2004.

[20] J. Vizcarrondo, J. Aguilar, E. Exposito, and A. Subias, ARMISCOM: Autonomic Reflective Middleware for Management Service Composition, in proceedings of the IEEE 2012 Global Information Infrastructure and Networking Symposium (GIIS), pp. 1-8, Choroni, Venezuela, December, 2012.

[21] J. Aguilar, M. Jerez, M. Mendoca, and M. Sanchez, *MiSCi: Autonomic Reflective Middleware for Smart Cities*, In Technologies and Innovation, Communication Conputing and Information Science Series, vol. 658, pp. 241-253, 2016.

[22] H. Wang and Q. Li, *Achieving Robust Message Authentication in Sensor Networks: A Public-key Based Approach*, Wireless Networks, vol. 16, no. 4, pp. 999-1009, 2009.

[23] J. Curtiss, *Acceptance Samples by Variables, with Special Reference to the Case in which Quality is Measured by Average or Dispersion*, Journal of Research of the National Bureau of Standards, vol. 39, pp. 271-290, 1947.

[24] H. Vogt, *Exploring Message Authentication in Sensor Networks*, European Workshop on Security in Ad-hoc and Sensor Networks, vol. 1, pp. 19-30, 2004.

[25] I. Mansour, D. Rusinek, G. Chalhoub, P. Lafourcade, and K. Bogdan, *Multihop Node Authentication Mechanisms for Wireless Sensor Networks*, in proceedings of the International Conference on Ad-Hoc Networks and Wireless, vol. 1, pp. 402-418, Messina, Italy, September, 2014.

[26] T. Wu, Y. Cui, B. Kusy, A. Ledeczi, J. Sallai, N. Skirvin, J. Werner, and Y. Xue, *A Fast and Efficient Source Authentication Solution for Broadcasting in Wireless Sensor Networks*, New Technologies, Mobility and Security, vol. 1, pp. 53-63, 2007.

[27] H. Wang, D. Lymberopoulos, and J. Liu, *Sensor Based-User Authentication*, in proceedings of the European Conference on Wireless Sensor Networks, vol. 1, pp. 168-185, Porto, Portugal, February, 2015.

[28] G. Kelly, *Body Temperature Variability (Part 1): A Review of the History of Body Temperature and its Variability due to Site Selection, Biological Rhythms, Fitness and Aging*, Alternative Medicine Review, vol. 11, pp. 278-293, 2006.

[29] K. Diller, *Heat Transfer in Health and Healing*, Journal of heat Transfer, vol. 10, pp. 138-166, 2015.

[30] E. Pinto, *Blood Pressure and Ageing*, Postgraduate Medicinal Journal, vol. 83, pp. 109-114, 2007.

[31] B. Friedman, S. Steinberg, W. Meggs, M. Kaliner, M. Frieri, and D. Metcalfe, *Analysis of Plasma Histamine Levels in Patients with Mast Cell Disorders*, The American Journal of Medicine, vol. 87, pp. 649-654, 1989.

[32] D. Hassabis, S. Legg, D. Wierstra, D. Kumaran, H. King, L. Antonoglou, A. Sadik, C. P. S. Beattie, G. Ostrovski, A. Fidjeland, M. Riedmiller, A. Graves, and M. Bellemare, *Human-Level Control through Deep Reinforcement Learning*, Nature, vol. 518, pp. 529-533, 2015.

[33] B. Liu, Y. Lai, and S. Ho, *High Spatial Resolution Photodetectors based on Nanoscale Tridimensional Nanostructures*, IEEE Photonics Technology Letters, vol. 10, pp. 929-931, 2010.

[34] R. Smith, A. Arca, X. Chen, L. Marques, M. Clark, J. Aylott, and M. Somekh, *Design and Fabrication of Ultrasonic Transducers with Nanoscale Dimensions*, Journal of Physics, vol. 1, pp. 012035, 2011.

Revista Venezolana de Computación
ISSN: 2244-7040
http://www.svc.net.ve/revecom
Vol. 5, No. 2, pp. 23-30, Diciembre 2018
Selección de los Mejores Artículos de CoNCISa 2018

Diseño e Implementación de una Aplicación Web para la Administración de Conferencias Académicas para Venezuela

Antonio Alarcon[1], Gabriel Espinel[1], Eric Gamess[2]
antonio.alarcon@outlook.com, gabriel.espinel.c@gmail.com, egamess@jsu.edu

[1] Escuela de Computación, Universidad Central de Venezuela, Caracas, Venezuela
[2] MCIS Department, Jacksonville State University, Jacksonville, AL, USA

Resumen: Este artículo presenta el diseño y la implementación de una aplicación web orientada a apoyar la administración de conferencias académicas en Venezuela, como lo son el Simposio Científico y Tecnológico en Computación (SCTC) y la Conferencia Nacional de Computación, Informática y Sistemas (CoNCISa). La aplicación automatiza varios procesos tediosos y repetitivos como la reservación de asistencia, la gestión de pagos, y la generación de documentos digitales (distintivos de identificación, certificados de asistencia y recibos de pago). El sistema fue realizado con software libre, usando herramientas modernas de desarrollo y de fácil acceso. Es de aclarar que tanto el SCTC como CoNCISa tienen requerimientos muy específicos y ningún software existente de administración de conferencias se pudo configurar para cubrir las necesidades de estas conferencias nacionales. Por ende, los organizadores de estos eventos científicos se vieron en la obligación de llevar a cabo los procesos de administración en forma manual, por años, resultando en un número significativo de horas trabajadas. Con este nuevo sistema, la administración del SCTC y de CoNCISa se ve mejorada significativamente, resultando en un significante ahorro de tiempo, en una reducción de costos al utilizar software de dominio público y al eliminar el uso de papel, y adicionalmente en una minimización de errores.

Palabras Clave: Aplicación Web; Desarrollo de Software; Conferencias Académicas; Sistemas de Administración de Conferencias; SCTC; CoNCISa.

Abstract: This paper presents the design and implementation of a web application aimed at supporting the management of academic conferences in Venezuela, such as the "Simposio Científico y Tecnológico en Computación" (SCTC) and the "Conferencia Nacional de Computación, Informática y Sistemas" (CoNCISa). The application automates several tedious and repetitive processes such as the reservation of attendance, the management of payments, and the generation of digital documents (identification badges, attendance certificates, and payment receipts). The system was made with free software, using modern development tools and that are easily available. It is worth mentioning that both the SCTC and CoNCISa have very specific requirements and no existing conference management software could be configured to meet the needs of these national conferences. Therefore, the organizing committees of these scientific events were forced to carry out the management processes manually, for years, resulting in a significant number of hours worked. With this new system, the management process of the SCTC and CoNCISa is significantly improved, resulting in a considerable saving of time, in a reduction of costs by using public domain software and eliminating the use of paper, as well as in a minimization of mistakes.

Keywords: Web Application; Software Development; Academic Conferences; Conference Management Systems; SCTC; CoNCISa.

I. Introducción

Las tecnologías de información y comunicación se han abierto camino en nuestra sociedad, permitiendo nuevas formas de colaboración, de automatización y optimización de procesos, de recopilación y compartición de información, e incluso nuevas formas de pensar y analizar la información para tomar mejores decisiones.

La automatización de procesos mediante computadores intenta dejar la intervención del hombre en segundo plano, con el fin de facilitar los procesos repetitivos, liberando así a las personas de funciones rutinarias, obteniendo ahorro significativo de tiempo, dinero y esfuerzo, reduciendo el margen de error y ofreciendo un esquema de trabajo más controlado.

Internet, específicamente la World Wide Web (WWW), dio un gran auge a las tecnologías de información, ya que, al mantener un gran número de nodos interconectados, hizo posible el

acceso a recursos remotos de una manera simple a través de la red. Este hecho hizo que las aplicaciones web hayan aumentado en popularidad de manera exponencial, logrando que muchas organizaciones prefieran apalancarse en estas tecnologías.

Desde hace algunos años, existen dos conferencias nacionales en computación muy renombradas: (1) el Simposio Científico y Tecnológico en Computación (SCTC) [1] y (2) la Conferencia Nacional de Computación, Informática y Sistemas (CoNCISa) [2]. El SCTC ha sido organizado por la Escuela de Computación de la Universidad Central de Venezuela desde el año 2006. CoNCISa existe desde el año 2013, y es uno de los eventos respaldado por la Sociedad Venezolana de Computación (SVC) [3]. Para ambos organizadores, estos eventos siempre se han caracterizados por un gran esfuerzo administrativo. Por ende, surgió la necesidad de automatizar la organización y administración de estos, basándose en las tecnologías de información y comunicación, para hacerlos más accesibles desde un punto de vista comunicacional, y más rentable desde un punto de vista organizacional.

El Simposio Científico y Tecnológico en Computación (SCTC) [1] es una conferencia académica que cubre las áreas afines a la computación y se lleva a cabo de manera bienal, con la finalidad de dar a conocer el trabajo investigativo de alta calidad que se lleva a cabo al nivel nacional e internacional. Adicionalmente, el SCTC promueve el encuentro de investigadores en computación y áreas afines, para permitir el intercambio de ideas, experiencias y conocimiento, que estimularán colaboraciones futuras y la creación de redes de conocimiento.

El SCTC [1] se ha caracterizado por una filosofía de gratuidad para el público, donde todas sus actividades se ofrecen sin costo alguno. Esta filosofía está motivada por el pensamiento de que es primordial que la transferencia del conocimiento sea libre y se haga sin cargo. Este beneficio no es sólo para el público en general que asiste a las presentaciones y a los tutoriales, sino que también se extiende a los autores de las contribuciones aceptadas donde no hay costo de registro.

CoNCISa [2] es uno de los principales eventos anuales de la Sociedad Venezolana de Computación (SVC) [3]. En ciertos años, esta conferencia cuenta con el respaldo académico del Centro Latinoamericano de Estudios en Informática (CLEI) [4] y tiene como finalidad consolidar el intercambio de experiencias investigativas, académicas, y tecnológicas, para impulsar el desarrollo del área de computación y crear lazos estrechos de cooperación a nivel nacional.

El propósito de este artículo es detallar el proceso de diseño y desarrollo de confVen, un sistema computacional para la administración de conferencias nacionales en los campos afines a la computación. En su actual versión, confVen cubre tanto las necesidades particulares del SCTC como las de CoNCISa. Fue utilizado en la organización de CoNCISa 2017 [2], automatizando con éxito la mayoría de los procesos que se realizaban de forma totalmente manual, ahorrando así un significante número de horas de trabajo tedioso.

El resto del presente documento está organizado como especificado a continuación. Los trabajos relacionados son revisados en la Sección II. En la Sección III, se presentan las funcionalidades del sistema. Las herramientas tecnológicas utilizadas en el desarrollo de la aplicación son descritas en la Sección IV. El desarrollo de la aplicación está detallado en la Sección V. Se presenta un balance del trabajo en la Sección VI a través de las conclusiones. Finalmente, se dan direcciones para trabajos futuros en la Sección VII.

II. Trabajos Relacionados

En la literatura especializada, se encuentran trabajos que comparan varios sistemas de administración de conferencias o "Conference Managment Systems" (CMSs). Por ejemplo, Jain, Tewari y Singh [5] compararon EDAS [6], Confious [7][8], OpenConf [9], ConfTool [10] y PaperDyne. Parra, Sendra, Ficarelli y Lloret [11] optaron por contrastar EasyChair [12][13], EDAS [6], OCS [14], START V2 [15], ConfTool [10], CMT [16] y CyberChair [17]. En su tesis de maestría, Groot [18] enfocó más su trabajo hacia la generación automática de memorias de conferencia (proceedings).

EasyChair [12][13] es una aplicación web para la sumisión y la evaluación de artículos de investigación, que es flexible, fácil de usar, y tiene muchas características que lo hacen un buen complemento para la administración de conferencias. Se ofrece bajo el modelo "Software as a Service" (SaaS) con características dependientes de la licencia escogida. Actualmente, EasyChair soporta tres licencias: (1) Free, (2) Professional y (3) Executive. En EasyChair, el proceso común de evaluación, soportado por las tres licencias, es el siguiente. Los autores someten sus contribuciones para evaluación. Una vez cerrado el llamado a trabajos, los revisores apuestan para especificar las contribuciones que quisieran revisar. Los presidentes del comité de programa hacen la asignación definitiva. Los revisores evalúan sus contribuciones asignadas y hacen recomendaciones sobre la aceptación. De acuerdo a las evaluaciones, los presidentes del comité de programa aceptan o no las contribuciones. Los autores de las contribuciones aceptadas someten la versión final que será publicada en las memorias de la conferencia (proceedings). Según la licencia escogida, EasyChair ofrece un soporte más amplio. En la versión Free, los autores son limitados a un único archivo de 20 MB como tamaño máximo. La versión Executive permite el sometimiento de varios archivos de hasta 100 MB cada uno. En lo que se trata de exportar la información en formato Excel o Comma-Separeted Value (CSV), está permitido solamente con las licencias Professional y Executive. Las tres licencias permiten la generación de las memorias de la conferencia, basada en la versión final de las contribuciones aceptadas. El manejo de salones es soportado con las licencias Professional y Executive, mientras que las multiconferencias (tener varios tracks en una conferencia grande como en el CLEI [4]) están disponibles únicamente en la licencia Executive. Ninguna de las licencias de EasyChair permite manejar tutoriales (donde hay un cupo máximo), hacer cobros, o generar documentos digitales como lo son los distintivos de identificación, los certificados de asistencia, o los recibos de pago. Por estas razones, EasyChair es usada para cubrir parcialmente la actividad relacionada con la organización de una conferencia.

HotCRP [19] es similar a EasyChair y permite que los autores sometan a evaluación sus trabajos, además de facilitar el proceso de revisión. HotCRP en muy bueno en cuanto se trata de navegar entre contribuciones, hacer búsqueda en éstas, y marcarlas para su clasificación. HotCRP es de código abierto y requiere que los organizadores de conferencias corran el

sistema en sus propios servidores. Al igual que EasyChair, no cubre la parte administrativa relacionada con tutoriales, cobros, emisión de distintivos de identificación y certificados de asistencia.

Open Conference Systems (OCS) [14] es una aplicación de código abierto que permite administrar conferencias y artículos a través de la web. Fue desarrollada por el proyecto Public Knowledge Project (PKP). Se distribuye de manera gratuita para ser usada en instalaciones locales. También ofrecen la aplicación bajo el modelo SaaS, alojada directamente en los servidores del proyecto. OCS permite: (1) la generación de un sitio web para la conferencia, (2) la composición y envío de CFPs (Call For Papers), (3) la recepción, evaluación, y aceptación de artículos, y (4) el registro de participantes.

ConfTool [10] es una herramienta multi-idioma basada en la web para la administración de eventos enfocados a conferencias académicas, talleres, congresos, y seminarios. Esta herramienta se presenta en dos versiones para suplir diferentes necesidades:

- VSIS ConfTool: Es la versión gratuita con funciones básicas o limitadas, orientada a eventos pequeños, no comerciales, como los eventos académicos. Esta se distribuye sin ningún soporte y para instalación local con el uso de un servidor web.
- ConfTool Pro: Viene con funcionalidades adicionales y sin limitaciones. Se ofrece bajo modalidad alojada directamente en los servidores de la compañía, bajo un modelo SaaS.

Independientemente de la versión, las características más destacadas de ConfTool son: (1) interfaz multi-idioma, (2) registro en línea de participantes, (3) foros en línea, (4) envío y revisión de artículos, (5) múltiples métodos de pago, e (6) integración con pasarelas de pago como PayPal.

OpenConf [9] es una aplicación web que ofrece tres ediciones diferentes para cubrir diferentes necesidades:

- OpenConf Community Edition: Esta edición ofrece de manera gratuita las funcionalidades más básicas: envío online, revisiones, aceptaciones, notificaciones, exportación de datos a formatos CSV, XML, Excel, SQL, entre otros. Esta edición se descarga libre de costo y se instala en los servidores propios de los organizadores de la conferencia, mientras no se generen o reciban pagos relacionados con la aplicación.
- OpenConf Plus Edition: Incluye todas las funcionalidades de la edición Community más otras adicionales como: soporte técnico, apuestas por parte de los revisores, subida de archivos de evaluación por parte de los revisores, memorias en línea, entre otros. Esta edición se ofrece bajo previa compra de licencia y se instala en los servidores propios de los organizadores de la conferencia. Se permite una única instalación. Eso es, cada instalación adicional requiere de la compra de una nueva licencia.
- OpenConf Professional Edition: Es la versión más avanzada y completa, donde se ofrecen docenas de módulos que extienden las funcionalidades. Además de incluir las funcionalidades de la edición Plus, la edición Professional tiene: alojamiento en servidores bajo modalidad SaaS, personalización de formularios, chequeo de plagio, aceptación de pagos por envíos de artículos, entre otros.

Como se pudo apreciar en esta revisión de los CMSs actuales, la mayoría de ellos son comerciales. Pocos ofrecen soporte para la generación de documentos digitales (distintivos de identificación, certificados de asistencia y recibos de pago). Más aún, ninguno soporta la administración de tutoriales que se desarrollan en forma paralela a la conferencia. Por estas razones, se decidió desarrollar confVen. Es de aclarar que en el 2015, se implementó un prototipo de sistema [20] para la administración de la membresía de los asociados a la Sociedad Venezolana de Computación (SVC). Además de renovar su membresía, este sistema permitía a los asociados registrar eventos independientes organizados por la SVC. No era específicamente orientado a la administración de conferencias tipo SCTC o CoNCISa.

III. Funcionalidades de la Aplicación

La implementación de las funcionalidades fue dividida en dos fases. teniendo como objetivo la entrega de la primera fase para CoNCISa 2017 (octubre 2017), y de la segunda fase para mayo 2018.

Durante la primera fase, se realizó el diseño general del sistema, que fue mejorándose iterativamente, siguiendo una metodología de desarrollo ágil (ver Sección V). Por restricciones de tiempo y disponibilidad de recursos, fue necesario priorizar aquellas funcionalidades necesarias para desarrollar un Producto Mínimo Viable (versión de un producto no completa, pero con suficientes características para satisfacer las necesidades de los primeros usuarios [21]) para una ejecución exitosa de una conferencia tipo CoNCISa. Una vez finalizada la primera fase, y después de realizar pruebas básicas de depuración, la aplicación fue utilizada exitosamente en un ambiente de producción en CoNCISa 2017.

En la segunda fase, se implementaron aquellas funcionalidades que no fueron incluidas en la primera fase porque se consideraron de menor prioridad. Adicionalmente, se agregaron todas las funcionalidades específicas para la administración de una conferencia tipo SCTC.

En esta sección, se describen cada uno de los módulos que conforman confVen, el sistema de administración de conferencias académicas.

A. Registro de Usuario

El sistema no tiene ningún módulo expuesto a usuarios anónimos o no autenticados. Todos los usuarios deben registrarse y autenticarse en el sistema para poder acceder a las funcionalidades de asistencia a conferencias o administración.

Durante el registro de una nueva cuenta de usuario, se solicita al usuario sus datos de cuenta como el nombre de usuario, correo electrónico, y contraseña, como se puede observar en la Figura 1.

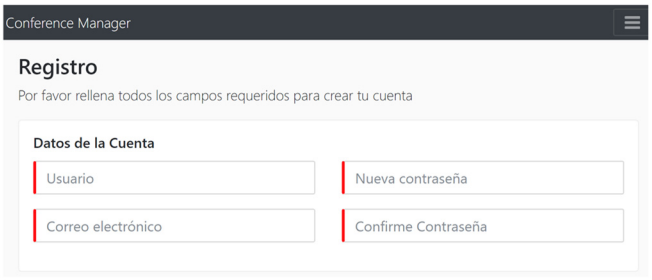

Figura 1: Datos de Cuenta durante el Registro de Usuario

De igual forma, se recolectan datos claves del usuario como sus ocupaciones y afiliaciones a instituciones académicas (ver Figura 2), datos de contacto, entre otros. Estos datos son esenciales para el módulo de reserva, ya que de estos depende los costos y descuentos a los cuales aplica el usuario cuando realiza una reserva a alguna conferencia.

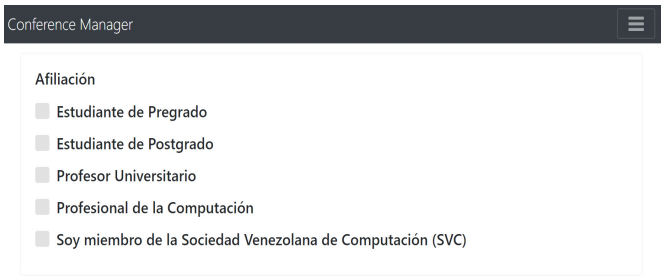

Figura 2: Datos de Afiliación durante el Registro de Usuario

B. Reservación de Asistencia a Conferencia

Una vez que el usuario esté registrado y que haya sido autenticado exitosamente a través del inicio de sesión, se le presentan todas las conferencias habilitadas para hacer reservas. El usuario tiene la opción de ingresar a cualquiera de ellas para realizar la reservación a la misma, seleccionando la modalidad de participación: "Autor" o "Asistente".

Dependiendo de las ocupaciones y afiliaciones del usuario, los costos asociados a eventos de la conferencia pueden variar. Si el usuario selecciona su asistencia como Autor, una lista adicional se le presenta para seleccionar el o los artículos que presentará en la conferencia, y el costo asociado a la selección.

Finalmente, la conferencia puede ofrecer eventos adicionales con capacidad limitada, como cursos cortos o tutoriales. Estos eventos son listados en la misma página de manera opcional para que el usuario seleccione si desea asistir a alguno de estos tutoriales, calculando los costos y disponibilidad de estos automáticamente como se aprecia en la Figura 3.

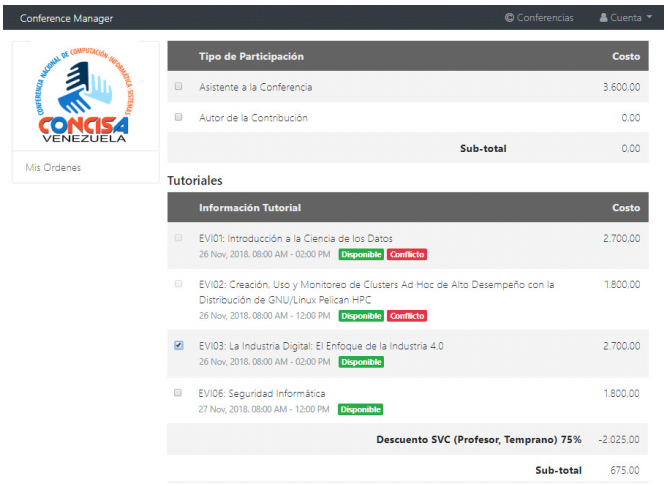

Figura 3: Listado de Tutoriales Ofrecidos para Reserva

C. Registro de Pago

Las conferencias en modalidad paga, como CoNCISa, esperan recibir los datos de pago de la reservación de los usuarios. Una vez creada la reserva por el usuario, esta entra en estado "Esperando Pago", que habilita un formulario de captación de los datos del pago.

En la Figura 4, se puede apreciar el formulario para tal fin, donde se recibe el pago realizado a través de depósito bancario o transferencia electrónica. En caso de que el usuario sea exonerado del pago, puede enviar la razón de su exoneración para su revisión por los organizadores de la conferencia.

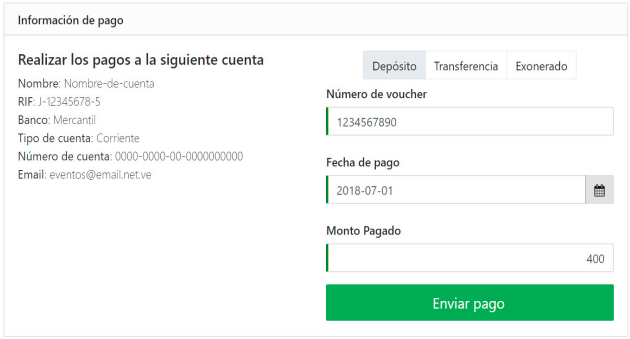

Figura 4: Formulario de Información de Pago

D. Administración de Reservaciones

Una vez autenticado por el sistema, un usuario con privilegios de administrador cuenta con un menú "Administración". Al seleccionar la opción de administrar reservas, este usuario podrá visualizar toda la información de las reservas de cada conferencia. La Figura 5 muestra la interfaz del sistema con la que interactúa un usuario administrador.

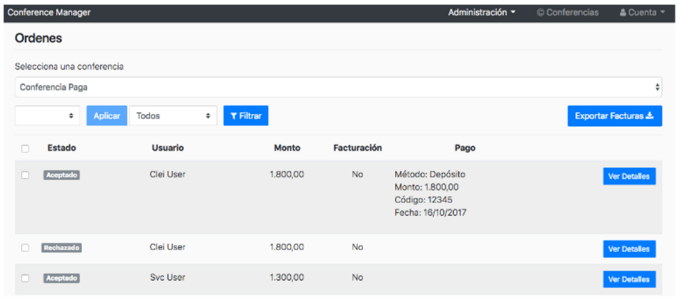

Figura 5: Listado de Reservaciones de una Conferencia

Un usuario con privilegios de administrador puede filtrar las reservas de acuerdo a su estado. También, tiene la opción de exportar aquellas reservas que requieren factura legal a un archivo con formato CSV, para que sean procesadas adecuadamente.

Cada reserva tiene la opción "Ver Detalles" para visualizar información más amplia de esta, como por ejemplo los tutoriales seleccionados durante la reserva, y la opción de descargar el documento digital del recibo de pago en formato PDF. De igual forma, existe la opción de enviar el recibo de pago a través de correo electrónico, con el fin de ahorrar papel.

Esta funcionalidad permite gestionar el estado de las reservas, actualizando el estado de una o varias para aceptarlas o rechazarlas, según el criterio de un usuario administrador. El cambio de estado de las reservas es notificado al usuario a través de un correo electrónico, para que este pueda finalizar su petición.

E. Administración de Distintivos de Identificación

Esta opción permite a un usuario administrador gestionar la generación automática de distintivos para los asistentes a una conferencia. Puede filtrar el listado de reservaciones de acuerdo a los tutoriales seleccionados, como también filtrar por "Asistentes" o "Autores" de la conferencia.

Los distintivos son generados al accionar el botón "Generar Distintivos", cuando haya uno o más ítems seleccionado en la lista. Esta acción inicia la descarga de un archivo PDF con todos los distintivos seleccionados para la conferencia. La Figura 6 muestra un ejemplo de un distintivo generado automáticamente por la aplicación, de un cursante de un tutorial.

Figura 6: Ejemplo de Distintivo de Cursante de Tutorial

F. Administración de Certificados de Asistencia

Esta funcionalidad tiene una interfaz de usuario similar a la de la Sección III.E. Presenta una lista de las reservaciones confirmadas a un usuario administrador para seleccionar aquellas a las que se desea generar el certificado de asistencia, como se aprecia en la Figura 7.

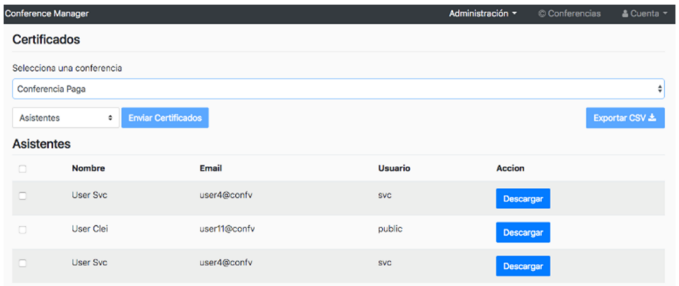

Figura 7: Listado de Asistentes para Generar Certificados

Se puede filtrar el listado de acuerdo al tipo de asistencia de la reservación ("Asistente" o "Autor").

En esta interfaz, un usuario administrador puede descargar los certificados de asistencia en formato PDF de manera individual, o puede seleccionar múltiples ítems de la lista para el envío del certificado por correo electrónico. Adicionalmente, un usuario administrador puede generar un listado con los ítems seleccionados y descargarlos en un archivo con formato CSV, que puede ser usado durante la conferencia para tomar asistencia. La Figura 8 muestra un ejemplo de un certificado de asistencia generado por esta funcionalidad.

Figura 8: Ejemplo de Certificado de Asistencia a un Tutorial

G. Administración de Conferencias

Un usuario con privilegios de administrador tiene la posibilidad de crear y editar conferencias. Cuando se crea una nueva conferencia, se debe llenar un formulario con la información de la misma, como se puede observar en la Figura 9.

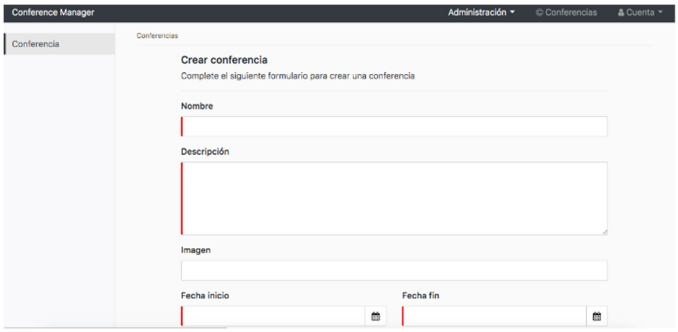

Figura 9: Formulario de Creación de Conferencia

Es importante que las conferencias tengan el nombre y la descripción clara, para que los usuarios puedan identificarlas con facilidad y no genere conflicto o confusión con alguna otra conferencia del sistema.

Una de las opciones de configuración más importantes de la creación de una conferencia es la modalidad de la conferencia. Esta define si la conferencia es "gratuita" (como el SCTC) o "paga" (como CoNCISa). Dependiendo de esta modalidad, si habilitan o no otras opciones o flujos del sistema.

Al crear una conferencia en modalidad paga, el sistema habilita una opción adicional para gestionar los costos de la conferencia, como se puede ver en la Figura 10.

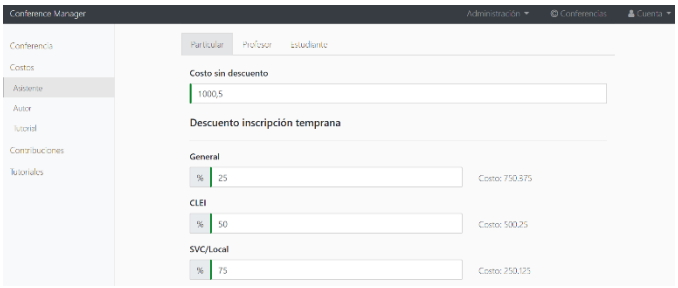

Figura 10: Formulario de Edición de Conferencia Paga

En la opción de "Costos", un usuario con privilegios de administración puede modificar el costo de la conferencia y de los tutoriales para los "Asistentes" y "Autores". Asimismo, el un usuario administrador puede definir los descuentos a los que aplican los usuarios por categorías, ocupaciones y afiliaciones.

IV. Herramientas Tecnológicas

El desarrollo de la aplicación se realizó completamente con software libre. La aplicación fue desplegada con éxito en servidores con sistema operativo GNU Linux, pero puede ser desplegada en cualquier sistema operativo diferente que soporte Java 8 sin mayor dificultad.

Para el desarrollo del Front-End, se usó el framework de desarrollo Angular bajo el lenguaje de programación TypeScript [22]. Typescript es un lenguaje de programación de código abierto, basado en JavaScript, desarrollado y mantenido por Microsoft con la finalidad de facilitar la creación de aplicaciones JavaScript a gran escala.

Angular [23] es un framework de desarrollo Front-End que permite la creación de Single Page Application de una manera organizada, siguiendo el patrón de diseño Modelo-Vista-Controlador. Sobre Angular, se usan varios componentes de código abierto disponibles en la comunidad, gestionados por el gestor de paquetes NPM (Node Package Manager) [24].

Del lado del Back-End, los servicios web fueron desarrollados en el lenguaje Java, utilizando el framework Spring [25], más específicamente Spring MVC y Spring Boot. Además, se usó Hibernate [26] como framework ORM para interactuar con el repositorio de datos. Todas las dependencias del Back-End y el ciclo de vida de construcción de la aplicación fue gestionado por Maven [27].

Como sistema manejador de bases de datos se utilizó H2 [28], por su simplicidad en relación a su administración y despliegue. Este manejador ofrece su propia interfaz de administración, que puede ser accedida a través de un navegador web.

V. Arquitectura y Desarrollo de la Aplicación

El desarrollo de la aplicación web inició con la captación de los requerimientos de alto nivel para poder tener una visión clara de los objetivos de la aplicación, orientados a satisfacer las necesidades del usuario. Una vez obtenidos los requerimientos del usuario, se hizo un análisis general para evidenciar los elementos fundamentales que servirían de guía y apoyo para la implementación de las funcionalidades. Uno de estos elementos resultantes del análisis fue la arquitectura general de la aplicación. Esta es representada en la Figura 11, usando el modelo de arquitectura de software C4 [29] para representar los actores, las interacciones de la aplicación al nivel de sus componentes internos, y las interacciones de sus componentes internos con sistemas externos.

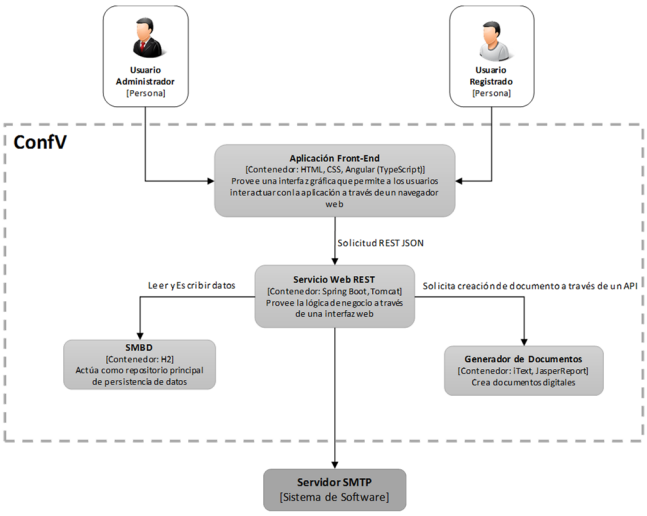

Figura 11: Diagrama de Contenedores del Sistema

La arquitectura de la aplicación tiene claramente definida los componentes del sistema responsables de la interacción con el usuario, desde el lado del cliente, como es el Front-End de la aplicación. Este componente se comunica con el Back-End a través de solicitudes HTTP hacia un servicio web, que se encarga del procesamiento y la persistencia de los datos, haciendo uso del componente SMBD para delegar dicha tarea. La interacción entre la aplicación web y el componente de generación de documentos digitales está bien definida por el uso de Interfaces de Programas de Aplicación (API, por sus siglas en inglés) dentro de la aplicación. Finalmente, la

arquitectura muestra la interacción con el sistema externo, en este caso, el servidor SMTP usado para el envío de los correos electrónicos generados por la aplicación.

Otro de los elementos claves generados en el análisis de los requerimientos fue el modelo de datos. Este modelo se creó a un alto nivel para servir como base para la implementación de las funcionalidades. Su definición fue revisada iterativamente a medida que se avanzaba en el desarrollo de la aplicación, para ajustarse a los requerimientos específicos de cada funcionalidad.

La metodología de desarrollo usada para el proyecto fue AgilUs [30]. Esta metodología ágil, creada en el Centro de Ingeniería de Software y Sistemas (ISYS) de la Escuela de Computación de la Universidad Central de Venezuela, se basa principalmente en el concepto de usabilidad y pone mayor peso en la construcción de interfaces de usuario, siendo estas, las que guíen las decisiones de la ingeniería de software del proyecto. AgilUs presenta un ciclo de vida bien definido: Requisitos, Análisis, Prototipaje, y Entrega. Este ciclo de vida es marcado por la usabilidad de la aplicación.

De acuerdo al ciclo de vida definido por la metodología AgilUs, los requerimientos fueron implementados de manera iterativa e incremental, recopilando los detalles particulares de cada funcionalidad, para luego ser analizados, estudiando el impacto sobre el modelo de datos existente e interfaces de usuario.

La etapa de prototipaje bosqueja la visión de la funcionalidad desde la perspectiva del usuario, orientada a la usabilidad. Este prototipo es el que guía la implementación, tanto del lado del Front-End con el diseño de la interfaz de usuario, como del lado del Back-End con su interacción con los distintos componentes y servicios.

Finalmente, la etapa de entrega verifica el correcto comportamiento de la funcionalidad y cumple con los principios de usabilidad. La aplicación ha sido probada y depurada, en su mayoría, usando un proceso manual y con el feed-back recibido de los organizadores de CoNCISa 2017, donde se usó plenamente. Sin embargo, para validar las funcionalidades críticas de la aplicación como el cálculo de los costos de las reservas, se crearon pruebas automatizadas.

VI. Conclusiones

Las conferencias o eventos académicos se presentan como uno de los métodos de distribución de conocimientos más efectivos y usados a nivel mundial. La organización de estos eventos puede consumir mucho tiempo en tareas repetitivas que pueden ser automatizadas con las tecnologías de información y comunicación existentes en la actualidad.

En este artículo, se ha presentado una aplicación para la administración de conferencias académicas como el Simposio Científico y Tecnológico en Computación (SCTC) [1] y la Conferencia Nacional de Computación, Informática y Sistemas (CoNCISa) [2]. Se automatizaron procesos de administración de conferencias como el registro de usuarios, las reservaciones de las asistencias (a la conferencia como a los tutoriales), y la captación de los datos de pagos. Asimismo, procesos que consumían una alta cantidad de tiempo como la generación de distintivos de identificación, certificados de asistencia, recibos de pago y lista de facturas a preparar fueron automatizados, disminuyendo en gran parte la carga de administración de este tipo de eventos académicos. Aún más, el uso de papel fue completamente eliminado, ya que todos los documentos generados son digitales y enviados por correo electrónico, resultando en un ahorro monetario significativo para los organizadores.

Es de aclarar que la aplicación no fue concebida para cubrir todos los procesos de una conferencia. Por ejemplo, la gestión de los trabajos de investigación no está contemplada en confVen, sino que se utiliza paralelamente a un software de envío y revisión de trabajos como EasyChair, EDAS, o HotCRP.

La aplicación está basada en el uso de componentes y plataformas de software libre, como: Angular con TypeScript en el Front-End, y Java, Spring, Hibernate en el Back-End. Todas estas herramientas agilizaron el desarrollo de la aplicación por la alta disponibilidad de información referente a su uso. Además, su naturaleza de software libre permitió mantener los costos de desarrollo del producto al mínimo.

Esta aplicación tuvo una salida a producción exitosa durante CoNCISa 2017 que se organizó en la Universidad Católica Andrés Bello de Ciudad Guayana, en octubre del 2017, y se estará usando también para CoNCISa 2018, que se celebrará en la Universidad de Los Andes de Mérida, en noviembre del 2018.

VII. Trabajos Futuros

Si bien la aplicación desarrollada cumple con los objetivos planteados, se puede pensar en una serie de extensiones que podrían mejorar su uso. Por ejemplo, los organizadores de conferencias nacionales se podrían beneficiar de nuevos módulos para la administración de recursos como salones o salas de conferencias. Adicionalmente, se podría automatizar la generación de reportes de participación, resumen de pagos y de montos recibidos y facturados por tipo de evento y categoría de participantes.

En lo que se trata de la interacción con los usuarios administradores, se considera que se puede hacer uso de componentes Front-End que aumenten las capacidades de la aplicación, facilitando la interacción al usar tablas interactivas que admitan paginación y filtros para todas las columnas.

Finalmente, se podría implementar la autenticación de usuarios contra sistemas externos como Google o Facebook. Una integración con estos populares sistemas facilitaría el uso de la aplicación al nivel de todos los usuarios, ya que no tendrían que crear una cuenta y contraseña exclusiva para la aplicación, sino que serían capaces de reusar las cuentas existentes de manera segura.

Agradecimientos

Queremos agradecer a Rosseline Rodríguez de la Universidad Simón Bolívar por todo el apoyo que prestó durante el desarrollo de esta aplicación.

Referencias

[1] SCTC, *Simposio Científico y Tecnológico en Computación*, http://www.sctc.org.ve.

[2] CoNCISa, *Conferencia Nacional de Computación, Informática y Sistemas*, http://www.concisa.net.ve.

[3] SVC, *Sociedad Venezolana de Computación*, http://www.svc.net.ve.

[4] CLEI, *Centro Latinoamericano de Estudios en Informática*, http://www.clei.org.

[5] M. Jain, T. Tewari, and S. Singh, *Survey of Conference Management Systems*, International Journal of Computer Applications, vol. 2, no. 2, pp. 14-20, May 2010.

[6] EDAS, *EDAS: Editor's Assistant*, https://edas.info/doc.

[7] M. Papagelis, D. Plexousakis, and P. N. Nikolaou, *Confious: Managing the Electronic Submission and Reviewing Process of Scientific Conferences*, in Proceedings of the 6th Intenational Conference on Web Information System Engineering, New York City, NY, USA, November 2005.

[8] Confious, *Introducing Confious - The Conference Nous*, http://www.confious.com.

[9] OpenConf, *Peer-Review, Abstract and Conference Management*, https://www.openconf.com.

[10] ConfTool, *ConfTool: Conference Management Software*, http://www.conftool.net.

[11] L. Parra, S. Sendra, S. Ficarelli, and J. Lloret, *Comparison of Online Platforms for the Review Process of Conference Papers*, in Proceedings of the Fifth International Conference on Creative Content Technologies (CONTENT 2013), Valencia, Spain, June 2013.

[12] EasyChair, *EasyChair Home Page*, http://www.easychair.org.

[13] A. Voronkov, *The Design of EasyChair*, in Proceedings of the 12th International Conference on Distributed Computing and Internet Technology (ICDCIT 2016), Bhubaneswar, India, January 2016.

[14] Public Knowledge Project, *Open Conference Systems (OCS)*, https://pkp.sfu.ca/ocs.

[15] *START V2 Conference Manager*, http://www.softconf.com.

[16] Microsoft, *Conference Management Toolkit (CMT)*, https://cmt3.research.microsoft.com/Content/CMT.html.

[17] CyberChair, *A Free Web-based Paper Submission and Reviewing System with PC Meeting and Proceedings Preparation Support*, http://www.borbala.com/cyberchair.

[18] J. de Groot, *Document Understanding for Automatic Proceedings Generation*, Master Thesis, University of Groningen, Groningen, The Netherlands, August 2013.

[19] HotCRP, https://hotcrp.com.

[20] R. Monascal, R. Rodríguez y L. Tineo, *Sistema de Trámites y Comunidad de la SVC*, en las Memorias de la Tercera Conferencia Nacional de Computación, Informática y Sistemas (CoNCISa 2015), Valencia, Venezuela, Octubre 2015.

[21] E. Maidana, *¿Qué es un Producto Mínimo Viable y Cómo lo Puedes Desarrollar?* https://www.puromarketing.com/13/19295/producto-minimo-viable-como-puedes-desarrollar.html.

[22] R. Jensen, *Learning TypeScript*, Packt Publishing, October 2015.

[23] A. Chandermani, *AngularJS by Example*, Packt Publishing, March 2015.

[24] npm, *What is npm?*, https://docs.npmjs.com/getting-started/what-is-npm.

[25] Spring, *Spring Framework*, https://spring.io.

[26] C. Bauer, G. King, and G. Gregory, *Java Persistence with Hibernate*, Manning Publications, Second Edition, November 2015.

[27] Apache Maven Project, *Welcome to Apache Maven Project*, https://maven.apache.org.

[28] H2, *H2 Database*, http://www.h2database.com/html/main.html.

[29] S. Brown, *The C4 Model for Software Architecture*, https://c4model.com.

[30] A. E. Acosta, *AgilUs: Construcción Ágil de la Usabilidad*, Caracas, Venezuela, 2011.

Revista Venezolana de Computación
ISSN: 2244-7040
http://www.svc.net.ve/revecom
Vol. 5, No. 2, pp. 31-40, Diciembre 2018
Selección de los Mejores Artículos de CoNCISa 2018

Extensión UML para Clustering Difuso en Data Warehouse

Livia Borjas[1], Rosseline Rodríguez[2], Betzaida Romero[2]
livacaro7@gmail.com, crodrig@usb.ve, betzaidaromero@usb.ve

[1] Instituto Universitario de Tecnología Dr. Federico Rivero Palacios, Caracas, Venezuela
[2] Departamento de Computación, Universidad Simón Bolívar, Caracas, Venezuela

Resumen: La Minería de Datos (MD) aplica métodos que generan modelos inteligibles desde grandes volúmenes de datos, usando técnicas de análisis introspectivo para descubrir patrones y relaciones ocultos. Ésta es una fase importante del proceso conocido como Descubrimiento de Conocimiento en Bases de Datos (KDD). *Clustering* es una técnica de aprendizaje no supervisado, ampliamente utilizada para encontrar "comportamientos" en una larga colección de datos, popularmente usada en KDD. Esta técnica ha sido mejorada aplicando conjuntos difusos, surgiendo los algoritmos de *Clustering* Difuso que permiten descubrir "*clusters*" que se solapan en la frontera. El problema con la aplicación de estas técnicas es que se hace en niveles bajos de abstracción en donde la información es compleja. Sería ideal modelar el proceso de extracción de conocimiento desde niveles altos de abstracción donde los datos son sencillos, inteligible y cuyos modelos no dependen de las herramientas subyacentes para su implementación. Además, se pueden aprovechar los beneficios que ofrecen los modelos multidimensionales que facilitan el modelado del KDD disminuyendo su complejidad de las fases de recopilación e integración de los datos. En el presente artículo se propone una extensión por medio de perfiles UML para la modelación de Minería de Datos, basada en *Clustering* Difuso.

Palabras Clave: Minería de Datos, Clustering Difuso, Data Warehouses, UML, KDD.

Abstract: Data Mining (MD) applies methods that generate intelligible models from large volumes of data, using introspective analysis techniques to discover hidden patterns and relationships. This is an important phase of the process known as Knowledge Discovery in Databases (KDD). Clustering is an unsupervised learning technique, widely used to find "behaviors" in a long data collection, popularly used in KDD. This technique has been improved by applying fuzzy sets, resulting in Fuzzy Clustering algorithms that allow discovering "clusters" that overlap at the border. The problem with the application of these techniques is that it is done at low levels of abstraction where the information is complex. It would be ideal to model the process of extracting knowledge from high levels of abstraction where the data is simple, intelligible and whose models do not depend on the underlying tools for its implementation. In addition, you can take advantage of the benefits offered by the multidimensional models that facilitate the modeling of the KDD, reducing the complexity of the phases of data collection and integration. In the present article, an extension is proposed by means of UML profiles for the modeling of Data Mining, based on Fuzzy Clustering.

Keywords: Data Mining, Fuzzy Clustering, Data Warehouses, UML, KDD.

I. Introducción

Como resultado de la automatización de procesos a toda escala y de los logros alcanzados en tecnologías de información y de almacenamiento de datos, en los últimos años ha aumentado el uso de bases de datos de gran volumen. El análisis de estos extensos volúmenes de datos tiene un gran valor agregado para las organizaciones, brindando conocimiento nuevo de interés para la toma de decisiones estratégicas propias de funciones complejas como la planificación y la predicción, en donde los sistemas de bases de datos tradicionales son insuficientes.

El procesamiento automático de grandes volúmenes de datos a fin de encontrar conocimiento útil para un usuario, es el objetivo principal del proceso de Descubrimiento de Conocimiento en Bases de Datos (*Knowledge Discovery in Databases* o KDD), el cual identifica "patrones comprensibles que se encuentran ocultos en los datos" [1].

Las fases del proceso de KDD son iterativas e incluyen: (1) Recopilación e Integración de los datos en una tabla llamada Atributo-Valor; (2) Selección, Limpieza y Transformación de los Datos para construir la Vista Minable; (3) Minería de Datos (Extracción de Conocimiento) que genera Modelos; (4) Interpretación o Evaluación, para llegar finalmente al producto o Conocimiento; y la fase de (5) Difusión y uso (Divulgación) que lleva a las decisiones estratégicas [2]. Uno de los caminos a seguir en el proceso KDD es implementar un Almacén de Datos (AD), o *Data Warehouse* (DW) durante el pre-procesamiento de los datos. El DW es un repositorio de datos

históricos coleccionado de diversas fuentes bajo un esquema unificado e integrado, frecuentemente modelado en forma multidimensional. En la Figura 1, se observa el esquema del KDD integrado con DW, donde la tabla Atributo-Valor es sustituida por el modelo multidimensional del almacén de datos.

Fayyad [3] describe las fases del KDD integradas con un DW:

1) Recopilación e Integración de los Datos en un DW, que incluye la determinación de fuentes de información que pueden ser útiles y su ubicación; el diseño del esquema DW que unifique de manera operativa toda la información recogida y la implantación del DW que permita la navegación y visualización de sus datos, para discernir aspectos a ser estudiados.

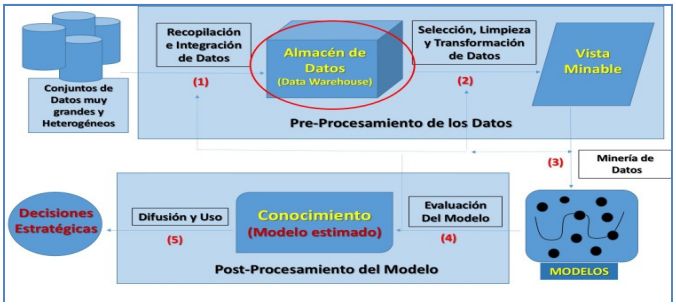

Figura 1: Proceso de KDD Integrado con un DW

1) Recopilación e Integración de los Datos en un DW, que incluye la determinación de fuentes de información que pueden ser útiles y su ubicación; el diseño del esquema DW que unifique de manera operativa toda la información recogida y la implantación del DW que permita la navegación y visualización de sus datos, para discernir aspectos a ser estudiados.

2) Selección, limpieza y transformación de los datos que se van a analizar. Considerando la información disponible relacionada con el dominio de los datos, en esta fase se corrigen o eliminan los datos incorrectos, inconsistentes, ausentes e incompletos. De igual manera se seleccionan los atributos relevantes para el estudio.

3) Minería de Datos (MD). En esta fase se identifica la tarea de MD a realizar, así como el método o la técnica más apropiada para alcanzar los objetivos de análisis planteados.

4) Evaluación del modelo, que incluye la interpretación, transformación y representación de los patrones extraídos; se evalúan los modelos descubiertos con los expertos del dominio del problema y se resuelven posibles inconsistencias o conflictos con el conocimiento disponible.

5) Divulgación y uso del nuevo conocimiento a todos los usuarios, de manera tal que se puedan realizar decisiones estratégicas en la organización interesada en el estudio.

La implantación de un DW como paso previo a la Minería de Datos dentro del KDD resulta útil por dos razones: los pasos de procesamiento e integración de datos para producir el DW genera un repositorio que facilita los objetivos del análisis y el desarrollo del modelo multidimensional del DW favorece la labor de conceptualización de los fenómenos del universo en observación. Un ejemplo de aplicación puede observarse en [4]. Los problemas que se pudieran encontrar durante el proceso KDD son:

- Visualizar el KDD como operaciones simples aisladas en lugar de un proceso integrado lo que produce duplicidad de tiempo y desperdicio de recursos. Por ejemplo, el pre-procesamiento de datos que comparten tanto el KDD como la Minería de Datos.
- Cuando los datos a ser analizados durante la Minería de Datos se encuentran almacenados en archivos planos, sobre todo si se utilizan en un modelo previo al KDD.
- Carencia de mecanismos de modelación del mundo real donde se aplique técnicas de MD.
- Perdida de oportunidad de nuevos conocimientos en cada paso del KDD.

El proceso de modelamiento de almacenes de datos como modelos multidimensionales ha arrojado avances [5][6][7][8] aprovechables en técnicas de minería de datos descriptiva y difusa ampliamente utilizada por su poder expresivo conocida como *Clustering* Difuso. A pesar que existen esfuerzos de propuestas de mecanismos y metodologías que permiten la especificación, modelación o implementación de requisitos difusos [9][10][11][12], no existen propuestas para el modelado de requisitos difusos que involucren técnicas de MD para el tratamiento de grandes volúmenes de datos en espacios multidimensionales.

La disposición de un mecanismo de modelaje de *clustering* difuso en grandes volúmenes de datos que use espacios multidimensionales, contribuiría con el objetivo del modelado conceptual [13]: "captar y enumerar exhaustivamente los requisitos y el dominio de conocimiento, de forma que todos los implicados puedan entenderlos y estar de acuerdo con ellos". El presente trabajo plantea una extensión del mecanismo propuesto por Zubcoff et al. [14] del proceso de KDD en espacio multidimensional, para la modelación de *clustering* difuso como técnica de Minería de Datos, con el fin de resolver los problemas mencionados encontrados en este proceso.

El resto del documento está estructurado de la siguiente forma: la Sección II describe el marco teórico que sustentan este trabajo. En la Sección III, se presenta la propuesta de un perfil UML para diseñar modelos de *clustering* difusos sobre espacios multidimensionales. La Sección IV muestra su aplicación a un caso de estudio. Finalmente, la Sección V presenta las conclusiones y trabajos futuros.

II. Marco Teórico

Para facilitar la comprensión de esta propuesta, se presentan las bases teóricas relacionadas con los mecanismos de extensión de UML, así como, los componentes fundamentales de la arquitectura de Minería de Datos con *Clustering* Difuso en espacios Multidimensionales.

A. Lenguaje Unificado de Modelación (UML)

Un modelo es una representación que describe un sistema o parte de él en un lenguaje con una sintaxis y semántica precisa, que puede ser interpretado y manipulado por un ordenador, de manera que pueda ser comprendido por diferentes diseñadores, independientemente de su implementación. Un artefacto es un

modelo o pieza de información producido en el proceso de desarrollo de software.

UML [15] es un lenguaje gráfico de propósito general, altamente flexible y expresivo, definido para la modelación de sistemas, de amplio uso por los arquitectos de software, cuyo propósito es especificar, construir y documentar los componentes de estos sistemas. Cuando UML resulta insuficiente para modelar dominios muy específicos se restringe o especializa los constructores propios de dicho lenguaje, como son: clases, asociaciones, atributos, operaciones, transiciones, entre otros. Además, UML incluye un mecanismo de extensión que permite definir lenguajes de modelación que son derivados de él [16]. El paquete *Profiles* de UML 2.0 provee mecanismos para extender y adaptar las metaclases de un metamodelo cualquiera a las necesidades concretas de una plataforma o de un dominio de aplicación.

Los perfiles UML están basados en estereotipos, restricciones y valores etiquetados adicionales que son aplicados a los elementos o relaciones de un diagrama. Un perfil se define en un paquete UML, estereotipado «profile», que extiende a un metamodelo o a otro perfil. Para definir perfiles se utilizan tres mecanismos: estereotipos (*stereotypes*), restricciones (constraints), y valores etiquetados (*tagged values*).

Un estereotipo está definido por un nombre y por una serie de elementos del metamodelo sobre los que puede asociarse. Gráficamente, los estereotipos se definen dentro de cajas etiquetadas «stereotype», a las cuales es posible asociarles restricciones, usando el lenguaje OCL o lenguaje natural, que imponen condiciones sobre los elementos del metamodelo. Un valor etiquetado es un metaatributo adicional que se asocia a una metaclase del metamodelo extendido por un perfil. Todo valor etiquetado ha de contar con un nombre y un tipo, y se asocia a un determinado estereotipo.

OCL (*Object Constraint Language*) es un lenguaje formal propuesto por OMG [17], usado para describir expresiones sobre UML que modelan condiciones invariantes que el sistema debe cumplir, así como para modelar pre y post condiciones y consultas sobre los objetos del modelo. Estas restricciones OCL pueden ser omitidas dentro del modelo gráfico, sin embargo son muy útiles en aquellos casos donde el modelo no es suficientemente expresivo y/o se quiere evitar estados indeseables del sistema. Las expresiones OCL son de la forma context TypeName inv Expression, en donde: *context* e *inv* son palabras reservadas del lenguaje; TypeName el nombre de la clase que representa el contexto y Expression la restricción cuyo resultado es un valor booleano. La declaración del contexto es opcional.

Según Fuentes y Vallecillo [16] definir un perfil UML incluye las siguientes consideraciones:

1) Definición del metamodelo de la plataforma o dominio de aplicación a modelar.

2) Definición del perfil dentro del paquete «profile» incluyendo un estereotipo por cada uno de los elementos del metamodelo. Estos estereotipos tendrían el mismo nombre que los elementos del metamodelo, a fin de establecer la relación entre el metamodelo y el perfil.

3) Aplicación de cada estereotipo a la metaclase de UML que se utilizó en el metamodelo del dominio para definir un concepto o una relación.

4) Los elementos del perfil serán los atributos del metamodelo, definidos como valores etiquetados, incluyendo la definición de sus tipos y sus posibles valores iniciales.

5) Las restricciones del dominio serán las restricciones que forman parte del perfil.

En este trabajo se propone extender UML con un perfil que permite representar requisitos difusos para clustering difuso en espacios multidimensionales. El perfil propuesto se basa en estereotipos y en el uso del lenguaje OCL [17] para la especificación formal de tales requisitos.

B. Modelo de Dominio de Clustering Multidimensional

En la presente investigación se reutiliza el perfil UML para el modelo multidimensional de un DW propuesto por Lujan-Mora et al. [18]. Para comprender el perfil UML es necesario comprender primero el modelo multidimensional usado para modelar almacenes de datos así como la técnica de agrupamiento o *clustering* de MD, los cuales se describen a continuación.

Los Almacenes de Datos (*Data Warehouse* o DW) se modelan como espacios multidimensionales donde los datos se organizan en hechos y dimensiones. Los hechos representan colecciones de medidas en forma de datos numéricos, de tal manera que tienen dimensiones asociadas que representan descripciones (datos textuales) que ofrecen un contexto al análisis, formando jerarquías, cuyas medidas pueden ser agregadas a distintos niveles de granularidad [14]. En la Figura 2 [14], se muestra un modelo multidimensional que representa la cantidad de productos adquiridos en una organización, a través cuatro dimensiones: Tiempo, Producto, Cliente y Causa. Cada una de estas dimensiones tiene distintas granularidades o niveles de agregación, como por ejemplo, Fecha, Mes, Año y Todo en la dimensión Tiempo.

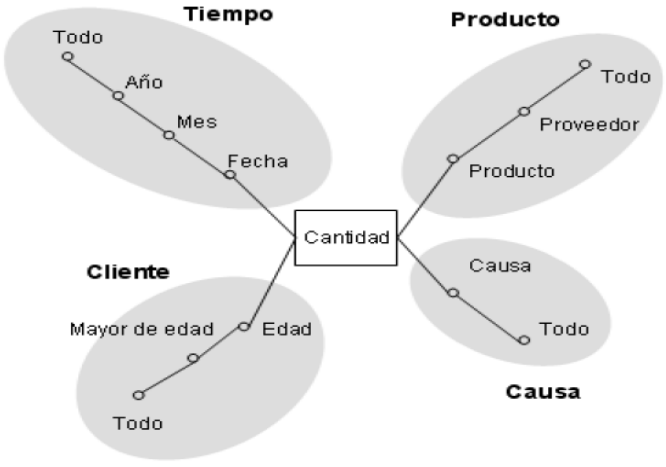

Figura 2: Modelo Multidimensional

Por otro lado, la Minería de Datos (*Data Mining* o DM) es "un mecanismo de explotación, consistente en la búsqueda de información valiosa en grandes volúmenes de datos, con el fin de descubrir patrones, relaciones, reglas, asociaciones o incluso excepciones útiles para la toma de decisiones" [1]. Las técnicas DM se dividen en dos categorías [19]: predictivas y

descriptivas, según su funcionalidad. Las predictivas o supervisadas predicen el valor de un atributo (etiqueta) de un conjunto de datos a partir de datos previamente conocidos, entre ellas están: la clasificación, la regresión y la predicción. Las técnicas descriptivas o no supervisadas descubren patrones y tendencias en los datos, entre las cuales se tienen, el clustering, la asociación y la correlación y dependencia. Dado que el perfil UML aquí presentado se basa en el clustering se dará más detalle de esta técnica.

El *clustering* se basa en la división de los datos en grupos de objetos llamados *clusters* [2]. Consiste en agrupar una colección dada de patrones no etiquetados con el fin de detectar grupos de individuos. También se le denomina clasificación no supervisada pues durante este proceso no hay clases predefinidas ni registros que permitan conocer las relaciones existentes entre los datos. En esta técnica, los grupos se van formando de acuerdo a las características de los datos, maximizando la similitud dentro de los grupos pero a la vez minimizando la similitud entre los distintos grupos [20]. De esta manera, se busca que los objetos que pertenecen a un grupo sean homogéneos entre sí, y que los distintos grupos sean lo más heterogéneos posible (Figura 3).

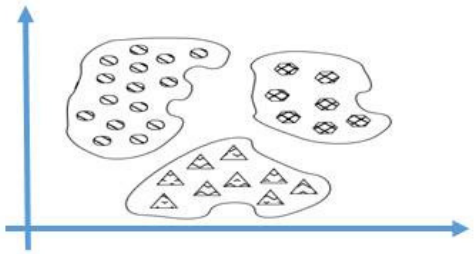

Figura 3: Representación Gráfica del Resultado del *Clustering*

Los algoritmos de *Clustering* intentan minimizar la distancia dentro del grupo de objetos y maximizar la distancia entre grupos, por lo que en ocasiones surge ambigüedad pues no siempre es claro por cuál características agrupar o cuántos grupos hacer. En un modelo de dominio para *clustering* multidimensional, los distintos algoritmos de *clustering* (K-Means, EMI, entre otros) pueden aprovechar los datos estructurados en un modelo multidimensional, de tal manera que los hechos del espacio multidimensional se relacionan directamente con las técnicas de minería de datos, como se observa en la Figura 4 [14].

Los diferentes algoritmos de *clustering* en espacios multidimensionales se caracterizan por:

- Identifican comportamientos comunes en un conjunto de datos cuyos usuarios no podrían derivar a través de la observación casual, aprovechando la potencia de los modelos multidimensionales para descubrir grupos con comportamientos similares.
- La estructura multidimensional facilita la comprensión de los datos, dado que representa el dominio del sistema de una manera muy cercana a la forma de pensar de los analistas.

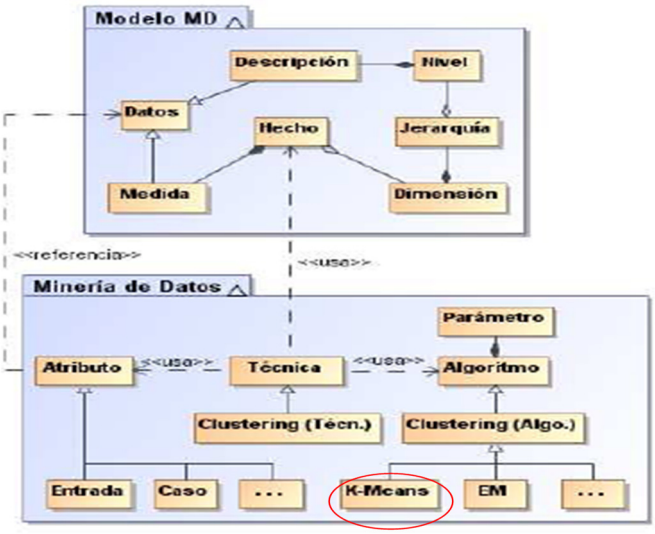

Figura 4: Modelo del Dominio de *Clustering* Multidimensional

- El resultado del análisis es una estructura de partición del conjunto de datos. Por lo que un apropiado modelo multidimensional facilita la representación de los datos en estudio.
- La técnica de *clustering* descubre grupos de objetos similares o con patrones de comportamiento comunes en base al hecho en estudio, considerando las distintas dimensiones en cualquier nivel de detalle en sus jerarquías.
- El modelo multidimensional representa de una manera apropiada y fácil los datos a analizar. Por ejemplo, en la Figura 5 [14] se observa que el hecho bajo análisis (H) contiene medidas (M1 y M2) que son contextualizadas por las dimensiones (D1 a D6). Cada dimensión usada como entrada representará un eje de *clustering* y cada caso corresponderá a particiones de esos ejes (planos C1, C2 y C3). Los *clusters* se muestran como agrupaciones de puntos presentes en las particiones de los ejes de *clustering* que representan las dimensiones usadas como entrada al proceso de análisis.

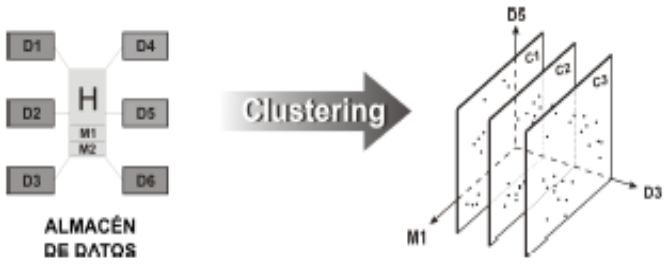

Figura 5: *Clustering* en Minería de Datos

- Los algoritmos de *clustering* tienen como entradas los atributos que utiliza para construir el espacio multidimensional en el cual se miden las similitudes de los datos. La salida de este proceso es un número de

clusters que forman una partición del conjunto de datos en el espacio multidimensional.

C. Clustering Difuso

En el *clustering* clásico, cada patrón pertenece a un único *clúster*. Sin embargo, existen situaciones reales donde los objetos agrupados, debido a su naturaleza, no sólo pertenecen a una partición excluyente e inequívoca, sino que podrían pertenecer a varias particiones que se solapan. Esto genera la necesidad de realizar agrupamientos más flexibles donde la similitud entre los objetos y la pertenencia de un objeto a *clúster* no sea precisa. Una manera de representar esta pertenencia gradual es a través de conjuntos difusos [21], que se caracterizan por una función de membresía cuyo rango está en el intervalo real [0,1]. Cuando el grado de membresía de un elemento es cercano a 1, se dice que está más posiblemente (o certeramente) incluido en el conjunto. Así 0 es la medida de completa exclusión y 1 la de completa inclusión. De esta forma se puede representar cuando un objeto tiene una pertenencia difusa a un grupo. En la Figura 6 [2] se observa el resultado de realizar *clustering* difuso, que produce dos *clústers* F1 y F2, con datos (4, 6, 7) en la intersección. El problema del agrupamiento difuso es encontrar la caracterización de una partición difusa óptima, en base a una relación de similitud entre los objetos.

Son muchos los casos donde resulta útil aplicar un análisis de agrupamiento difuso. Uno de ellos es el agrupamiento de noticias en la web, donde la clasificación de la naturaleza de una noticia es una partición difusa ya que la misma puede pertenecer a diferentes categorías (deportivas, cultural, económica, social, etc.). Otra aplicación en el medio de los negocios es la segmentación de clientes utilizando la agrupación difusa.

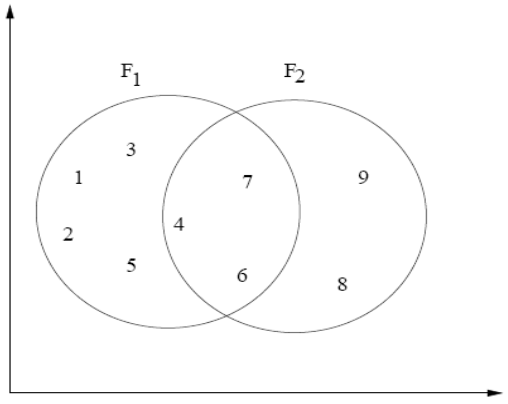

Figura 6: *Clustering* Difuso

D. Algoritmo Fuzzy C-Means

Fuzzy C-Means es un algoritmo de *clustering* difuso de gran difusión, introducido por Ruspini [22], formalizado por Dunn [23] y generalizado por Bezdek [24], cuya idea es obtener particiones difusas óptimas del conjunto de objetos, minimizando una función objetivo que determina los prototipos o centroides de los grupos buscados. Una presentación detallada del *Fuzzy C-Means* y sus versiones se encuentra en Bezdek [25]. Otros algoritmos de clustering difuso pueden encontrarse en [26] así como un análisis comparativo que permite medir el desempeño de éstos sobre diferentes conjuntos de datos. Algunas aplicaciones de *clustering* difuso se describen en [27][28]. Aquí se utilizará una versión general descrita por Hernández [2]. El algoritmo *Fuzzy C-Means* tiene los siguientes pasos [24][28][29]:

1) Dada la matriz de pertenencia $\boldsymbol{\mu}_{n\times k}$ donde un elemento μ_{ij} representa el grado de membresía del objeto i al clúster j, tal que $\mu_{ij} \in [0,1]$, se selecciona una partición difusa inicial de n en k clústers por medio de dicha matriz de pertenencia.

2) Se utiliza $\boldsymbol{\mu}$ para encontrar el valor de la función objetivo de criterio difuso, la cual se explica más adelante. Se reasignan los datos a los clústers para reducir el valor de la función de criterio y se reevalúa $\boldsymbol{\mu}$.

3) Se repite el paso 2 hasta que los valores de $\boldsymbol{\mu}$ no cambien significativamente.

De esta manera, el algoritmo *Fuzzy C-Means* asigna un conjunto de objetos, caracterizados por sus respectivos valores de atributos, a un número *c* determinado de clases (grupos). El resultado del algoritmo *Fuzzy C-Means* se muestra en una tabla donde cada objeto tiene un grado de pertenencia μ_{ij} a cada clase, representada por su centro de clases o grupos construidos, por ello el número de grupos suele ser un parámetro *c* conocido.

Básicamente, el algoritmo *Fuzzy C-Means* requiere de los siguientes parámetros [24][29]:

- Conjunto de Datos: X
- Número de clases o grupos difusos a encontrar: c
- Número de objetos a agrupar: n
- Difusor o grado de difusión: m. Se trata del factor difuso que indica cuánto se quiere que se solapen los grupos. Tiene que cumplirse $m > 1$ ya que la partición se vuelve más difusa conforme se incrementa m y con m=1 la partición dejaría de ser difusa.
- Vector de atributos del objeto j: $y_j, j = 1, \ldots, n$
- Grado de membresía del objeto i a clase j: μ_{ij}, $i = 1,\ldots,n, j = 1,\ldots,c$.

Se han propuesto varios criterios de agrupamiento para obtener la partición difusa óptima. Una de las funciones objetivo de criterio difuso más utilizada es la propuesta por Dunn [23], la cual está asociada con la función de error mínimo cuadrático. Dunn propone minimizar iterativamente la siguiente fórmula $J_m(U,v) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m d_{ik}^2$. Donde

U: es una matriz de pertenencias, con la *c-partición* difusa de X, contiene el grado de membresía de cada objeto a cada grupo, $U \in M_{fc}$: conjunto de *c-particiones* difusas

$V_i = (v_{i1}, v_{i2}, \ldots, v_{ic})$ es el vector centro del grupo i, es decir, el conjunto de particiones difusas

d_{ik}^2: indica la distancia cuadrada entre los elementos de X= (x_1, x_2, …, x_n), el conjunto de n objetos que es subconjunto del espacio euclidiano de dimensión s con $X \in \mathbb{R}^s$, y los centros de los grupos, en decir distancia cuadrática entre el objeto k el centro V del *clúster* i, calculados con: $d_{ik}^2 = ||x_k - v_i||_A^2 = (x_k - v_i)^T A(x_k - v_i)$ siendo $\| \cdots \|$ la norma inducida por A.

La idea es buscar la partición que produzca la distancia mínima de los objetos al centro de su grupo. Esta distancia está

ponderada por el grado de membresía de cada objeto a un *clúster* y por el factor difuso *m* que indica cuánto se quiere que se solapen los grupos. Para minimizar esta función, se utilizará la propuesta de Bezdek [24], la cual intenta minimizarla de manera iterativa usando el siguiente teorema: "una partición difusa puede ser un mínimo local de la función objetivo *J*, para $m > 1$", cuando se cumplen las siguientes condiciones:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{2/(m-1)}} \text{ y } u_i = \frac{\sum_{k=1}^{n}(u_{ik})^m x_k}{\sum_{k=1}^{n}(u_{ik})^m}$$

III. Modelación Conceptual de Clustering Difuso en Espacios Multidimensionales

En base a los conceptos teóricos mencionados, se presenta a continuación el perfil UML propuesto para modelar *clustering* difuso multidimensional y sus restricciones en OCL.

A. Perfil UML para Clustering Difuso

Con el objetivo de facilitar el análisis de agrupamiento como técnica de minería de datos en un proceso KDD, se propuso en [14] la integración de esta técnica con almacenes de datos, a través de una extensión de *profile* UML cuyo fin es el modelado conceptual de minería de datos con *clustering* en espacios multidimensional. En esta extensión se definen cuatro estereotipos: «clustering» que representa una generalidad del algoritmo con sus parámetros; «entrada» que son los atributos de entrada a la técnica de *clustering* que referencia datos a través de los hechos; «caso» que son los atributos utilizados como caso; y «atributo abstracto» que son aquellos atributos de minería de datos que hacen referencia a los datos multidimensionales. Los ajustes al *clustering* se toman de los parámetros del algoritmo utilizado, por lo que para el caso de *clustering* difuso, se usan los del algoritmo *Fuzzy C-Means*.

En la Figura 7, una adaptación de la propuesta de Zubcoff [14], se muestra el perfil UML para minería de datos con *clustering* difuso, donde en la parte izquierda se observa un extracto de la especificación del perfil UML para modelación multidimensional de DW, propuesto en [18]. Allí se definen las cajas etiquetadas correspondientes a los estereotipos («stereotype») y a las metaclases («metaclass»). Los conceptos del modelo multidimensional (hechos, dimensiones y jerarquías de agregación) son traducidos a la metaclase UML **Class** con los estereotipos **Fact**, **Dimension** y **Base**. Asimismo, los datos multidimensionales: como las medidas (estereotipo **FactAttribute**), las descripciones de los niveles de jerarquía (estereotipo **DimensionAttribute**) y los identificadores de los objetos (estereotipo **OID**). Estos elementos se traducen a la metaclase UML **Property** que típicamente modela atributos de otras metaclases.

En la parte derecha de la Figura 7 se muestra el extracto de la especificación del perfil UML para *clustering* difuso. Aquí se observa el estereotipo **Clustering** que representa la generalización de los algoritmos de *clustering* difuso, que se definen extendiendo la metaclase UML **InstanceSpecification**. La clase **Ajustes** modela los parámetros de los algoritmos de *clustering* difuso, indicando para cada parámetro el dominio y el valor por defecto. Los estereotipos **Entrada**, **Caso** y el **Atributo Abstracto** son tomados de [14] con la misma interpretación, donde la etiqueta referencia permite enlazar con los datos asociados el modelo multidimensional.

B. Restricciones OCL

En cuanto a las restricciones OCL propuestas en [14] para enriquecer la semántica que no puede ser totalmente expresada por el perfil, éstas no pierden vigencia en esta propuesta, por lo que serán reutilizadas también. Entre ellas se destacan las condiciones necesarias para completar el perfil UML, a fin de resolver ambigüedades del dominio de *clustering*: "considerar al menos una entrada para *clustering*", "los parámetros ajustan el *clustering*", "las entradas referencian datos multidimensionales", "los atributos caso pueden referenciar solamente datos multidimensionales descriptivos", "el número de atributos de entrada está limitado en *clustering*". El detalle de estas restricciones puede verse en [14].

Para las restricciones propias de los requisitos de *clustering* difuso se utilizará la extensión de OCL propuesta en [9] que permite incluir términos vagos en las expresiones que representan la semántica formal de tales requisitos. Los términos de la lógica difusa que permite esta extensión abarcan: predicados, modificadores, comparadores, conectores y cuantificadores difusos.

IV. Caso de Estudio

Con el propósito de mostrar la aplicabilidad y viabilidad de la propuesta presentada, a continuación se expone un caso de estudio genérico y simplificado. Club Mercado es un proyecto de desarrollo de una aplicación Web para la compra y *marketing* de productos en línea. Se basa en el consumo colaborativo proveyendo a los clientes de búsquedas para

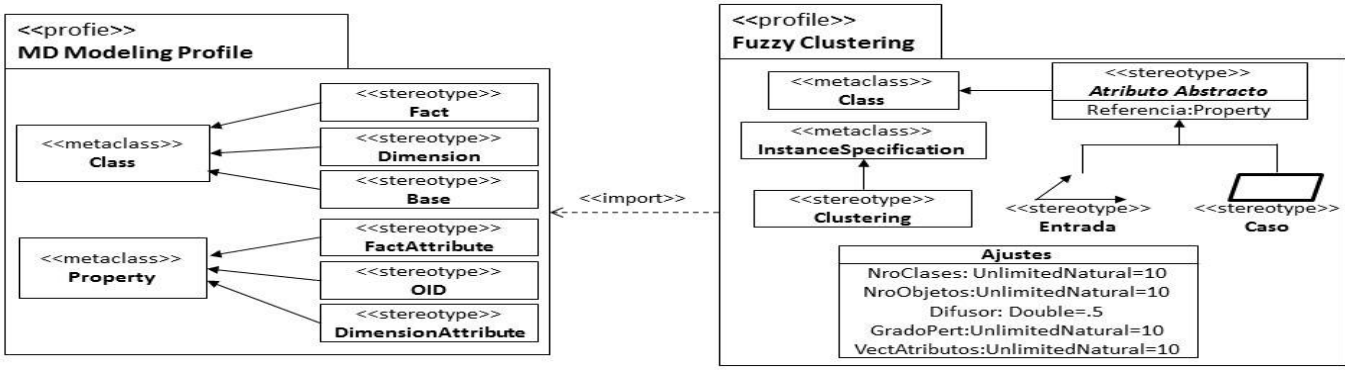

Figura 7: Perfil *Clustering* Difuso

adquirir productos a mejores precios y de superior calidad obtenidos directamente de los productores o distribuidores. La idea es que el sistema sea un apoyo para segmentar los clientes en grupos que tengan conductas similares en cuanto a hábitos de compras, y para agrupar los productos por categorías de mejores precios y/o mejor calidad. De manera especial se espera que el sistema realice el análisis de segmentación de los clientes de forma tal que entregue un detallado conocimiento del perfil de cada cliente a fin que la recomendación de los productos a través del sistema sea consistente con las necesidades y preferencias de éste. El perfil del cliente ha sido segmentado en diferentes grupos, con sus respectivos grados de pertenencia, respetando con mayor fidelidad los intereses reflejados por dichos clientes durante sus visitas y transacciones de compras realizadas históricamente en la aplicación web. Se desea que el sistema sea accesible por medio de dispositivos móviles con una base de datos que permita reducir el uso del internet y en algunos casos prescindir de éste. Las funcionales a proveer son: el registro de usuarios, manejo de carrito de compras, estados de cuenta de las compras realizadas, oferta y recomendación de productos basado en el perfil de pertenencia de cada cliente a través de diversos medios (en comerciales, anuncios de prensa etc.), consultas de productos según preferencias usuario, estadísticas de productos más vendidos a fin de hacer descuentos a los clientes, entre otros.

La interfaz para el registro de usuario solicita los siguientes campos: nombre, apellido, cédula de identidad, edad, número celular, correo electrónico, contraseña, confirmación de contraseña, dirección del usuario, estado, ciudad y municipio a la cual pertenece la dirección especificada. Los datos más relevantes para los productos, incluyen su código, descripción, precio, cantidad, categoría, fechas de inicio y cierre de anuncio publicitario, así como, fotos alusivas. El pago de las compras de los clientes se realizará a través de tarjetas de crédito. Todos los datos son conglomerados en un Almacén de Datos (AD) cuyo modelo multidimensional (MD) se presenta en la Figura 8. Es de notar que el análisis de este caso de estudio está enfocado en un nivel conceptual y de modelamiento, por ellos no se incluyen detalles relacionados a los valores de estos datos (tamaño, dimensionalidad, entre otros). Es decir, el análisis cuantitativo del modelo se escapa de los objetivos del presente trabajo, sin embargo, esta información no afecta la aplicabilidad a nivel conceptual del perfil UML propuesto.

El diagrama presentado en la Figura 8, se obtuvo utilizando el perfil UML para modelado MD [14] descrito en la sección 3.1. Una compra es un hecho de análisis (estereotipado como «fact»), cuya clase se ha identificado como Transacciones, la cual contiene el atributo cantidad de la compra (etiquetado como «FA»). Para el análisis del contexto, se presentan tres instancias de la clase «dimensión»: **Fecha**, **Titular** y **Productos**. Estas dimensiones agregan (flechas con punta de diamante) información a través de jerarquías de agrupación a la compra. Cada nivel de granularidad se indica con la etiqueta «Base». Además, cada nivel de agregación tiene atributos descriptivos. En el caso de la jerarquía definida por la dimensión **Titular** de la tarjeta, con tres niveles: «Base» Usuarios, «Base» Tipo Tarjeta y «Base» Ingresos.

En el primer nivel se observa la cédula de la persona como el identificador de objeto (etiquetado como «OID») y los atributos de la dimensión (etiquetados como «DA»): Nombre, Edad, Teléfono Celular, Sexo, Correo, Contraseña, Confirmación de la Contraseña, Dirección, Usuario, Estado, Ciudad, Municipio. En esta dimensión hay dos jerarquías de agregación que comparten el mismo nivel de granularidad, «Base» Tipo Tarjeta

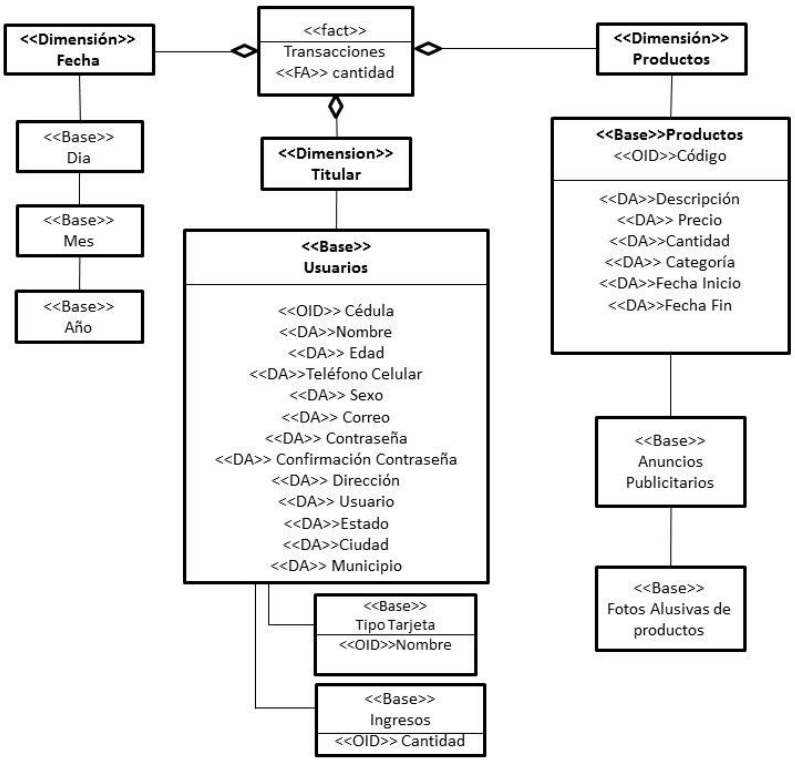

Figura 8: Modelo Multidimensional para Club Mercado

y «Base» Ingresos. Estos niveles de granularidad permiten a los analistas trabajar de manera intuitiva a distintos niveles de detalle, desde las etapas tempranas del desarrollo asegurando la calidad de los datos. En el nivel «Base» Tipo Tarjeta se tiene como identificador de objeto («OID») el nombre del titular de la tarjeta y en el nivel «Base» Ingresos a la cantidad («OID»). En estos niveles no hay más atributos porque corresponden a los datos de entrada necesarios para construir el modelo multidimensional en el cual se miden las similitudes existentes en los datos observados.

Para el análisis del contexto en el caso de la dimensiones **Productos** y **Fecha** de la compra se modelan cada uno con tres niveles de jerarquía. En la dimensión **Productos** se tienen los niveles «Base» Productos, «Base» Anuncios Publicitarios y «Base» Fotos alusivas a productos. En la dimensión **Fecha** los tres niveles de jerarquía son «Base» día, «Base» mes y «Base» año, con la finalidad de permitir diferentes consultas necesarias por los atributos descriptivos.

Para abordar el requisito de segmentación difusa de los clientes almacenados en el MD, se modeló un requisito de análisis de agrupamiento difuso sobre los clientes, en donde el objetivo de la analítica se resume en obtener una estructura de partición (agrupación) difusa (no disjunta) de todos los clientes potenciales del Club Mercado. Se quiere que la partición represente los criterios de preferencias de los clientes demostrada en los históricos de compras por tarjetas de crédito, y así construir un modelo de conocimiento que describa el perfil de *marketing* útil para especializar el sistema recomendador y de ofertas de productos a las necesidades específicas de cada cliente. Es importante resaltar que este modelo no resultaría tan real si se representa con una partición no difusa dentro de un contexto de clientes con diversidad de preferencias, poco sesgadas, donde sus intereses están solapados entre las categorías de la partición.

En este análisis de segmentación difusa de clientes, al aplicar el algoritmo *Fuzzy C-Means*, los clientes son el conjunto de objetos a particionar, caracterizados con los valores de los atributos descritos en la dimensión Titular, analizando las Transacciones que dichos Titulares han realizado. El modelo de agrupamiento difuso obtenido se observa en la Figura 9, que usa la propuesta de Rodríguez y Goncalves [9].

Para esto se usa una instancia de la clase **Ajustes** del perfil de agrupamiento difuso, estereotipada como «clustering». Para este modelo de agrupamiento, se han ajustado los valores de los parámetros como sigue: para NroClases se aplicará el proceso para valores de c desde c=2 hasta y c=10, El NroObjetos indica el número de clientes que en este caso son 25 millones, el factor difusor (mínSoporte) Difusor=10 se ajustó alto para garantizar una partición más difusa, manteniendo el resto de los parámetros con sus valores por defecto, en vista que variar estos valores no aportan diferencias al modelamiento. Las flechas punteadas indican que los atributos son tipos de datos dependientes y la etiqueta *use* que son datos de entrada.

Durante el proceso de *clustering* difuso sobre las transacciones de compras con tarjetas de créditos se utiliza la cantidad comprada como un atributo de entrada, lo cual se indica con la etiqueta [Referencia=Compra:Cantidad]. Los atributos del usuario (como cedula, nombre, edad, sexo, dirección, estados, teléfono, municipio, ciudad, tipo de tarjeta, ingresos, contraseñas, correo, cantidad), aparecen con la etiqueta *use* indicando que son datos de entrada.

En cuanto a los valores de referencia como es el caso de los atributos Tipo de Tarjeta, Cantidad, e Ingresos, corresponden a los atributos de entrada que permitirán la partición difusa de los datos del esquema multidimensional asociados a los clientes basado en el análisis de sus transacciones.

Al aplicar el algoritmo *fuzzy c-means*, se determinan los parámetros de este algoritmo utilizando alguna de las técnicas heurísticas más usadas sobre la base del estudio de casos, para luego modificar el número de agrupaciones. En este caso práctico se propone aplicar el algoritmo *fuzzy c-means* para cada número de agrupaciones entre c = 2 y c = 10. De tal manera que los valores de pertenencia de todos los clientes a

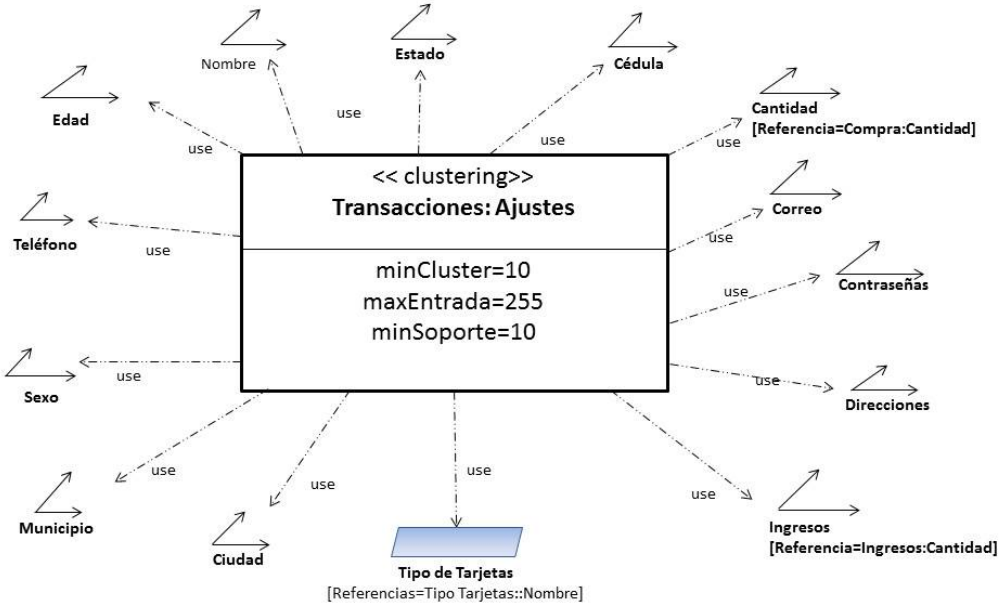

Figura 9: Modelo *Clustering* Difuso MD Compras TC

las agrupaciones 1 a c son calculados y presentados. El algoritmo finalmente muestra una matriz con el centro de todas las agrupaciones (clases) c.

De esta manera el análisis de partición difusa proporciona grados de pertenencia de los objetos a cada una de las agrupaciones. En este sentido, en la segmentación de clientes se discrimina de manera más diferenciada cada cliente en cada clase, en vista que el algoritmo calcula para cada cliente los valores de pertenencia en cada clase. De esta manera los clientes que muestran valores característicos de diferentes segmentos se tratarán con actividades de *marketing* especializadas de acuerdo a sus valores de pertenencia. Como resultado, el recomendador de *marketing* estará en capacidad de ofrecer a los clientes del sistema productos más ajustados a sus intereses, preferencias y necesidades.

Se puede observar que el perfil UML propuesto para *clustering* difuso simplifica el proceso del analista en la labor de modelado, abstrayéndolo hacia niveles altos del proceso de partición difusa separándole de los detalles de implementación que son mucho más complejos. De esta forma, resulta más sencillo configurar el requisito de agrupamiento difuso y modelarlo a través del perfil propuesto, el cual aprovecha las ventajas ofrecidas por el MD. También se facilita el proceso de realizar un análisis de agrupamiento difuso, a través de una notación más intuitiva que se abstrae e independiza de los complejos detalles de su implementación.

V. Conclusiones y Trabajos Futuros

En esta contribución se han propuesto un perfil UML para diseño conceptual de *clustering* difuso en el tope del modelo multidimensional de un almacén de datos. Este ha sido trasladado a una extensión del paquete Profile de UML 2.0, mecanismo de extensión ligera de UML.

Esta propuesta permite diseñar modelos de *clustering* difuso en espacio multidimensional articulado por un almacén de datos, al nivel de abstracción apropiado para concentrarse exclusivamente en los principales conceptos del *clustering* y aprovechando toda la información y el conocimiento del dominio bajo estudio capturado en el modelo multidimensional del almacén de datos. Este modelo facilita la abstracción, flexibilidad y reusabilidad, provee a los usuarios la semántica requerida para la comprensión del sistema modelado, simplificando el diseño de la minería de datos con técnicas difusas de *clustering* en una notación intuitiva. La principal ventaja de esta propuesta es que permite a los analistas llevar a cabo el proceso de KDD estableciendo los objetivos empresariales desde etapas tempranas del desarrollo del proyecto, asegurando así la calidad de los datos.

Los aportes principales de esta propuesta son:

- Facilita el diseño del proceso de minería de datos gracias al uso de modelos conceptuales considerando los objetivos empresariales desde etapas tempranas del proyecto de KDD.
- Incorporar a tempranas etapas del proceso de análisis, construcciones con una notación especializada para modelar la semántica relacionada con requisitos de agrupamiento difuso.
- Provee un Modelo Conceptual para *clustering* difuso, independiente de herramientas y algoritmos específicos, incorporando una nueva notación y semántica para símbolos ya existentes en UML.
- Proporciona una sintaxis y terminología común para el dominio de aplicaciones de agrupamiento difuso, cuyas construcciones actualmente no cuentan con una notación propia.
- Facilita el diseño de minería de datos en espacios multidimensionales.
- Aprovecha las ventajas derivadas de los pasos previos del DW, asegurando la calidad de los datos al integrar los DW al proceso global de KDD con requisitos de *clustering* difuso.
- Provee un camino para el modelado de software de minería de datos difusas guiado por arquitecturas (MDA, *Model Driven Architecture*), mediante la definición y transformación de modelos para este dominio de aplicación de uso y relevancia en la actualidad.

A partir de esta propuesta quedan caminos abiertos, de los cuales se quiere explorar en trabajos futuros, los siguientes:

1. Aplicación completa del perfil UML propuesto en el caso de estudio genérico, culminando el proceso de KDD utilizando datos reales, con el fin de ofrecer los resultados del análisis de segmentación difuso obtenido de manera cuantificada, ofreciendo una comparativa con los resultados alcanzados en la aplicación del caso homólogo preciso.
2. Proponer una metodología para tratamiento de requisitos difusos con técnicas de minería de datos.
3. Extender UML a un perfil de modelación de KDD con diversas técnicas de minería de datos difusa, visto como proceso integrado y como tratamiento de requisitos difusos.
4. Aplicación de la propuesta de extensión de perfiles UML para *clustering* difuso, en diferentes casos de estudio de interés real, tales como: segmentación de pozos petroleros sobre un DW de variables de producción y explotación de la Industria Petrolera Venezolana, segmentación de perfiles de estudiantes que han desertado del sistema educativo formal venezolano, así como segmentación de pacientes con enfermedades metabólicas, proclives a desarrollar enfermedades crónicas y degenerativas. Esto permitirá validar si esta propuesta es un modelo de diseño conceptual novedoso para *clustering* difuso sobre DW.

Proponer un escenario de transformaciones, de los perfiles UML propuestos para minería de datos difusa, que puedan ser automatizadas, de tal manera de realizar un aporte al modelado de software de minería de datos difusa guiado por arquitecturas (MDA, *Model Driven Architecture*).

Agradecimientos

Agradecemos Aquél que nos da fe y valor para emprender proyectos hacia lo desconocido: “Por la fe Abraham, siendo llamado, obedeció para salir al lugar que había de recibir como herencia; y salió sin saber a dónde iba” (Hebreos 11:8).

REFERENCIAS

[1] P. Tan and M. Steinbach, V. Kumar. *Introduction to Data Mining*. Addison Wesley, USA, 2006.

[2] E. Hernández. *Algoritmo de Clustering basado en Entropía para Descubrir Grupos en Atributos de Tipo Mixto*. Tesis para obtener el grado de Maestro en Ciencias en la Especialidad de Ingeniería Eléctrica Opción Computación. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, México, D.F. 2006. https://www.cs.cinvestav.mx/TesisGraduados/2006/tesisEdnaHernandez.pdf.

[3] U. Fayyad. *Data Mining and Knowledge Discovery: Making Sense out of Data*. IEEE Expert, vol. 11, no. 5, pp. 20-25, October 1996.

[4] O. Moscoso-Zea, A. Sampedro, and S. Luján-Mora. *Datawarehouse Design for Educational Data Mining*. 2016 15th International Conference on Information Technology Based Higher Education and Training (ITHET), pp. 1-6, Istanbul, Turkey, September 2016.

[5] R. Agrawal, A. Gupta, and S. Sarawagi. *Modeling Multidimensional Databases*. Technical Report. IBM. IBM Almoden Research Center. 1995. https://infolab.usc.edu/csci599/Fall2002/paper/I3_agrawal95modeling.pdf.

[6] L. Cabibbo and R. Torlone. *A Logical Approach to Multidimensional Databases*. EDBT ´98 Proceedings of the 6th International Conference on Extending Database Technology, vol. 1, pp. 183-197, Valencia, España, March 1998.

[7] A. Datta and H. Thomas. *The Cube Data Model: a Conceptual Model and Algebra for On-line Analytical Processing in Data Warehouses*. Decision Support Systems, vol. 27, no. 3, pp. 289-301. December 1999.

[8] A. Gosain, S. Sabharwal, and S. Nagpal. *Predicting Quality of Data Warehouse using Fuzzy Logic*. International Journal of Business and Systems Research (IJBSR), vol. 6, no. 3, pp. 255-268. January 2012.

[9] R. Rodríguez y M. Goncalves. *Perfil UML para el Modelado Visual de Requisitos Difusos*. Enl@ce: Revista Venezolana de Información, Tecnología y Conocimiento, vol. 6, no. 3, pp. 29-46, Septiembre 2009.

[10] R. Rodríguez y L. Tineo. *Elementos Gramaticales y Características que Determinan Aplicaciones con Requerimientos Difusos*. Revista Tekhne, vol. 12, pp.50-64, Enero 2009.

[11] R. Rodríguez y M. Goncalves. *Implementación de Requisitos en Sistemas Orientados a Datos con Lenguaje OCL y Lógica Difusa*. Enl@ce Revista Venezolana de Información, Tecnología y Conocimiento, vol. 8, no. 1, pp. 31-54, Enero 2011.

[12] W. Pereira y L. Tineo. *Modelo Orientado a Objetos Difuso*. Acta Científica Venezolana, vol. 51, no. 2, pp. 357, Noviembre 2000.

[13] G. Booch, J. Rumbaugh, and I. Jacobson. *The Unified Modeling Language User Guide*. Addison Wesley. USA. 2005.

[14] J. Zubcoff, J. Pardillo, and J. Trujillo. *Integrating Clustering Data Mining into the Multidimensional Modeling of Data Warehouses with UML Profiles*. In Data Warehousing and Knowledge Discovery. DaWaK 2007. Lecture Notes in Computer Science, 4654:199-208. Springer, Berlin, Heidelberg. Septiembre 2007.

[15] ISO/IEC. *Unified Modeling Language (UML). Version 1.5*. International Standard ISO/IEC 19501.

[16] L. Fuentes y A. Vallecillo. *Una Introducción a los Perfiles UML*. Novática: Revista de la Asociación de Técnicos de Informática, ISSN 0211-2124, no.168, pp. 6-11, Enero 2004.

[17] Object Management Group. *Object Constraint Language Specification, version 2.0*. http://www.omg.org/technology/documents/ formal/ocl.htm.

[18] S. Lujan-Mora, J. Trujillo, and I. Song. *A UML Profile for Multidimensional Modeling in Data Warehouses*. Data & Knowledge Engineering, vol. 59, no. 3, pp. 725-769. December 2006.

[19] S. Mitra and T. Acharya. *Data Mining: Multimedia, Soft Computing and Bioinformatics*. Wiley-InterScience, John Wiley & Sons, Inc., Publication, USA, 2003.

[20] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc, Elseiver, UK, 2006.

[21] L. Zadeh. *Fuzzy Sets. Information and Control*, vol. 8, no. 3, pp. 338-353, 1965.

[22] E. H. Ruspini. *Numerical Methods for Fuzzy Clustering*. Information Sciences, vol. 2, no. 3, pp. 319-350, July 1970.

[23] J. C. Dunn. *A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact Well-Separated Clusters*. Journal of Cybernetics and Systems, vol. 3, no. 3, pp. 32-57, September 1973.

[24] J. Bezdek, R. Ehrlich, and W. Full. *FCM: The Fuzzy c-Means Clustering Algorithm*. Computers & Geosciences, vol. 10, no. 2-3, pp. 191-203, 1984.

[25] J. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers Norwell, MA, USA. 1981.

[26] A. Gosaina and S. Dahiya. *Performance Analysis of Various Fuzzy Clustering Algorithms: A Review*. In Proceedings of the International Conference on Communication, Computing and Virtualization (ICCCV), vol 79. pp. 100-111, Procedia Computer Science. Elseiver. Mumbai, India, Febrary 2016.

[27] W. Meier, R. Weber, and H. Zimmermann. *Fuzzy Data Analysis - Methods and Industrial Applications*. Fuzzy Sets and Systems, vol. 61, no. 1, pp.19-28, January 1994.

[28] J. Strackeljan and R. Weber. *Quality Control and Maintenance*. In: Practical Applications of Fuzzy Technologies. The Handbooks of Fuzzy Sets Series, vol 6. Springer, Boston, MA, pp. 161-184. 1999.

[29] J. Bezdek, M. Pal, J. Keller, and R. Krisnapuram. *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Kluwer Academic Publishers Norwell, MA, USA, 1999.

Revista Venezolana de Computación
ISSN: 2244-7040
http://www.svc.net.ve/revecom
Vol. 5, No. 2, pp. 41-52, Diciembre 2018
Selección de los Mejores Artículos de CoNCISa 2018

Reconocimiento de Estados Emocionales de Personas Mediante la Voz Utilizando Algoritmos de Aprendizaje de Máquina

Nerio Morán[1,2], Jesús Pérez[1,2], Wladimir Rodriguez[2]
neriojmoran@ula.ve, jesuspangulo@ula.ve, wladimir@ula.ve

[1] LaSDAI, Universidad de Los Andes, Mérida, Venezuela
[2] Departamento de Computación, Universidad de Los Andes, Mérida, Venezuela

Resumen: El reconocimiento de estados emocionales de las personas se ha popularizado en aras de mejorar las interacciones entre personas y robots. Actualmente, los investigadores han mostrado un creciente interés por desarrollar técnicas que permitan reconocer emociones a través de la voz. Las técnicas más populares para reconocer emociones mediante la voz, utilizan bases de datos con registros de voz de diferentes personas que expresan diferentes emociones, para entrenar algoritmos de aprendizaje de máquina. Particularmente, las emociones humanas pueden ser expresadas de diversas maneras, lo cual afecta la capacidad de reconocimiento de estos algoritmos, y en consecuencia, la capacidad de interacción eficaz de los robots, ya que reconocer todas las formas de expresión de una misma emoción a través de la voz es una tarea compleja. En este sentido, en aras de proporcionar la capacidad a los robots de reconocer emociones de un amplio grupo de personas, en esta investigación se construye una base de datos en condiciones controladas y actuadas de seis emociones (ira, sorpresa, felicidad, miedo, tristeza y asco). Luego, con el propósito de hacer comparaciones, se entrenan tres modelos de aprendizaje automático (Máquinas de Vectores de Soporte, Bosques Aleatorios y Aumento del Gradiente). Posteriormente, se construyen dos bases de datos adicionales (una en condiciones controladas y semi-naturales, y otra en condiciones no controladas y naturales) para probar con mayor rigurosidad los modelos entrenados. Los resultados obtenidos indican que la mejor tasa de reconocimiento se obtiene cuando se hacen predicciones sobre muestras capturadas en las mismas condiciones que las muestras de la base de datos de entrenamiento, y además, para muestras pertenecientes a las otras bases de datos hay resultados prometedores, como por ejemplo, la alta tasa de reconocimiento de la ira en todas las pruebas realizadas.

Palabras Clave: Emociones; Reconocimiento; Aprendizaje de Máquina; Voz.

Abstract: The emotional state recognition has become popular in order to improve the interactions between people and robots. Currently, researchers have shown a growing interest in developing techniques to recognize emotions from human speech. The most popular techniques for recognizing emotions from speech, use databases with voice records of different people expressing different emotions, to train machine learning algorithms. Particularly, human emotions can be expressed in different ways, which affects the recognition capacity of these algorithms, and consequently, the ability of robots to interact effectively, since recognize all forms of expression of an emotion from speech is a complex task. Therefore, in order to provide robots with the ability to recognize emotions of a large group of people, this research builds a database under controlled and acted conditions of six emotions (anger, surprise, happiness, fear, sadness and disgust). Then, in order to make comparisons, three machine learning models are trained (Support Vector Machines, Random Forests and Gradient Boost). Subsequently, two additional databases are constructed (one under controlled and semi-natural conditions, and another under uncontrolled and natural conditions) to test the trained models with greater rigor. The results obtained indicate that the best recognition rate is obtained when predictions are made on samples captured in the same conditions as the samples of the training database, also, for samples belonging to the other databases, there are promising results, as for example, the high rate of anger recognition in all tests performed.

Keywords: Emotions; Recognition; Machine Learning; Speech.

I. Introducción

A lo largo de los años, ha sido el ser humano quien se ha adaptado a las diferentes formas de comunicación que ofrecen las computadoras. Investigaciones actuales, están dirigidas por la iniciativa de disminuir la brecha de comunicación entre personas y robots. Para ello, algunos de los aspectos que se consideran son el reconocimiento y adaptación de las computadoras según el estado emocional de la persona [1]. El reconocimiento de emociones es realizado mediante diferentes medios, tales como: la voz [2]–[11], imágenes de rostros [12], conductancia de la piel [13], frecuencia cardíaca [14], señales inalámbricas [15], entre otros. Dado que las señales de la voz se consideran fáciles de obtener y es una de las formas de comunicación más usadas, se le considera como una de las fuentes de información más adecuadas para la clasificación de emociones [4].

Dentro de las aplicaciones más resaltantes de los algoritmos de aprendizaje de máquina aplicados al reconocimiento de emociones mediante la voz, están los robots sociales de asistencia personal [16], con la capacidad de detectar emociones y regularlas. El objetivo de estos robots es mantener el bienestar del estado afectivo de las personas ubicadas en un entorno inteligente. Cada robot cuenta con dos componentes principales de reconocimiento: voz y expresiones faciales; los cuales usa de manera conjunta para determinar los estados afectivos y regularlos en caso de ser necesario. Para entrenar los algoritmos de aprendizaje de máquina, han sido utilizadas múltiples bases de datos, y dentro de las más populares se encuentran: “A Database of German Emotional Speech”, también conocida como Emo-DB [17], “Polish Emotional Speech Database” [18], “The eNTERFACE’05 audio-visual emotion database” [19], “Surrey Audio-Visual Expressed Emotion”, también conocida como SAVEE [20], entre otras. Estas bases de datos cuentan con múltiples muestras de audio en un idioma específico, etiquetadas con distintos estados emocionales, que son procesadas para extraer diferentes características y servir como entrada a los algoritmos de clasificación.

Las investigaciones actuales, han sugerido la extracción de numerosas combinaciones de características de las señales de audio; dentro de éstas, las más populares para el reconocimiento de emociones son: tono, energía, los Coeficientes Cepstrales de las Frecuencias de Mel (MFFCs, por sus siglas en inglés), los Coeficientes Dinámicos de Energía de Mel (MEDC, por sus siglas en inglés) y los formantes [2]–[11].

Gran parte de las investigaciones relacionadas al reconocimiento de emociones a través de la voz, no son rigurosas en las pruebas que le hacen a los modelos entrenados, alcanzando una tasa de reconocimiento considerablemente elevada, no obstante, estos resultados suelen estar sobre-entrenados y muy pocas veces el modelo es sometido a pruebas utilizando un conjunto más amplio. Dado que el objetivo principal es reconocer emociones en un amplio grupo de personas, muchas consideraciones deben tomarse, principalmente con los criterios utilizados para construir la base de datos de entrenamiento. Esto se debe, a que las bases de datos presentan diversas problemáticas con respecto a la diversidad de las expresiones humanas y las diversas características asociadas a la población utilizada para grabar las emociones. A pesar de que existen bases de datos orientadas al reconocimiento de emociones en español [21]–[23]; en esta investigación, se diseñará y se establecerán diferentes criterios de construcción para 3 bases de datos distintas. Esto, en aras de mantener un mayor control sobre las condiciones de ambiente y de las características asociadas a la población que formará parte del proceso de grabación. Estas bases de datos permitirán probar con rigurosidad los modelos entrenados, y de esta forma realizar una comparativa entre el desempeño de los algoritmos de aprendizaje de máquina seleccionados. En concordancia con [3] se utilizarán MFCCs y la energía, como parte de las características que se extraen del conjunto de datos. Además, se utilizarán 3 algoritmos de aprendizaje supervisado distintos: Bosques Aleatorios (RF, por sus siglas en inglés), Aumento del Gradiente (GB, por sus siglas en inglés) y Máquinas de Soporte Vectorial (SVM, por sus siglas en inglés).

El documento se organiza de la siguiente manera: la segunda sección es una descripción de los antecedentes que se usaron como parte de la investigación; la tercera sección explica de manera breve los procesos involucrados en la clasificación de las emociones (construcción de las bases de datos, procesamiento de datos y entrenamiento); la cuarta sección muestra los resultados; la quinta sección presenta una discusión; y la sección final muestra las conclusiones y trabajos futuros de esta investigación.

II. Antecedentes

Uno de los contenidos de mayor disponibilidad y de mayor uso en la actualidad, son los archivos de audio. La voz, es el principal medio de comunicación en los seres humanos y como componente para-verbal de la comunicación [24], se considera que contiene mucha información sobre el estado emocional de la persona que la emite. Las investigaciones relacionadas al reconocimiento de emociones mediante la voz, se basan en la extracción de características del audio para obtener una representación matemática. Ésta, es utilizada para entrenar los algoritmos de aprendizaje de máquina y de esta manera realizar clasificaciones. Gran parte de las investigaciones se basan en la precisión o tasa de reconocimiento de los clasificadores, enfocándose en 3 aspectos: algoritmos de aprendizaje de máquina utilizados, características extraídas del audio y bases de datos empleadas. Los antecedentes de esta investigación se dividen en dos: reconocimiento de emociones, haciendo énfasis en los algoritmos de aprendizaje de máquina, características extraídas, bases de datos y tasas de reconocimiento; y bases de datos orientadas al reconocimiento de emociones, haciendo énfasis en los criterios utilizados para su construcción y aspectos relevantes adicionales.

A. Reconocimiento de Emociones

En la investigación [25], utilizan 6 tipos de clasificadores para comparar la tasa de reconocimiento en la predicción de las 6

emociones universales (ira, sorpresa, felicidad, miedo, tristeza y asco) [26], utilizando como entrada la voz. La base de datos utilizada en esa investigación se llama eNTERFACE'05 [19], la cual es una base de datos audio-visual, que contiene muestras de las 6 emociones mencionadas anteriormente. Cada video es convertido en formato WAV utilizando la herramienta MATLAB. Las características extraídas del audio fueron las siguientes: los Coeficientes Cepstrales de la Frecuencia de Mel (MFCCs, por sus siglas en inglés), coeficientes de Predicción Lineal Cepstral (LPC, por sus siglas en inglés), método de los momentos, segundo método de los momentos, centroide espectral, punto de caída espectral, flujo espectral, compacidad, variabilidad del centroide espectral, media cuadrática, fracción del marco de baja energía, tasa de cruces por cero, frecuencia máxima mediante la tasa de cruces y la transformada discreta de Fourier. Los algoritmos de aprendizaje supervisado utilizados fueron los siguientes: SVM lineal y polinomial, árboles de decisión, redes neuronales, redes bayesianas, algoritmo de los k-vecinos más cercanos y Bayes ingenuo. Los resultados obtenidos mostraron que el árbol de decisión obtuvo la mejor tasa de reconocimiento la cual fue de 96.21%.

Uno de los clasificadores de mayor popularidad en el área de reconocimiento de emociones a través de la voz es el SVM. La principal motivación se debe a que el SVM ha demostrado ser uno de los algoritmos que mayor tasa de reconocimiento tiene cuando se trata de pruebas dependientes e independientes de la persona. En la investigación [6], se utiliza el clasificador SVM con 4 núcleos distintos. El conjunto de datos utilizado es la base de datos Pocala [18] y las emociones utilizadas fueron: ira, miedo, felicidad, tristeza y aburrimiento. Las características seleccionadas fueron: el tono, los formantes, la tasa de cruces por cero, los MFFCs, y parámetros estadísticos. Se utilizaron los núcleos: lineal, cuadrático, radial y polinomial. Los resultados obtenidos mostraron que el SVM con núcleo radial obtuvo la mejor tasa de reconocimiento del 84%. Como conclusión se obtuvo que los núcleos lineal y cuadrático tienen una mejor tasa de reconocimiento en las emociones: ira, miedo y tristeza. A diferencia, el núcleo polinomial tiene la peor tasa de reconocimiento.

Un factor que influye en el desempeño del algoritmo son los hiper-parámetros. En la investigación [5], utilizan la base de datos Emo-DB [17]. Se utilizan 5 emociones: la ira, la tristeza, la alegría, la neutralidad y el miedo. Las características seleccionadas del audio son: los MFCCs y los Coeficientes del Espectro Dinámico de la Energía de Mel (MEDC, por sus siglas en inglés). La tasa de reconocimiento obtenida utilizando un clasificador SVM con núcleo radial, fue de 93.75%. En comparación a la investigación [3], a pesar de que las mismas bases de datos fueron usadas, el cambio de las características y una selección óptima de los hiper-parámetros, produjo que el SVM radial obtuviera una mejor tasa de reconocimiento con respecto a los árboles de decisión. Investigaciones han explorado el uso de diferentes clasificadores como: SVM con núcleo de función base radial (RBF-SVM, por sus siglas en inglés), SVM con núcleo gaussiano (GSV, por sus siglas en inglés), modelo oculto de Márkov (HMM, por sus siglas en inglés), Bosques Aleatorios (RF, por sus siglas en inglés), aumento del gradiente (GB, por sus siglas en inglés), modelo de red neuronal de McCulloch y Pits (MCP-NN, por sus siglas en inglés), perceptrón multicapa (MLP, por sus siglas en inglés), red neuronal probabilística (PNN, por sus siglas en inglés), entre otros. El número de clases y la base de datos varía según la investigación. En la Tabla I se muestra de manera sintetizada los resultados de los trabajos relacionados.

Tabla I: Tasa de Reconocimiento para Diferentes Clasificadores

Ref.	Modelo	N°clases	% Exactitud	Base de Datos
[2]	GSVM	4	67.1%	Susas [27]
[2]	HMM	4	70.1%	Susas [27]
[2]	HMM	2	96.3%	Susas [27]
[2]	GSVM	5	42.3%	Aibo [28]
[3]	Rand-SVM	7	55.89%	Emo-DB [17]
[3]	RF	7	81.05%	Emo-DB [17]
[3]	GB	7	65.23%	Emo-DB [17]
[4]	MCP NN	2	85%	Sin nombre
[5]	RBF-SVM	5	93.75%	Emo-DB [17]
[6]	RBF-SVM	6	84%	Polish-DB [18]
[7]	MLP	7	83.1%	Emo-DB [17]
[7]	RF	7	77.19%	Emo-DB [17]
[7]	PNN	7	94.1%	Emo-DB [17]
[7]	SVM	7	83.1%	Emo-DB [17]
[8]	RBF-SVM	7	86.6%	Emo-DB [17]
[11]	SVM	3	91.30%	Emo-DB [17]
[11]	SVM	3	95.09%	SJTU-DB [11]

En esta investigación se usarán 2 de las características más populares en las investigaciones actuales: la Energía y los Coeficientes Cepstrales de Mel. Bajo el interés de realizar una comparativa del desempeño de los clasificadores sobre muestras en diferentes condiciones, se utilizarán 3 algoritmos de aprendizaje de máquina: Bosques aleatorios (RF), Máquinas de Vectores de Soporte (SVM) y Aumento del Gradiente (GB).

B. Bases de Datos

A pesar de que existen muchas investigaciones que logran una gran precisión reconociendo emociones [5], [7], [11], en la práctica estos clasificadores no suelen tener el mismo desempeño. Esto se debe a muchos factores, tales como las diferencias que existen entre las personas que forman parte del conjunto de entrenamiento y el conjunto de

prueba, por ejemplo: las condiciones de ambiente, los dispositivos utilizados, diferencias culturales, el idioma, la edad, entre otros. Existen otros factores que afectan la tasa de reconocimiento en las emociones, por ejemplo el desbalance en las bases de datos, la calidad de las emociones capturadas, la diversidad de frases, el número de emociones, entre otros. Muchos son los criterios utilizados para diseñar una base de datos orientada al reconocimiento de emociones, entre las características más relevantes se encuentran: número de personas, origen de las personas, idioma utilizado, frases utilizadas, entre otros. A continuación se presentan las bases de datos más populares orientadas al reconocimiento de emociones.

En la investigación [17], se presenta una base de datos conocida como Emo-DB. Esta base de datos, fue construida utilizando 10 actores (5 mujeres y 5 hombres), simulando o actuado emociones. Las frases seleccionadas son 10 (5 cortas y 5 largas), usadas diariamente e interpretables en todas las emociones aplicadas. Las grabaciones fueron realizadas en una cámara anecoica, con equipo de grabación de alta calidad. La base de datos consistió en 800 frases, en las cuales están contenidas 7 emociones: neutralidad, ira, miedo, alegría, tristeza, asco y aburrimiento. La base de datos fue evaluada mediante una prueba de percepción con respecto a su reconocibilidad y su naturalidad. Las frases que fueron reconocidas con un porcentaje mayor al 80% y juzgadas con un porcentaje mayor al 60% como natural fueron seleccionadas y etiquetas. Para mejorar la calidad de las muestras, se utilizaron diferentes audios para ayudar a los actores a reproducir cada una de las emociones. Una de las características más notorias de esta base de datos, es que las 10 frases son expresadas para las 7 emociones distintas, y aunado a eso, estas frases son interpretables en cada uno de estos casos.

Existen muchas técnicas desarrolladas para reconocer emociones, una de las formas de aumentar la capacidad de reconocer emociones, es mediante el uso de información multimodal. Estos algoritmos utilizan información de diferentes canales de entrada como: la voz y la imagen del rostro; para mejorar la capacidad de reconocimiento de los clasificadores. En la investigación [19], se presenta una base de datos audio-visual conocida como eNTERFACE'05, cuyo propósito es la evaluación de algoritmos de reconocimiento de emociones (unimodal y multimodal). Para reproducir las emociones a cada uno de los participantes se les pidió escuchar 6 historias sucesivas, cada una de ellas evocando una emoción en particular. Cada uno de ellos debía reaccionar en su propio idioma a cada una de las situaciones mientras eran grabados, luego, dos jurados detallaban si el sujeto reaccionaba de manera auténtica, y según este criterio se añadía la muestra a la base de datos. No obstante, la distribución geográfica de las personas que fueron parte de la base de datos era muy dispersa y debido a esto, las características como las variaciones del tono y la tasa del habla no eran comunes entre los participantes. Por lo tanto, se tomó la decisión de realizar el mismo protocolo pero reaccionando en inglés. En el segundo protocolo se tomó la decisión de predefinir las respuestas ante los distintos escenarios de las historias, debido a que cuando los actores reaccionaron libremente a cada uno de los escenarios no se expresaron de una manera completamente espontánea. El protocolo final se realizó de la siguiente manera: cada sujeto escuchaba una pequeña historia por cada emoción para intentar entrar en el contexto del escenario, luego el sujeto reaccionaba mediante cada una de 5 frases predefinidas. La base de datos consistió en 1166 secuencias de video, de las cuales 264 eran constituidas por mujeres y 902 por hombres. Una de las características más notorias de esta base de datos es la distribución geográfica de las personas que fueron parte de la misma, además, esta base de datos está constituida por 5 frases diferentes por emoción.

Una de las dificultades más comunes en la construcción de bases de datos es la captura de emociones auténticas. Aunque gran parte de las bases de datos utilizan emociones actuadas o simuladas, existe un gran esfuerzo por validar la reconocibilidad y naturalidad de cada una de las frases que las conforman. En la investigación [18], se presenta una base de datos polaca, la cual esta conformada por muestras extraídas de discusiones naturales en programas de televisión. Las frases utilizadas están conformadas por interacciones espontáneas, y además, provee un amplio rango de emociones básicas y complejas. Cada una de las muestras extraídas fueron etiquetadas por un grupo de expertos y voluntarios. La base de datos está constituida de 15 estados emocionales, los cuales se dividen en primarias: ira, anticipación, alegría, miedo, sorpresa, tristeza, asco; y secundarias: rabia, molestia, éxtasis, serenidad, terror, detención, dolor y pensamiento. En total se recolectaron y etiquetaron 784 muestras. La característica más notoria de esta investigación es la espontaneidad de las frases, y además, contiene un rango de emociones amplio y altamente diferenciado.

Numerosas investigaciones han propuesto diferentes tipos de bases de datos para el reconocimiento de emociones, entre ellas: la base de datos SAVEE [20], constituida por 480 muestras de audio con las emociones: ira, asco, sorpresa, alegría, miedo, tristeza y neutralidad. Una de las características más notorias de esta base de datos, es que las emociones son inducidas mediante videos. Existen otros tipos de bases de datos cuyo propósito general no fue la evaluación de algoritmos de aprendizaje para el reconocimiento de emociones, no obstante, son utilizadas para ese fin, entre ellas: la base de datos SUSAS [27], cuyo propósito principal fue el análisis y formulación de algoritmos del reconocimiento del habla en condiciones de ruido y estrés. Otras bases de datos populares como AIBO [28], son construidas a partir de escenarios naturales; en este caso, grabaciones de niños mientras interactúan con un robot. Las grabaciones obtenidas están conformadas por 110 diálogos y 29200 palabras en 11 categorías emocionales de ira, aburrimiento, enfático, indefenso, ironía, alegría, autoridad materna, represión, descanso, sorpresa e irritación. El etiquetado de los datos se basa en el juicio de los oyentes. Adicionalmente, existen otras bases de datos cuyo propósito es realizar análisis sentimental como [29], donde se presenta

una base de datos de videos en español conformada por 105 muestras etiquetadas mediante su polaridad: positiva o negativa.

III. Método

En las investigaciones sobre algoritmos para el reconocimiento de estados emocionales mediante la voz, son imprescindibles tres aspectos: la base de datos, el procesamiento de los datos y el entrenamiento. La base de datos está constituida por el conjunto de muestras de audio que serán parte del entrenamiento del algoritmo de clasificación, cuya calidad y diversidad de las muestras se relaciona directamente con la tasa de precisión del algoritmo. El procesamiento de los datos se basa en la selección de las características del audio apropiadas que permitirán representar las muestras matemáticamente; el procesamiento y la selección correcta de características influye directamente en la tasa de reconocimiento. Por otro lado, el entrenamiento consiste en utilizar la base de datos para entrenar el algoritmo de aprendizaje de máquina seleccionado. De acuerdo a las características de las muestras, algunos algoritmos permiten realizar una mejor clasificación. Por esta razón, en esta investigación se realizará una comparativa de los resultados obtenidos de cada uno de los algoritmos de aprendizaje de máquina seleccionados en cada uno de los experimentos realizados.

A. Construcción de las Bases de Datos

Las bases de datos emocionales de audio, pueden ser clasificadas en tres tipos según la forma en que se pide a las personas demostrar las emociones [6]:

- **Lenguaje actuado**: Se pide a los actores expresar directamente una emoción predefinida.
- **Lenguaje de la vida real**: Respuestas naturales de conversaciones, las cuales son auténticas por naturaleza.
- **Lenguaje emocional evocado**: Las emociones son inducidas y son auto-reportadas en lugar de ser etiquetadas, es decir, la persona reconoce su propia emoción y le asigna por si mismo una etiqueta.

Entre las bases de datos que se basan en lenguaje de la vida real se tiene: “Polish Emotional Natural Speech Database” [18] y “Automatic Classification of Emotion-Related User States in Spontaneous Children Speech” [28]. Basada en lenguaje actuado: “A Database of German Emotional Speech” [17]; y basada en la evocación de emociones: “The eNTERFACE’05 audio-visual emotion database” [19] y “Surrey Audio-Visual Expressed Emotion (SAVEE) database” [20].

Actualmente, la mayoría de investigaciones relacionadas al reconocimiento de emociones no son rigurosas con el tipo de pruebas que hacen a los modelos. Por esta razón, y en aras de mantener un mejor control sobre las condiciones de ambiente, en esta investigación se construirán 3 bases de datos orientadas al reconocimiento de emociones, una bajo condiciones controladas y actuadas, otra en condiciones controladas y semi-natural, y finalmente, otra en condiciones no controladas y naturales.

1) Bases de Datos en Condiciones Controladas y Actuadas: Para realizar la construcción de esta base de datos se realizó la selección del conjunto de frases por cada una de las emociones como en [19], las cuales fueron sometidas a 3 tipos de validación: validación de las frases de forma textual, validación por parte de los participantes del proceso de grabación y validación por parte de un jurado de 4 personas. La validación de las frases de forma textual, se realizó mediante una encuesta en las cuales participaron 96 personas (66 hombres y 30 mujeres). A cada una de las personas se les presentó un conjunto de frases por cada una de las emociones: ira, sorpresa, felicidad, tristeza, asco y miedo. La encuesta consistía en seleccionar aquellas frases con las cuales expresarían cada una de las emociones. Las frases con las cuales se sintieron identificados gran parte de los participantes de la encuesta fueron: ira, tristeza, felicidad, miedo y sorpresa. Las frases utilizadas para expresar el asco fueron las menos seleccionadas. Adicionalmente, se les pidió a los participantes sugerir qué frases utilizarían ellos para expresar cada emoción. Luego cada una de las frases, junto con las sugerencias de los participantes fueron seleccionadas por 2 jueces y el conjunto de frases resultante se presenta en la Tabla II.

El proceso de grabación se realizó en una oficina, con poco ruido. Adicionalmente, todos los participantes fueron ubicados en un mismo sitio para grabar, a una distancia de 40 centímetros del micrófono. El proceso de grabación fue realizado de la siguiente manera:

- A cada uno de los participantes se les pidió sentarse en una silla ubicada a 40 centímetros del micrófono.
- A cada participante se le pidió leer el conjunto de frases de cada emoción, luego, se le pidió reproducir (actuar) cada una de las frases de cada emoción 4 veces de distintas maneras.
- En caso de no expresar correctamente alguna declaración o de que el participante no estuviese satisfecho con el resultado, se le pedía al participante repetir dicha emoción utilizando como ayuda la orientación del operador o muestras de participantes anteriores.
- Para realizar las grabaciones fue utilizado el software Audacity [30]. Se utilizó un solo canal de grabación y la frecuencia de muestreo fue de 48 kHz.
- Luego del proceso de grabación cada una de las frases fue seleccionada por el operador, donde se descartaron aquellas frases contaminadas (ruidos de golpes de mesa, movimientos de sillas, entre otros) o de poca calidad (frases incompletas o ambiguas).
- Luego de seleccionadas las muestras, se recortaron cuidadosamente y se transformaron a 16 kHz. Adicionalmente, cada una fue etiquetada.

La validación por parte de los participantes de la base de datos, consistió en reproducir cada una de las frases del participante y hacerle dos preguntas por cada una de ellas: ¿Considera que en esta declaración se expresó la emoción? y ¿Considera que esta declaración pudiera interpretarse de otra manera?. Si algunas de las dos preguntas anteriores eran respondidas de manera negativa, se descartaba la muestra. En el caso particular en donde se descartaban todas las muestras de una

Tabla II: Conjunto de Frases de cada Emoción

Ira	1) ¿Qué te pasa? 2) !Eso a mi que me importa! 3) ¡O te vas o te boto! 4) ¿Me vas a atender o no? 5) ¿Sabes qué? !Déjalo así! 6) No me molestes!
Sorpresa	1) ¡No puede ser! ¿En serio? 2) ¡Qué! ¡Yo no sabía eso! 3) ¡Jamás lo hubiera creído! 4) ¡No me lo esperaba! 5) ¡No te creo! ¿De verdad? 6) ¿De verdad? ¡No sabía! 7) ¿Es en serio?
Felicidad	1) ¡Gané! 2) ¡Que genial! Pasé! 3) ¡No me lo creo! ¡qué suerte! 4) ¡Lo logré! ¡Al fin! 5) ¡No puede ser! ¡qué bien! 6) ¡No lo creo! ¡Funciona!
Miedo	1) No, no me hagas daño 2) Ya no tengo más, no tengo nada. 3) No, no me robes 4) Aléjate, Aléjate 5) Aléjate por favor 6) no, por favor
Asco	1) Esto si está feo! 2) ¿Qué hay en el plato? 3)¡Qué repugnante! 4) ¿Qué asco? ¿Qué es esto? 5) ¡Un bicho! 6) ¿Qué es esto?
Tristeza	1) Todo iba tan bien, no sé qué pasó 2) Lo/La extraño pero se fue 3) Ya no será lo mismo 4) Dime que no es verdad 5) Aún sentía algo por ella 6) El/Ella fue parte de mi vida 7) No pude hacerlo

frase, se repetía el proceso de grabación.

Finalmente, la validación por parte de un jurado consistió en cuantificar la validez del contenido mediante la "V" de Aiken. El número de jurados fue 4. A cada uno se le pidió calificar cada una de las muestras previamente filtradas por las validaciones anteriores según las preguntas de la escala presentada en la Tabla III.

Luego de realizar el etiquetado en base a la escala anterior, se obtuvo el coeficiente "V" de Aiken para cada una de las muestras.

$$V = \frac{S}{(n(C-1))} \tag{1}$$

En la ecuación 1, el valor S, representa la suma de los valores

Tabla III: Escala Utilizada para la Validación de las Muestras de Audio

Significado	Valor
El audio es entendido e interpretado inequívocamente de una única manera	**3**
Para algunas personas podría tener otro significado	**2**
El audio es susceptible de ser entendido en sentidos diversos	**1**
El audio definitivamente se presta para múltiples interpretaciones.	**0**

de cada jurado por cada muestra. El valor n, representa el número de personas en el jurado, y el valor C, el número de valores en la escala de valoración. Las muestras cuyo cálculo de validación fue mayor a 0.75 fueron aceptadas y formaron parte de la base de datos. El resto de muestras fue descartado.

La base de datos en condiciones controladas y actuadas, fue conformada por un total de 1351 muestras. La frecuencia de las muestras por emoción puede verse en la Figura 1.

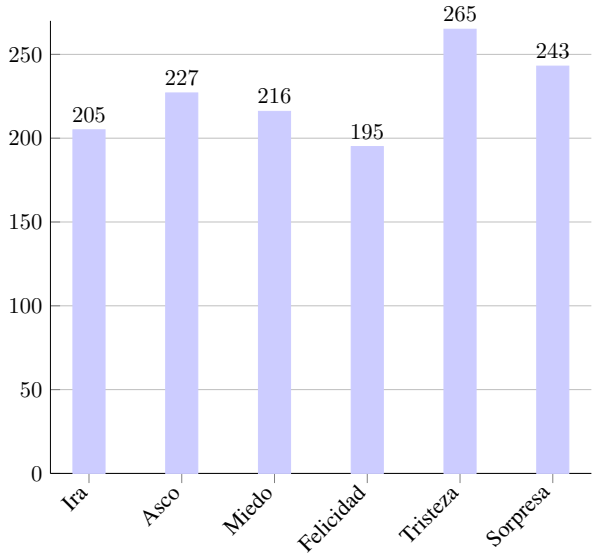

Figura 1: Frecuencia de las Emociones de la Base de Datos en Condiciones Controladas y Actuadas

2) Bases de Datos en Condiciones Controladas y Semi-naturales: Para crear un conjunto que permitiera probar de manera rigurosa los modelos entrenados con la base de datos anterior, se realizó una base de datos en las mismas condiciones pero variando las palabras utilizadas inicialmente. De manera similar a la base de datos anterior, fueron realizados 3 tipos de validación.

Bajo la intención de determinar la capacidad de los modelos para reconocer un amplio grupo de expresiones en las personas, esta base de datos consistió en expresar cada una de las frases de cada emoción utilizando sus propias palabras. Es decir expresando el mismo significado y la emoción de la frase original pero utilizando las palabras que utilizaría el participante en vida cotidiana.

Todo el proceso de grabación fue similar al anterior, el único

cambio que se realizó, se basó en que los participantes debían utilizar sus propias palabras para expresar 4 veces cada una de las frases de cada emoción (ver Tabla II).

La base de datos en condiciones controladas y semi-naturales, fue conformada por un total de 1163 muestras. La frecuencia de las muestras por emoción puede verse en la Figura 2.

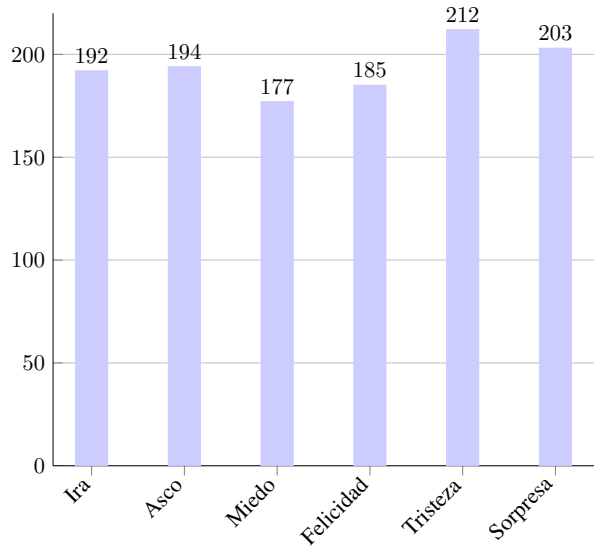

Figura 2: Frecuencia de las Emociones de la Base de Datos en Condiciones Controladas y Semi-naturales

3) Bases de Datos en Condiciones no Controladas y Naturales: La base de datos en condiciones no controladas, cuenta con 105 muestras, 70 muestras provenientes de videos de Internet, y 30 muestras provenientes de 3 personas (1 hombre y 2 mujeres). El proceso de grabación fue realizado de la siguiente manera:

- A cada participante se le muestran segmentos de audio que usarán de guía para reproducir la emoción.
- Todos los segmentos de audio contienen información sobre una situación en particular.
- No se tomó en consideración la ubicación donde se realizaron las grabaciones.
- Cada participante es grabado mediante un micrófono convencional utilizando la biblioteca PyAudio [31].
- De la grabación se extraen las frases que se consideran naturales.
- Cada segmento de audio tiene un tamaño entre 2 y 6 segundos.
- Cada segmento de audio, se graba con una frecuencia de muestreo de 16 kHz y se almacena en formato wav.

Para realizar la validación de cada una de estas muestras se le pide al participante escuchar las frases seleccionadas y luego se valida la emoción tomando en consideración la opinión de dos jueces y la del participante. Si dos de tres opiniones coinciden, entonces la muestra es etiquetada y luego añadida a la base de datos.

Las muestras de Internet fueron obtenidas y procesadas mediante la biblioteca Youtube-dl [32]. Todas las muestras de audio son en español, principalmente países de América Latina. El proceso de obtención de muestras de Internet fue realizado de la siguiente manera:

- Fueron seleccionados videos en español, conformados por una sola persona sin sonidos musicales de fondo.
- Para cada video fueron registrados los segmentos que se corresponden con una emoción particular.
- Se utilizó el tiempo de inicio y fin, el enlace del video y la etiqueta para obtener el segmento de audio correspondiente mediante la biblioteca youtube-dl [32].
- Todos los segmentos fueron transformados a formato wav con una frecuencia de muestreo de 16 kHz.

Para realizar la validación de cada una de las muestras de Internet, se utilizó la opinión de 3 jueces, de manera similar al proceso anterior, se basó en la opinión de cada uno de ellos.

La base de datos en condiciones no controladas, contiene audios correspondientes a las 6 emociones universales descritas por [26]: ira, miedo, felicidad, asco, tristeza y sorpresa. Adicionalmente, contiene la neutralidad. Si dos de tres opiniones coinciden, entonces la muestra es etiquetada y luego añadida a la base de datos. La base de datos fue conformada por un total de 105 muestras. La frecuencia de cada una de las emociones puede verse en la Figura 3.

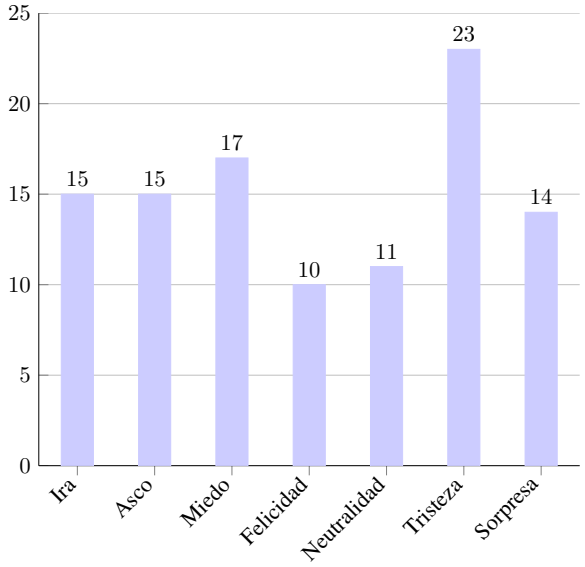

Figura 3: Frecuencia de las Emociones de la Base de Datos en Condiciones no Controladas

B. Procesamiento de los Datos

Los segmentos de audio se procesaron para obtener las características que serán representadas como un vector. Para realizar esto, un proceso de extracción a largo plazo fue llevado a cabo con la ayuda de la biblioteca de análisis de audio PyAudioAnalysis [33].

El proceso de extracción de características a largo plazo, consiste en obtener el promedio de características de mediano plazo que a su vez depende del procesamiento a corto plazo de la señal de audio. Esta forma de procesar el audio también se le conoce como segmental (corto y mediano

plazo) y suprasegmental (largo-plazo) [34]. Para cada audio se utilizaron marcos de 20 mili-segundos en el procesamiento a corto plazo sin solapamiento, y segmentos de 1 segundo para el procesamiento a mediano plazo con solapamiento, para finalmente obtener un vector con 28 estadísticas.

En la Figura 4, se puede observar un esquema que muestra cómo se lleva a cabo el proceso de extracción de características. En la primera fase del procesamiento (análisis a corto plazo), se obtienen las características $c1, c2, ...c_N$ de cada marco $(m_1, m_2, ..., m_b)$; en la segunda fase del proceso (análisis a mediano plazo), se extraen estadísticas de las características particulares de cada uno de los marcos del bloque, en este caso el promedio μ_N y la desviación estándar σ_N; por último, se realiza un procesamiento suprasegmental (análisis a largo plazo), cuyo objetivo es obtener el promedio $\mu_{2N}^{'}$ de las estadísticas de la fase anterior.

Las señales de audio, y en particular aquellas con contenido emocional, se caracterizan por tener un gran número de información. Una de las cosas más importantes en las investigaciones de reconocimiento de emociones a través de la voz, es seleccionar un conjunto de características adecuadas de tal manera que se pueda representar lo mejor posible cada una de las muestras de audio.

Para esta investigación se utilizaron 2 tipos de características: del dominio del tiempo y del dominio cepstral. Todas fueron obtenidas mediante la biblioteca PyAudioAnalysis [33], una descripción formal de las características junto con sus algoritmos puede encontrarse en [35]. A continuación se muestra una descripción de las características que se seleccionaron para esta investigación.

1) Energía o Potencia de la Señal: La energía se define como la suma de los cuadrados de las muestras, que usualmente se normaliza dividendo entre la longitud de la muestra. La energía es la característica más básica en el procesamiento de señales de la voz. Ésta juega un papel importante en el reconocimiento de emociones. Por ejemplo, las emociones como la felicidad o la ira contienen una mayor energía en comparación a la tristeza. Gran parte de las investigaciones la utilizan [2], [3], [7], [8], [10], [11].

Sea $X_i(n), n = 1, ..., W_L$ la secuencia de muestras de audio en el i-ésimo marco, donde W_L es el tamaño del marco. La energía a corto plazo es calculada como sigue:

$$E(i) = \sum_{n=1}^{W_L} |X_i(n)|^2 \tag{2}$$

Usualmente la energía es normalizada dividiéndola por el tamaño del marco W_L para remover la dependencia de la longitud del marco, quedando el cálculo de la siguiente manera:

$$E(i) = \frac{1}{W_L} \sum_{n=1}^{W_L} |X_i(n)|^2 \tag{3}$$

2) MFFCs: los coeficientes cepstrales de las frecuencias de Mel han sido muy populares en el campo del análisis de voz. En la práctica, los MFCCs son los coeficientes discretos de la transformación coseno del espectro de potencia logarítmica en la escala de Mel. Los MFCCs han sido ampliamente utilizados en el reconocimiento de voz, agrupamiento de altavoces, reconocimiento de emociones y muchos otros tipos de aplicaciones de análisis de audio y aprendizaje de máquina. Caracterizan la magnitud del espectro y por lo general son usados los 12 primeros coeficientes. En la gran mayoría de investigaciones los MFFCs han mostrado ser la característica que mejores cualidades tiene para el reconocimiento de emociones [2], [3], [5]–[8], [10], [11].

Para extraer los coeficientes cepstrales de las frecuencias de Mel de un marco, son necesarios los siguientes pasos:

(a) La transformada discreta de Fourier (DFT, por sus siglas en inglés) es calculada. Esta es usada para derivar la representación de la señal en el dominio de la frecuencia (espectral), cuyo propósito es servir como entrada para la obtención de muchas otras características importantes.

Dada una señal discreta en el dominio del tiempo $x(n), n = 0, ..., N-1$, con N muestras de longitud, su DFT es calculada como sigue:

$$X(k) = \sum_{n=0}^{N-1} x(n) exp(-j\frac{2\pi}{N}kn),\ k = 0, ..., N-1, \tag{4}$$

(b) El espectro resultante es utilizado como entrada a un banco de filtros de la escala de Mel que consiste en L filtros. Los filtros usualmente tienen una frecuencia triangular superpuesta. La escala de Mel introduce una función de distorsión de frecuencia que intenta ajustarse a ciertas observaciones psicoacústicas. A través de los años varias funciones de distorsión de frecuencia han sido propuestas por ejemplo:

$$f_w = 2595 * log(1 + f/700) \tag{5}$$

Si $\tilde{O_k}, k = 1, ..., L$, es la potencia en la salida del k-ésimo filtro, entonces los MFCCs están dados por la siguiente ecuación

$$C_m = \sum_{k=1}^{L} (log\tilde{O_k}) cos[m(k - \frac{1}{2})\frac{\pi}{L}],\ m = 1, ..., L. \tag{6}$$

En total se genera un vector de 14 características $(c_1, c_2, c_3, ..., c_{14})$ por cada marco (1 valor correspondiente a la energía y 13 coeficientes de Mel), que será usado para generar un vector de una dimensión igual a 28 $(\mu_{2N}^{'})$, cuyos elementos corresponden al promedio μ y desviación estándar σ de las 14 características obtenidas mediante el procesamiento a largo plazo.

C. Entrenamiento

Muchos algoritmos de aprendizaje de máquina han sido utilizados en diferentes investigaciones sobre el reconocimiento de emociones a través del audio. Una lista de diferentes algoritmos y su desempeño se puede ver en la Tabla I. Dado que el objetivo de esta investigación es

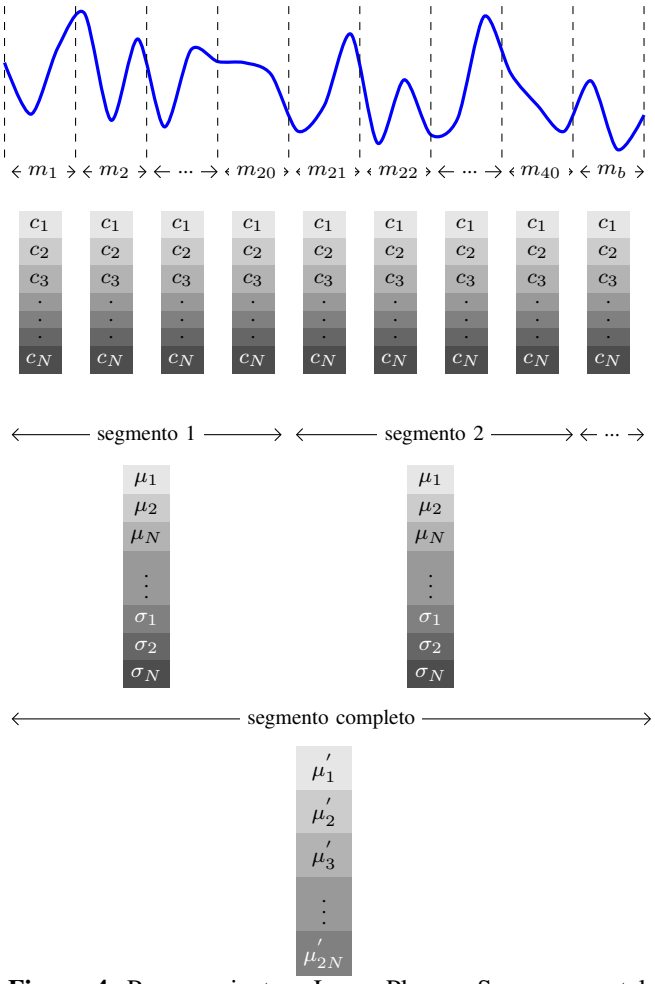

Figura 4: Procesamiento a Largo Plazo o Suprasegmental del Audio

reconocer emociones en un amplio grupo de personas, las pruebas realizadas están orientadas a estimar la capacidad de generalización que poseen los algoritmos con los datos de entrenamiento. Es por esto, que se realizan diferentes pruebas de retención, donde los conjuntos de entrenamiento y prueba se encuentran separados.

En esta investigación se realizará un modelo por cada uno de los algoritmos de aprendizaje de máquina utilizados (SVM, GB y RF). Cada uno de estos modelos será entrenado con el 70% de las muestras pertenecientes a la base de datos en condiciones controladas y actuadas. Posteriormente, se realizarán 3 tipos de pruebas a cada uno de los modelos:

1) P1: Pruebas utilizando el 30% restante de la base de datos en condiciones controladas y actuadas.
2) P2: Pruebas utilizando toda la base de datos en condiciones controladas y semi-naturales.
3) P3: Pruebas utilizando toda la base de datos en condiciones no controladas, a excepción de la neutralidad.

El proceso de clasificación utilizará los vectores provenientes del módulo de extracción de características para su entrenamiento y prueba. En la Figura 5 se puede observar el diagrama del proceso de clasificación. Se utilizarán 3 tipos de clasificadores: bosques aleatorios, Aumento del Gradiente y Máquinas de Vectores de Soporte. La biblioteca Scikit-learn [36], es utilizada para la implementación.

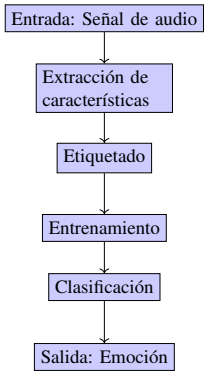

Figura 5: Diagrama del Proceso de Clasificación

IV. Resultados

A continuación se presentan los resultados obtenidos mediante cada una de las pruebas mencionadas anteriormente.

Tabla IV: Resultados de las Tasas de Reconocimiento para cada una de las Pruebas

	P1	**P2**	**P3**
SVM	83%	68%	17%
GB	79%	66%	27%
RF	79%	67%	23%

En la Tabla IV, se pueden apreciar las tasas de reconocimiento de cada uno de los modelos implementados en cada una de las pruebas. En este caso se puede apreciar que el modelo SVM, obtuvo la mejor tasa de reconocimiento en las pruebas 1 y 2, mientras que el modelo GB obtuvo mejores resultados en la prueba 3.

A continuación se presentan los resultados individuales de las tasas de reconocimiento de cada emoción de manera individual para cada uno de los modelos.

Tabla V: Resultados de las Tasas de Reconocimiento para cada una de las Emociones Utilizando el Modelo SVM

	Ira	**Tristeza**	**Asco**	**Felicidad**	**Sorpresa**	**Miedo**
P1	82%	89%	81%	84%	78%	**94%**
P2	52%	**87%**	69%	60%	56%	75%
P3	**57%**	0%	0%	16%	0%	20%

Tabla VI: Resultados de las Tasas de Reconocimiento para cada una de las Emociones Utilizando el Modelo GB

	Ira	**Tristeza**	**Asco**	**Felicidad**	**Sorpresa**	**Miedo**
P1	75%	84%	70%	83%	73%	**90%**
P2	64%	**87%**	70%	55%	49%	69%
P3	**80%**	0%	50%	14%	0%	12%

Tabla VII: Resultados de las Tasas de Reconocimiento para cada una de las Emociones Utilizando el Modelo RF

	Ira	Tristeza	Asco	Felicidad	Sorpresa	Miedo
P1	89%	80%	74%	79%	77%	**92%**
P2	65%	**90%**	70%	54%	53%	64%
P3	**53%**	0%	60%	0%	0%	18%

En las Tablas V, VI y VII, se pueden apreciar los porcentajes de reconocimiento obtenidos para cada modelo en cada una de las pruebas realizadas. El miedo y la tristeza, fueron las emociones que mejor se reconocieron en cada uno de los modelos para las pruebas 1 y 2; la sorpresa fue la emoción que menor tasa de reconocimiento obtuvo en las pruebas 1 y 2. Adicionalmente, la tristeza y la sorpresa no fueron reconocidas en ninguno de los modelos para la prueba 3.

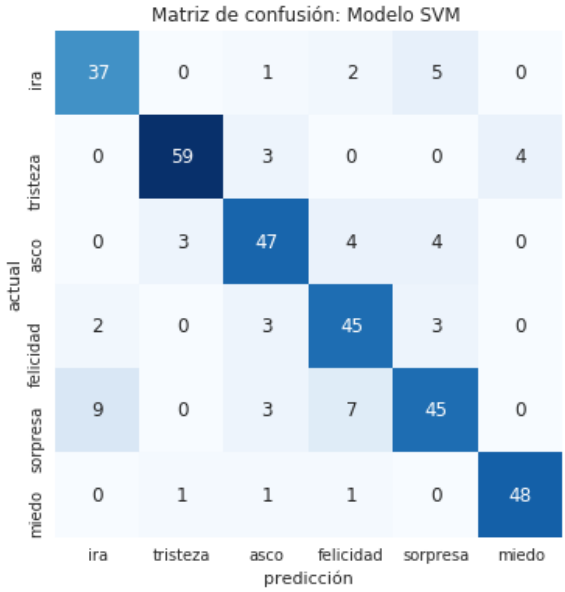

Figura 6: Matriz de Confusión: Resultados del Modelo SVM en la Prueba 1. Utilizando el Subconjunto de Muestras Actuadas en Condiciones Controladas

V. Discusión

Las emociones humanas pueden ser expresadas de diversas maneras, por lo que registrar un conjunto representativo de este rango es una tarea realmente compleja. Ésto, limita la capacidad de los modelos para reconocer emociones y por consiguiente, limita la capacidad de las aplicaciones robóticas que hacen uso de estos modelos. Es por esta razón, que en esta investigación se realizaron 3 bases de datos distintas, para probar rigurosamente la capacidad de los modelos para reconocer emociones en un amplio rango de expresiones y personas.

Los resultados obtenidos mostraron que el algoritmo Máquinas de Vectores de Soporte con núcleo radial obtuvo la mejor tasa de reconocimiento, 83% y 68% en las pruebas 1 y 2 respectivamente; en las Figuras 6 y 7 se pueden apreciar las

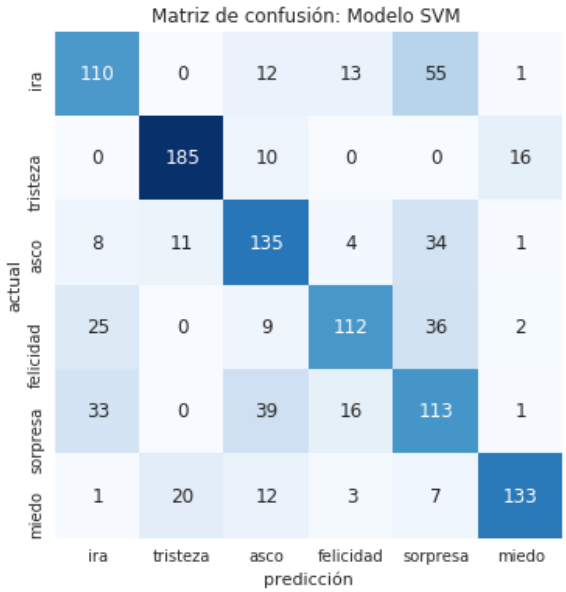

Figura 7: Matriz de Confusión: Resultados del Modelo SVM en la Prueba 2. Utilizando el Subconjunto de Muestras Semi-naturales en Condiciones Controladas

respectivas matrices de confusión. Los resultados obtenidos en las pruebas 1 y 2 para cada una de las emociones fue mayor al 52%, lo que se considera un buen resultado tomando en cuenta la cantidad de muestras en cada una de las pruebas que se realizaron. Adicionalmente, la emoción que mejor se reconoció en la prueba 3 fue la ira.

En el caso del modelo Aumento de Gradiente, los resultados obtenidos para estas dos pruebas, fueron mayores al 49% en las pruebas 1 y 2. Para la prueba 3, este modelo fue el que mejor reconoció la ira, alcanzando un porcentaje de reconocimiento del 80%. En caso del modelo Bosques Aleatorios, los resultados obtenidos en las pruebas 1 y 2 fueron mayores al 53%, lo que supera a los modelos SVM y GB descritos anteriormente. Para la prueba 3, este modelo fue el que mejor reconoció el asco, con un porcentaje de reconocimiento del 60%.

Las emociones que mejor se reconocieron en todas las pruebas, fueron el miedo y la tristeza. Para el miedo, el mejor resultado lo obtuvo el modelo SVM con un porcentaje de reconocimiento del 94%, mientras que en la tristeza, el mejor resultado lo obtuvo el modelo RF con un porcentaje de reconocimiento del 90%. En el caso de la tristeza, se puede atribuir este resultado a que esta emoción es la que tiene mayor número de muestras en las bases de datos con condiciones controladas (ver Figuras 1 y 2). No obstante, el miedo es la emoción que menor número de muestras tiene en la base de datos en condiciones controladas y semi-naturales, por lo tanto, este resultado se puede atribuir a que el miedo fue expresado de manera muy consistente tanto en la base de datos en condiciones controladas y actuadas (prueba 1) como en la base de datos en condiciones controladas y semi-naturales.

En el caso de la base de datos en condiciones no controladas, se pudo observar que se obtuvieron los peores resultados,

incluso algunas emociones como la tristeza y la sorpresa no fueron reconocidas por ningún modelo. A pesar de que estas muestras están correctamente validadas, este resultado puede ser atribuido a que una emoción puede ser expresada de distintas maneras y en diferentes intensidades como en [18]. Por lo tanto, bajo la intención de reconocer un amplio grupo de emociones en las personas, es necesario entrenar los modelos con un grupo altamente representativo con todas las variaciones que involucran la expresión de una emoción, concretamente mediante la voz.

A pesar de que en la prueba 3 se obtuvieron las peores tasas de reconocimiento (condiciones no controladas), estas pruebas permitieron descubrir que en el caso particular de la ira, en todos los modelos fue la emoción cuya tasa de reconocimiento fue mayor. Esto puede atribuirse a que a diferencia de otras emociones, la ira es expresada de una manera muy consistente entre todas las personas.

VI. Conclusiones

En esta investigación se construyeron 3 bases de datos orientadas al reconocimiento de emociones en español. Una de ellas en condiciones controladas y actuadas, otra en condiciones controladas y semi-naturales; y finalmente una en condiciones no controladas. La base de datos de emociones en condiciones controladas y actuadas fue utilizada para entrenar los algoritmos de aprendizaje de máquina seleccionados para esta investigación. Adicionalmente, se realizaron 3 pruebas distintas para probar la capacidad de reconocimiento de los modelos, utilizando muestras actuadas, semi-naturales y naturales. Como se puede apreciar en los resultados de la Tabla IV, los mejores resultados se obtienen cuando se realizan pruebas sobre muestras capturadas en las mismas condiciones que las muestras de entrenamiento. A medida que las pruebas realizadas se salen del marco del conjunto de entrenamiento del modelo, éste disminuye su capacidad de reconocimiento. Considerando los resultados obtenidos, se puede concluir que generalizar un rango de emociones es una tarea realmente compleja, incluso cuando se varían condiciones pequeñas, por ejemplo, en el caso de la base de datos con muestras semi-naturales, los resultados disminuyen considerablemente. Finalmente, trabajos futuros se orientan en la construcción de una base de datos con un rango más amplio de emociones y diferentes niveles de intensidad, así como también, la aplicación de nuevos algoritmos y métodos de procesamiento de audio para el reconocimiento de emociones.

Referencias

[1] R. Picard, *Toward Computers that Recognize and Respond to User Emotion*, IBM Systems Journal, vol. 39, no. 3.4, pp. 705–719, 2000.

[2] O. Kwon, K. Chan, J. Hao, y T. Lee, *Emotion Recognition by Speech Signals*, in Eighth European Conference on Speech Communication and Technology, Geneva, Switzerland, Septiembre 2003.

[3] M. Ghai, S. Lal, S. Duggal, y S. Manik, *Emotion Recognition on Speech Signals using Machine Learning*, in 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), Andhra Pradesh, India, Marzo 2017.

[4] V. Kirandzhiska and N. Ackovska, *Sound Features Used in Emotion Classification*, Ninth International Conference on Informatics and Information Technology, Bitola, Macedonia, Abril 2012.

[5] Y. Chavhan, B. Yelure, y K. Tayade, *Speech Emotion Recognition using RBF Kernel of LIBSVM*, in 2nd International Conference on Electronics and Communication Systems (ICECS), Estados Unidos, Febrero 2015.

[6] P. Dahake, K. Shaw, y P. Malathi, *Speaker Dependent Speech Emotion Recognition using MFCC and Support Vector Machine*, in International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), Pune, India, Septtiembre 2016.

[7] T. Iliou y C. Anagnostopoulos, *Classification on Speech Emotion Recognition - A Comparative Study*, International Journal on Advances in Life Sciences, vol. 2, no. 1, pp. 18–28, Enero 2010.

[8] P. Chandrasekar, S. Chapaneri, y D. Jayaswal, *Emotion Recognition from Speech using Discriminative Features*, International Journal of Computer Applications, vol. 101, pp. 31–36, Septiembre 2014.

[9] H. Palo, P. Kumar, y N. Mohanty, *Emotional Speech Recognition using Optimized Features*, International Journal of Research in Electronics and Computer Engineering, vol. 5, no. 4, pp. 4–9, Diciembre 2017.

[10] M. Sinith, E. Aswathi, T. Deepa, C. Shameema, y S. Rajan, *Emotion Recognition from Audio Signals using Support Vector Machine*, IEEE International Conference on Recent Advances in Intelligent Computational Systems (RAICS), Trivandrum Kerala, India, Diciembre 2015.

[11] Y. Pan, P. Shen, y L. Shen, *Feature Extraction and Selection in Speech Emotion Recognition*, IEEE Conference on Advanced Video and Signal Based Surveillance, Como, Italia, Septiembre 2005.

[12] R. Kosti, J. Alvarez, A. Recasens, y A. Lapedriza, *Emotion Recognition in Context*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, Julio 2017.

[13] M. Ménard, P. Richard, H. Hamdi, B. Daucé, y T. Yamaguchi, *Emotion Recognition based on Heart Rate and Skin Conductance*, in 2nd International Conference on Physiological Computing Systems, Angers, Francia, Febrero 2015.

[14] H. Guo, Y. Huang, C. Lin, J. Chien, K. Haraikawa, y J. Shieh, *Heart Rate Variability Signal Features for Emotion Recognition by using Principal Component Analysis and Support Vectors Machine*, in 2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE), Taichung, Taiwan, Octubre 2016.

[15] A. Pradhan, A. Singh, y S. Saraswat, *Emotion Recognition through Wireless Signal*, in 2017 4th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, Febrero 2017.

[16] J. Castillo, Á. Castro, F. Alonso, A. Fernández, y M. Salichs, *Emotion Detection and Regulation from Personal Assistant Robot in Smart Environment*, Personal Assistants: Emerging Computational Technologies, Springer, 2018.

[17] F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, y B. Weiss, *A Database of German Emotional Speech*, in INTERSPEECH, Lisbon, Portugal, Septiembre 2005.

[18] S. Grocholewski, *Corpora-speech Database for Polish Diphones*, in Fifth European Conference on Speech Communication and Technology, Rhodes, Greece, Septiembre 1997.

[19] O. Martin, I. Kotsia, B. Macq, y I. Pitas, *The enterface'05 Audio-Visual Emotion Database*, in Proceedings of the 22nd International Conference on Data Engineering Workshops, Washington, Estados Unidos, 2006.

[20] P. Jackson and S. Haq, *Surrey Audio-Visual Expressed Emotion (SAVEE) Database*, University of Surrey, Guildford, Reino Unido, 2014.

[21] R. Barra, J. Montero, J. Macías, S. Lutfi, J. Lucas, F. Fernández, L. D'haro, R. San Segundo, J. Ferreiros, y R. Córdoba, *Spanish Expressive Voices: Corpus for Emotion Research in Spanish*, 6th Conference of Language Resources and Evaluation, Morroco, Mayo 2008.

[22] I. Iriondo, R. Guaus, A. Rodríguez, P. Lázaro, N. Montoya, J. Blanco, D. Bernadas, J. Oliver, D. Tena, y L. Longhi, *Validation of an Acoustical Modelling of Emotional Expression in Spanish using Speech Synthesis Techniques*, ITRW on Speech and Emotion, Northern Ireland, Reino Unido, Septiembre 2000.

[23] F. Burkhardt y W. Sendlmeier, *Verification of Acoustical Correlates of Emotional Speech using Formant-Synthesis*, ITRW on Speech and Emotion, Northern Ireland, Reino Unido, Septiembre 2000.

[24] C. Van der-Hofstadt Román, *El Libro de las Habilidades de Comunicación*, Ediciones Díaz de Santos, 2005.

[25] J. Rázuri, D. Sundgren, R. Rahmani, A. Larsson, A. Cardenas, y I. Bonet, *Speech Emotion Recognition in Emotional Feedback for Human-Robot Interaction*, International Journal of Advanced Research in Artificial Intelligence, vol. 4, no. 2, pp. 20–27, Febrero 2015.

[26] R. Bates, *Ekman: The Face of Man: Expressions of Universal Emotions in a New Guinea Village*, Studies in Visual Communication, vol. 7, no. 1, pp. 83–85, Mayo 2017.

[27] J. Hansen y S. Bou-Ghazale, *Getting Started with Susas: a Speech under Simulated and Actual Stress Database*, in EUROSPEECH, Rhodes, Greece, Septiembre 1997.

[28] S. Steidl, *Automatic Classification of Emotion Related User States in Spontaneous Children's Speech*, Ph.D. Tesis, Universitat Erlangen-Nurnberg, Erlangen, 2009.
[29] V. Rosas, R. Mihalcea, y L. Morency, *Multimodal Sentiment Analysis of Spanish Online Videos*, IEEE Intelligent Systems, vol. 28, no. 3, pp. 38–45, Junio 2013.
[30] *Audacity (version 2.0. 0): Audio Editor and Recorder*, https://www.audacityteam.org.
[31] *Pyaudio: Portaudio v19 Python Bindings*, https://people. csail. mit. edu/hubert/pyaudio.
[32] *Youtube-dl: Download Videos from Youtube*, https://youtube-dl.org.
[33] T. Giannakopoulos, *Pyaudioanalysis: An Open-source Python Library for Audio Signal Analysis*, PloS one, vol. 10, no. 12, pp. 1-17, Diciembre 2015.
[34] C. Anagnostopoulos, T. Iliou, y I. Giannoukos, *Features and Classifiers for Emotion Recognition From Speech: a Survey from 2000 to 2011*, Artificial Intelligence Review, vol. 43, no. 2, pp. 155–177, Febrero 2015.
[35] T. Giannakopoulos y A. Pikrakis, *Introduction to Audio Analysis: a MATLAB Approach*, Academic Press, 2014.
[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss y V. Dubourg *Scikit-learn: Machine Learning in Python*, Journal of Machine Learning Research, vol. 12, no. 10, pp. 2825–2830, Octubre 2011.
[37] D. Ververidis y C. Kotropoulos, *Review of Emotional Speech Databases*, in Panhellenic Conference on Informatics (PCI), Thessaloniki, Greece, Semtiembre 2003.
[38] H. Nguyen, K. Kotani, F. Chen, y B. Le, *A Thermal Facial Emotion Database and its Analysis*, in Image and Video Technology, Berlin, Heidelberg, 2014.

Índice de Autores

REVECOM

Sociedad Venezolana de Computación

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

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

ISSN: 2244-7040

9 772244 704006

www.svc.net.ve/revecom