www.komputersapiens.org

KOMPUTER SAPIENS

Revista de Divulgación de la Socoedad Mexicana de Inteligencia Artificial

Año 10
Volumen 1
Ene-Abril 2018
$50.00

SEMBLANZA DE JOSÉ NEGRETE

HERRAMIENTA PARA EL LIGADO DE OBJETOS

ENFOQUES BASADOS EN LEXICÓN Y LSA

VERIFICACIÓN DE TUITS USANDO ALGORITMO DE CLASIFICACIÓN

SMARTPHONES COMO RECOLECTORES DE DATOS

CLASIFICANDO CONOCIMIENTO ARQUITECTÓNICO

TWITTER PARA VIGILAR LA INCIDENCIA DE ENFERMEDADES

ISSN 2007-0691

©Komputer Sapiens, Año X Volumen I, enero-abril 2018, es una publicación cuatrimestral de la Sociedad Mexicana de Inteligencia Artificial, A.C., con domicilio en Ezequiel Montes 56 s/n, Fracc. los Pilares, Metepec, Edo. de México, C.P. 52159, México, http://www.komputersapiens.org, correo electrónico: editorial@komputersapiens.org, tel. +52 (833)357.48.20 ext. 3024, fax +52 (833) 215.85.44. Impresa por Sistemas y Diseños de México S.A. de C.V., calle Aragón No. 190, colonia Álamos, delegación Benito Juárez, México D.F., C.P. 03400, México, se terminó de imprimir el 30 de abril de 2018, este número consta de 1000 ejemplares.
Reserva de derechos al uso exclusivo número 04-2009-111110040200-102 otorgado por el Instituto Nacional de Derechos de Autor. ISSN 2007-0691.

Komputer Sapiens es una revista de divulgación en idioma español de temas relacionados con la inteligencia artificial.
Creada en LaTeX, con la clase **papertex** disponible en el repositorio *CTAN*: Comprehensive TeX Archive Network, http://www.ctan.org/
Indizada en el IRMDCT de CONACYT y en Latindex.

Contenido

ARTÍCULO ACEPTADO

Herramienta para el ligado de objetos en la tendencia de Linked Data

por Alicia Martínez-Rebollar, Fernando Pech-May y Alfredo Temiquelt

pág. 7 ⇒ Desarrollo de un método para la generación de enlaces entre los objetos de un *dataset* inicial y un *dataset* externo.

ARTÍCULO ACEPTADO

Enfoques basados en Lexicón y LSA para la detección de polaridad en reseñas

por Mireya Tovar Vidal, Karen Leticia Vázquez-Flores, y Gerardo Flores Petlacalco

pág. 13 ⇒ Dos algoritmos para el análisis de sentimientos en reseñas de clientes sobre los dominios de restaurantes y venta de laptops.

ARTÍCULO ACEPTADO

Verificación automática del tema en tuits usando un algoritmo de clasificación

por J. Fidencio García-Amaro, José L. Martínez-Rodríguez, Ana B. Ríos-Alvarado e Iván López-Arévalo

pág. 18 ⇒ Utilización de un algoritmo de aprendizaje para clasificar tuits de acuerdo a experiencias previas.

ARTÍCULO ACEPTADO

Smartphones como medio de recolección de datos para aplicaciones de aprendizaje computacional

por Jorge Eduardo Ibarra Esquer, Félix Fernando González Navarro y Brenda Leticia Flores Ríos

pág. 24 ⇒ Aplicaciones que se han dado a los datos obtenidos de los dispositivos inteligentes, opciones existentes para obtenerlos y el procesamiento que debe realizarse para poder utilizarlos.

ARTÍCULO ACEPTADO

Clasificando conocimiento arquitectónico a través de técnicas de minería de texto

por Samuel González-López, Gilberto Borrego Soto, Aurelio López-López y Alberto L. Morán y Solares

pág. 29 ⇒ Etiquetado de interacciones en METNE para facilitar la recuperación de CA a través de herramientas de búsqueda.

ARTÍCULO ACEPTADO

Uso de Twitter para vigilar la incidencia de enfermedades infecciosas en México

por Pedro C. Santana-Mancilla, J Román Herrera-Morales, Gerardo Chowell-Puente, Víctor M. González y Francisco Javier Luna-Vázquez

pág. 34 ⇒ Análisis de tuits generados con base en palabras clave relacionadas a enfermedades infecciosas.

Columnas

Sapiens Piensa. Editorial pág. 2

e-Tlakuilo pág. 4

Estado del IArte pág. 6

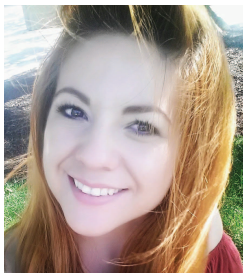
Sakbe pág. 5

IA & Educación pág. 40

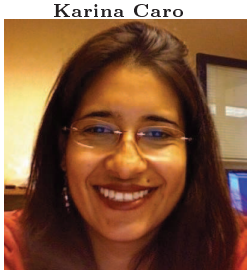
Deskubriendo Konocimiento pág. 42

Sapiens Piensa

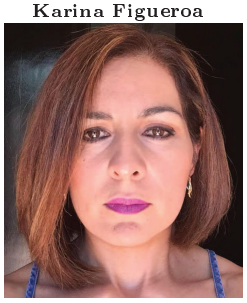
Karina Caro, **Karina Figueroa** y **Marcela D. Rodríguez**



Karina Caro



Karina Figueroa



Marcela Rodríguez

Nuestra revista Komputer Sapiens nació gracias al espíritu social de la Sociedad Mexicana de Inteligencia Artificial (SMIA). Uno de los principales iniciadores e impulsores de esta conciencia fue el ***Dr. José Negrete Martínez***, quien falleció a principios del presente año. Él fue el primer presidente de la SMIA, entregando su vida a la ciencia y a su divulgación. La última morada académica del Dr. Negrete fue el Centro de Investigación en Inteligencia Artificial de la Universidad Veracruzana, donde estuvo desde 1992 hasta su partida. En el año 1994, el Dr. Negrete participó en la creación de la Maestría en Inteligencia Artificial (MIA), primer programa en su tipo en el país y que el próximo año estará cumpliendo 25 años. Más de 150 egresados de la MIA llevan consigo las enseñanzas y el amor por la ciencia que el Dr. Negrete sembró en sus corazones. Agradecemos toda la entrega y dedicación que brindó a la SMIA, una *sociedad de amigos*, como él mismo la llamó. Dedicamos esta edición para recordar al Dr. Negrete. En la columna Deskubriendo Konocimiento presentamos una semblanza de su valiosa obra.

La inteligencia Artificial (IA) - la pasión del Dr. Negrete - está presente en infinidad de dispositivos electrónicos que utilizamos día con día. El desarrollo y uso masivo de estos dispositivos ha mantenido una clara sinergia con la IA, que busca detectar patrones en los comportamientos de los usuarios, aprender de ellos y adaptarse a su uso. Las aplicaciones de estas tecnologías contribuyen a resolver problemas en diversos dominios, como la educación, el sector salud, la interacción humano-computadora, entre otros.

En esta edición especial de **Komputer Sapiens** se presentan seis artículos seleccionados cuidadosamente, que discuten diferentes aplicaciones, incluyendo temas de arquitectura, análisis de sentimientos en texto y web semántica. Estos trabajos son versiones extendidas de una selección de los artículos presentados en el Encuentro Nacional de Computación (ENC) 2015 y 2016.

En "**Verificación automática del tema en tuits usando un algoritmo de clasificación**" los autores proponen un método para filtrar tuits en español que pertenecen a un tema específico enlazado a una etiqueta (i.e., *hashtag*). El método propuesto se basa en utilizar un algoritmo de aprendizaje para clasificar tuits de acuerdo a experiencias previas; es decir, un usuario se encarga de identificar y etiquetar tuits como correctos o incorrectos de acuerdo al tema que describe un *hashtag*, de manera que el algoritmo sea capaz de clasificar automáticamente el tema de nuevos tuits con el mismo *hashtag* como correctos o no.

En el artículo "**Smartphones como medio de recolección de datos para aplicaciones de aprendizaje computacional**", se presentan algunos casos donde se han utilizado smartphones para acceder a datos capturados por sensores (acelerómetros, giroscopios, GPS, etc.). Los autores muestran algunas aplicaciones que se han generado a partir de los datos recolectados. Además, muestran algunas opciones para acceder a los datos de los sensores.

Los autores del artículo "**Herramienta para el ligado de objetos en la tendencia Linked Data**" hacen énfasis en que la web semántica (o datos enlazados) es la evolución de la web pretendiendo dar significado explícito a los recursos utilizados en las páginas web. Dichos recursos pueden ser interconectados con fuentes externas por medio de los datos enlazados. Los autores muestran que las ontologías son un mecanismo para presentar y compartir el conocimiento. En este artículo se desarrolla un método para la generación de enlaces de objetos.

En el artículo "**Enfoques basados en Lexicón y LSA para la detección de polaridad en reseñas**" se muestra un estudio con técnicas de minería de datos para el análisis de sentimientos haciendo procesamiento de lenguaje natural. El ámbito para este estudio es un restaurante y el insumo las opiniones del servicio, las cuales pueden ser en inglés o español. Se presentan dos algoritmos, uno basado en un Lexicón y otro que utiliza un Análisis Semántico Latente (Latent Semantic Analysis) y un clasificador de K-vecinos.

En "**Clasificando conocimiento arquitectónico a través de técnicas de minería de texto**" se muestra una aplicación de la inteligencia artificial al campo de la arquitectura de software. Los autores muestran los grandes problemas que se tienen en grupos de trabajo para desarrollar software, uno de estos problemas es la falta de etiquetas en los mensajes de comunicación entre miembros remotos. En este artículo se muestra un método para clasificar conocimiento arquitectónico con

 ISSN 2007-0691

el objetivo de apoyar el etiquetado entre miembros y así contribuir a la recuperación de ese conocimiento a través de herramientas de búsqueda.

Finalmente, en el artículo "**Uso de Twitter para vigilar la incidencia de enfermedades infecciosas en México**" se aprovecha la creciente publicación de contenidos personales en las redes sociales que pueden ser minados para descubrir patrones. El objetivo de este artículo es determinar si en una herramienta como twitter, en donde los usuarios mencionan enfermedades infecciosas, los mensajes tienen alguna relación con los casos reales de enfermedades reportadas por el sector salud. Los autores muestran cómo analizar los tuits, agruparlos y modelarlos.

Deseamos que este número especial de **Komputer Sapiens**, que hemos preparado con mucha dedicación, sea de interés y del agrado de nuestros lectores.

Dra. Karina Caro *es investigadora postdoctoral en el College of Computing and Informatics y en el Antoinette Westphal College of Media Arts & Design en la Universidad de Drexel en la ciudad de Filadelfia, Pensilvania, EEUU. Sus áreas de investigación incluyen interacción humano-computadora, cómputo ubicuo y accesibilidad.*

Dra. Karina Figueroa *es investigadora asistente en la Facultad de Ciencias Físico-matemáticas de la Universidad Michoacana de San Nicolás de Hidalgo, México. Sus áreas de interés son recuperación de información, bases de datos métricas y algoritmos en general.*

Dra. Marcela D. Rodríguez *es profesor-investigador de la Facultad de Ingeniería, de la Universidad Autónoma de Baja California, Campus Mexicali, B.C., México. Sus intereses de investigación incluyen: cómputo ubicuo, interacción humano-computadora y agentes de software, con aplicaciones en la Informática Médica.*

Dr. José Negrete Martínez.
Doctor Honoris Causa por la Universidad Veracruzana

"Desde 1994 es investigador invitado de la Universidad Veracruzana, desde que decide tomar dos años sabáticos para ayudar a iniciar la Maestría en Inteligencia Artificial que inauguraba en 1994 en Xalapa. En 1996 se incorpora a la Unidad Periférica del Instituto de Investigaciones Biomédicas con sede en Xalapa. En la Maestría regresa a trabajar sobre la *auto-organización modular del cerebro*. Aquí inicia sus trabajos con robots físicos que, dotados de computadoras concurrentes, a bordo, le permiten experimentar con el control auto-organizando de módulos orientados a la conducta. Por su trabajo en este tipo de robótica es invitado a formar parte del comité editorial de la revista *Applied Bionics and Biomechanics*. La Universidad Veracruzana le publica *Pericia Artificial : un Aprendizaje Constructivista de Sistemas Expertos*. El centro actual de su investigación es la *evolución dirigida de cerebros de robot*".

El texto anterior fue leído al consejo universitario de la Universidad Veracruzana el 28 de noviembre del 2005 donde se le otorgó el *Doctorado honoris causam*.

Tomado de DR. JOSÉ NEGRETE MARTÍNEZ, Biografía Académica
`http://www.uv.mx/jnegrete/`

 ISSN 2007-0691

e-Tlakuilo: Cartas de nuestros lectores

Jorge A. Ruiz-Vanoye, **Ocotlán Díaz-Parra** y **Alejandro Fuentes-Penna**
etlakuilo@komputersapiens.org

En *Komputer Sapiens* nos hemos esforzado por estar "a sólo un click de distancia" a través de diferentes medios como Facebook, Twitter y correo electrónico. Les presentamos algunas de las preguntas que hemos recibido:

Kevin Serrano Camacho - Alumno de Universidad. (vía correo electrónico)
Actualmente soy estudiante de Ingeniería de software y me interesa la Inteligencia Artificial. Tengo una pregunta: ¿Qué tipo de procesador me recomiendan que sea especializado para usar en Inteligencia Artificial, pero que no sea muy costoso?

Hola, gracias por escribir. Específicamente, para el aprendizaje máquina y las redes neuronales artificiales sin muchos usuarios solicitando diversos procesos, puedes usar CPUs (unidades de procesamiento central) con GPUs (Unidades de procesamiento gráfico; son coprocesadores gráficos usados para ayudar a los CPUs con procesos gráficos o de videojuegos).

Para el aprendizaje máquina y las redes neuronales artificiales con muchos usuarios solicitando diversos procesos se recomienda usar TPUs (Tensor Processing Unit), desarrollado por Google en el 2016. Los TPUs son usados para realizar tareas específicas ASIC (Circuito Integrado de Aplicaciones Específicas). Google usa los TPUs para el reconocimiento de imágenes, Google Translate, Google Street View y transformar voz en texto). Los TPUs de uso específico pueden ser modificables a los nuevos procesos o necesidades de los usuarios, realizan más rápido y más operaciones que los CPUs o GPUs y con el menor costo energético. La mala noticia es que aún no están disponibles para su compra a los usuarios o empresas.

 ISSN 2007-0691

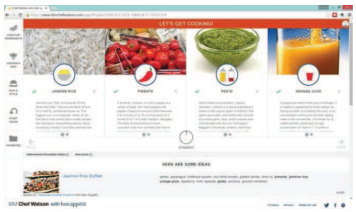
Sakbe

Claudia Guadalupe Gómez Santillán y **Héctor Gabriel Acosta Mesa**
sakbe@komputersapiens.org

El concepto de Inteligencia Artificial (IA) fue acuñado en 1956 durante la conferencia de Dartmouth. Dicho evento reunió a los científicos en computación más renombrados de esa época. A partir de ese momento, la disciplina se diseminó rápidamente por el mundo hasta nuestros días. El concepto de IA se fundamenta en programas informáticos que buscan emular el pensamiento humano. En la actualidad existen muchas aplicaciones de la IA en diversos dominios.

https://www.definicionabc.com/tecnologia/inteligencia-artificial.php

Una de estas aplicaciones es Chef Watson, un chef virtual que tiene las mejores recetas del mundo y que te guiará paso a paso para preparar ese deleite culinario que creías imposible. Sus algoritmos te darán recomendaciones con base en ingredientes que son precargados, dándote opciones variadas.



https://blog.adext.com/es/aplicaciones-de-la-inteligencia-artificial

En esta liga `https://blog.adext.com/es/tecnologias-inteligencia-artificial-2018` encontrarás 19 tecnologías de IA que están dominando el 2018, que van desde aplicaciones de Lenguaje natural, reconocimiento de voz, plataformas de aprendizaje de máquina, hardware optimizado con IA, hasta defensa cibernética, entre otras.

https://blog.adext.com/es/tecnologias-inteligencia-artificial-2018

Una de las aplicaciones de IA más esperadas es la de los vehículos autónomos: automóviles que sin la necesidad de un conductor son capaces de establecer rutas, interpretar señales de tránsito, rebasar, estacionarse, evadir obstáculos e interactuar con sus tripulantes de manera totalmente autónoma. Algunos prototipos de estos vehículos ya están a punto de salir a las calles. Un ejemplo es la línea de Google (`https://www.xataka.com/automovil/el-coche-autonomo-de-google-waymo-se-vuelve-completamente-autonomo-y-por-primera-vez-sale-a-la-calle-sin-conductor`).

https://www.xataka.com/automovil/el-coche-autonomo-de-google-waymo-se-vuelve-completamente-autonomo-y-por-primera-vez-sale-a-la-calle-sin-conductor

 ISSN 2007-0691

Estado del IArte

María del Pilar Gómez Gil (@pgomezgil) y Jorge Rafael Gutiérrez Pulido (@jrgpulido)
estadoiarte@komputersapiens.org

Como mencionamos anteriormente, nuestra revista *Komputer Sapiens* y la columna **Estado del IArte** están cumpliendo 10 años.

Esta es la segunda parte del tema *Aplicaciones de la inteligencia artificial*. En la primera hicimos un recuento de los cinco aspectos más sobresalientes de la última década: el poder de cómputo, *big data*, nuevos algoritmos de inteligencia computacional, robótica y nuevas interfaces de comunicación entre seres humanos y computadoras. Sin duda, seguiremos viendo avances importantes en estos aspectos en periodos cada vez más cortos. Para muestra, un par de botones:

Hace unos días, IBM, NVIDIA y el Departamento de Energía de los Estados Unidos de América develaron a *Summit*, la computadora más rápida del planeta hasta hoy. Luego de un periodo de prueba en el laboratorio Nacional Oak Ridge en Tennessee, se dio a conocer que es capaz de ejecutar 200 petaflops, 200×10^{15} operaciones por segundo; esto es 8 veces más rápido que su predecesora. *Summit* consta de $4,600$ sistemas IBM de 9 nodos. Cada nodo está equipado con 6 "Volta TensorCore GPUs" de NVIDIA, procesadores que resultan excelentes para realizar tareas y operaciones de inteligencia y aprendizaje artificial. Para saber más (en inglés): `https://www.ibm.com/thought-leadership/summit-supercomputer`.

Por otro lado, hace unos meses Google presentó como software libre sus bibliotecas de aprendizaje artificial. Tradicionalmente, los desarrollos de aprendizaje artificial, conocidos en inglés como *machine learning*, eran propietarios y se realizaban en ciertos lenguajes de programación; Python, por ejemplo. Ahora, Google dio a conocer que todo su músculo ha sido liberado, incluyendo nuevos lenguajes al repertorio: go, R, haskell, java, c++, julia, swift y javascript, entre otros. Las implicaciones de esto son obvias, una explosión en el número de aplicaciones para todo dispositivo: sensores, teléfonos, tablets, computadoras de escritorio y, por supuesto, en los navegadores Firefox, Chrome, Opera y Safari, principalmente. Para saber más (en inglés): `http://blog.tensorflow.org`.

Como podemos ver, la Inteligencia Artificial ha tenido un desarrollo impresionante, y con éste han aparecido también nuevos retos. Uno de los más importantes actualmente es el diseño de sistemas inteligentes que cumplan con las reglas éticas esperadas en los productos ingenieriles utilizados para actividades humanas. Para asegurar el cumplimiento de reglas acordes con el bienestar humano, el pasado noviembre la Asociación de Estándares del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE-SA) anunció la aprobación de 3 estándares, relacionados al aseguramiento de ética en el diseño de sistemas basados en inteligencia artificial. Estos estándares buscan dar prioridad al uso de ética y bienestar de las personas en todos los aspectos asociados al desarrollo de tecnologías autónomas e inteligentes.

Los estándares que se anunciaron son:

- IEEE P7008™ – "Estándar para las persuasiones conducidas éticamente en los sistemas robóticos, inteligentes y autónomos" (*Standard for Ethically Driven Nudging for Robotic, Intelligent and Autonomous Systems*). Algunos sistemas inteligentes tienen la capacidad de hacer sugerencias a las personas, de manera directa o indirecta, sobre qué acciones llevar a cabo sobre una determinada tarea. Por otro lado, la información que presentan puede manipular las emociones. Esto implica una gran responsabilidad en el diseño, a fin de asegurar que no se llegue a una manipulación no deseada por los usuarios.

- IEEE P7009™ – "Estándar para el diseñoo libre de fallas de sistemas autónomos y semi-autónomos (*Standard for Fail-Safe Design of Autonomous and Semi- Autonomous Systems*). Los sistemas inteligentes y autónomos realizan actividades donde pueden estar involucradas decisiones importantes asociadas a la salud o al bienestar de las personas. Por esto, es fundamental asegurar lo mejor posible que estos sistemas no contengan errores en sus diseños. Recordemos que no existen sistemas 100 % libres de fallas, pero siempre pueden diseñarse de forma que se minimice la probabilidad de tener errores.

- IEEE P7010™ – "Estándar sobre Métricas de bienestar para una Inteligencia Artificial ética y sistemas autónomos (*Wellbeing Metrics Standard for Ethical Artificial Intelligence and Autonomous Systems*). Este estándar está relacionado con el diseño de medidas que permitan cuantificar aspectos asociados a ética en el diseño de sistemas inteligentes.

Para saber más (en inglés): `http://standards.ieee.org/news/2017/ieeeglobalinitiative.html`.✳

 ISSN 2007-0691

ARTÍCULO ACEPTADO

Herramienta para el ligado de objetos en la tendencia Linked Data

Alicia Martínez-Rebollar, **Fernando Pech-May** y **Alfredo Temiquelt**

La Web Semántica, una tendencia exitosa en los últimos años

La Web Semántica es la evolución de la Web convencional y tiene como objetivo dar significado explícito a los recursos utilizados en páginas Web. El núcleo de la Web semántica son las ontologías. La ontología es un mecanismo para representar y compartir formalmente el conocimiento de un dominio (personas, lugares, etc.); consiste de diferentes tipos de componentes, tales como clases, individuos y relaciones y se representan con lenguajes como RDF y OWL [1]. Debido a su constante crecimiento han surgido nuevos desafíos y nuevas oportunidades para el acceso y búsqueda de información. La Web Semántica propone un cambio de paradigma en la búsqueda de información, en el cual las búsquedas no van a estar guiadas por la sintáctica, sino por la semántica, realizando búsquedas basadas en conceptos y relaciones entre conceptos [2].

Por otra parte, la evolución de la Web Semántica, denominada datos enlazados, permite que desde múltiples fuentes de datos estructurados (en forma de tripletas) puedan ser entrelazados y sean de mayor utilidad a través de consultas semánticas. Esto hace posible construir la Web de datos como una gran base de datos interconectados y distribuidos en la Web. En la Figura 1 se muestran distintos datasets de diversos tipos, conectados entre sí, que componen la nube de datos enlazados.

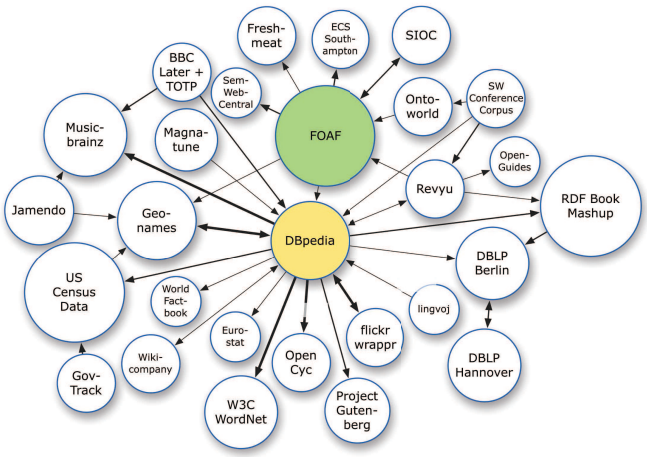


Figura 1. Datasets de diversos tipos enlazados entre sí.

Hoy en día son muchas las organizaciones que publican sus datos mediante los principios de Enlazado de datos, además de utilizar estándares semánticos como RDF [3] (siglás en inglés de *Resource Description Framework*), OWL [4] (acrónimo del inglés *Web Ontology Language*) y SPARQL [5, 6] (es un acrónimo recursivo del inglés *SPARQL Protocol and RDF Query Language*). Estas tecnologías han permitido que tanto seres humanos como agentes de software puedan interpretar y procesar estos datos para producir información de utilidad para el usuario final [7].

Es fundamental para la Web de datos conectar los conjuntos de datos (dataset) que se publican y que permanecen aislados en la Web [8]. Existen dos enfoques que permiten crear enlaces entre los objetos de dos datasets: 1) ligar objetos de forma manual, lo cual es factible siempre y cuando ambos datasets involucrados sean pequeños y estáticos (aquellos datasets que con el tiempo no cambian), y 2) ligar objetos de forma automática o semiautomática cuando alguno de los dos datasets sea grande o dinámico que cambie rápidamente en el tiempo [9].

El inconveniente que se presenta en el Enlazado de datos de forma automática o semiautomática es que no siempre los dos datasets involucrados hacen uso de los mismos formalismos y/o vocabularios para describir los datos que éstos contienen, provocando que el total de los posibles enlaces de dos o más dataset no se lleven a cabo con éxito.

La Web Semántica propone un cambio de paradigma, en el cual las búsquedas no van a estar guiadas por la sintáctica, sino por la semántica, realizando búsquedas basadas en conceptos y relaciones entre conceptos [2].

El objetivo de este artículo es desarrollar un método para la generación de enlaces entre los objetos de un dataset inicial y un dataset externos, este último perteneciente a la nube de datos enlazados, haciendo uso de formalismos, vocabularios, lenguajes y *frameworks*. Este método fue implementado en una herramienta de software y ha sido evaluado utilizando las métricas de precisión y exhaustividad en cinco casos de estudio.

Trabajos relacionados

Actualmente existen diversas metodologías que hacen uso de técnicas de datos enlazados. A continuación, se muestran algunas metodologías utilizadas en la generación de dataset enlazados en la nube de Linked Data.

 ISSN 2007-0691

Hyland & Wood [10] presentan una metodología dirigida al sector gobierno con el objetivo de producir y publicar datos de alta calidad basándose en las mejores prácticas para fomentar el aprovechamiento de los datos enlazados.

Alvarez Rodriguez [11] aborda el tema de licitaciones públicas y datos enlazados. El trabajo recoge la aplicación de métodos semánticos para la producción, publicación y consumo de datos abiertos concretamente en el campo de la contratación pública electrónica.

Pérez et al. [12] proponen un método para generar dataset dentro del ámbito nutricional, partir de una o varias fuentes de datos convencionales y generar un repositorio Web. Permite que herramientas automáticas operen sobre la información y presten nuevas funcionalidades en el ámbito de la salud pública.

Delgado et al. [13] presentan la plataforma BM2LOD, que implementa un ciclo de vida para la publicación de datos bibliográficos siguiendo los principios de datos enlazados. BM2LOD está compuesta por herramientas que trabajan en conjunto de forma automática para generar datos bibliográficos enlazados.

Auer et al. [14] proponen una plataforma que implementa un método que representa un ciclo de vida para la producción, publicación y consumo de datos enlazados denominado LOD2 Stack. Esta plataforma está compuesta por herramientas que trabajan de forma manual para generar datos enlazados.

Método para el enlazado de objetos

El método para el enlazado de objetos está formado por cuatro actividades que permiten generar objetos entre dos conjuntos de datos diferentes. La Figura 2 muestra el método propuesto y las fases que lo componen, las cuales son: 1) búsqueda de datasets de dominio, 2) extracción de términos del dataset, 3) obtención de sinónimos de los términos y 4) generación de enlazado de recursos.

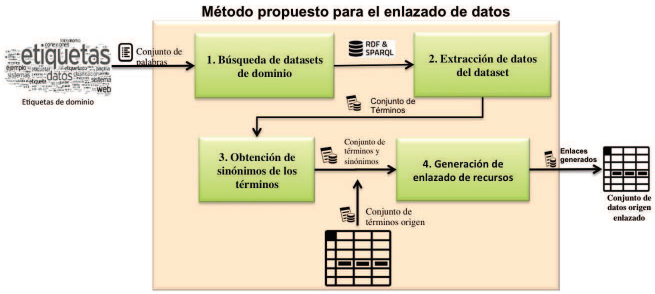


Figura 2. Método propuesto para el enlazado de datos.

Búsqueda de conjunto de datos (datasets) de dominio en la nube de datos enlazados

El objetivo de esta fase es identificar y recuperar un grupo de datasets de dominio referenciados en la nube de datos (ver Figura 3). Existen dos enfoques que son actualmente referenciados: a) búsqueda de instancias de conjuntos de datos de dominio y b) búsqueda de conjuntos de datos de dominio a partir de etiquetas de dominio. Este último fue utilizado en nuestra propuesta porque permite recuperar sólo aquellos datasets de un dominio específico dentro de la nube de datos enlazados [15].

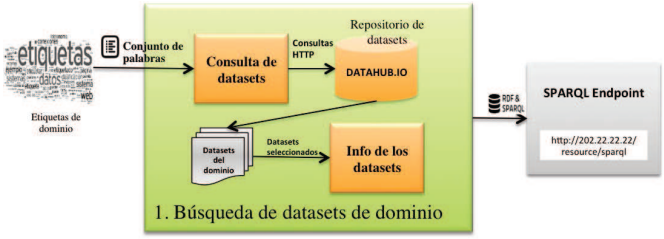


Figura 3. Búsqueda de datasets de dominio.

Para la búsqueda se utiliza conjunto de palabras o etiquetas de dominio que sirven como criterios de búsqueda y pueden hacer referencia al a) título del dataset de dominio, b) etiquetas que describen el dominio o c) nombre de un dominio. Posteriormente se genera una consulta utilizando un conjunto de patrones proporcionados por el API REST de Ckan al repositorio DataHub.io [16]. Las consultas son enviadas vía http al repositorio DataHub.io donde se encuentran referenciados los datasets de dominio que integran la nube de datos enlazados. Como resultado se obtiene una lista de datasets recuperados en DataHub.io. Posteriormente, se muestra la información del dataset buscado para que el experto de dominio seleccione cada uno de los datasets y recupere la información que los describe. En la información recuperada de cada dataset se encuentra: a) el tipo de licencia, b) la información del encargado del mantenimiento, c) el nombre del autor, d) el listado de recursos, e) el estado actual, etc, de cada dataset. Finalmente, se obtiene un Sparql Endpoint, el cual hace referencia a la ruta donde se encuentran alojadas las tripletas RDF para obtener el dataset de dominio.

Extracción de términos del conjunto de datos

n este módulo se extraen y almacenan en una base de datos los términos que componen a un conjunto de datos de dominio y obtiene el conjunto de sinónimos de cada uno de los términos extraídos (ver Figura 4). Los términos son las clases y propiedades usadas en la estructura del dataset. Para extraer los términos de un dataset se utiliza el extractor de términos Vocab-express (herramienta que explora y proporciona el vocabulario utilizado en un dataset), la lista de los graphs en el triplestore, la lista de los vocabularios, la lista de todas las clases, la lista de todas las propiedades, el número de instancias de cada clase y el número de instancias de

cada propiedad. Tiene como entrada el Sparql endpoint del conjunto de datos de dominio almacenado en Virtuoso. Los términos extraídos del dataset de dominio son almacenados de forma temporal en tablas de MySql.

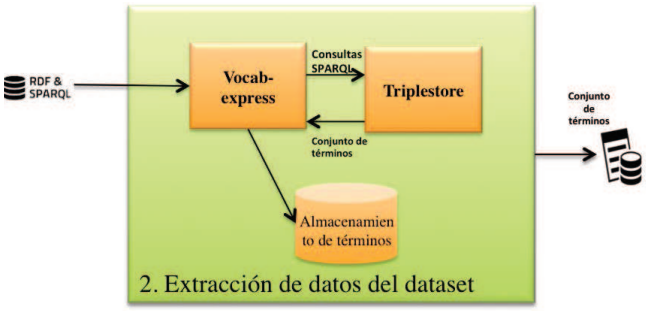


Figura 4. Extracción de datos del dataset.

Obtención de sinónimos de los términos

Esta fase obtiene los sinónimos de cada uno de los términos extraídos del dataset de dominio desde WordNet [17] (ver Figura 5). WordNet contiene conjuntos de sinónimos cognitivos denominados synsets. Primero, se consultan los términos que fueron almacenados en la base de datos léxica WordNet. Se generan consultas por medio de la librería JWI [18], cuyo fin es obtener de la base de datos léxica WordNet los sinónimos de cada uno de los términos recuperados. Segundo, se obtienen los sinónimos por cada término con el objetivo de obtener el conjunto o synset de sinónimos identificados en la tarea "Base de datos léxica".

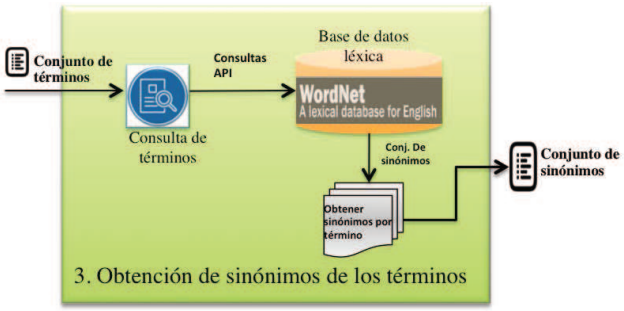


Figura 5. Obtención de sinónimos de los términos.

Generación de enlazado de recursos

Esta fase se generan enlaces entre objetos de dos datasets haciendo uso de una o más métricas de similitud y la relación "*sameAs*".

En la Figura 6 se muestra el conjunto de tareas que se llevan a cabo para el enlace de objetos. La tarea de selección de términos para el enlazado muestra una lista de los términos y sinónimos que fueron obtenidos del dataset de dominio, para que un experto de dominio de forma manual seleccione los términos. Para generar el enlazado de objetos se utiliza el *framework* Silk [19]. Si un par de objetos cumple el umbral establecido, entonces se genera un enlace entre ellos por medio de la relación sameAs. Como resultado se obtiene el conjunto de datos origen enlazado en forma de tripletas RDF.

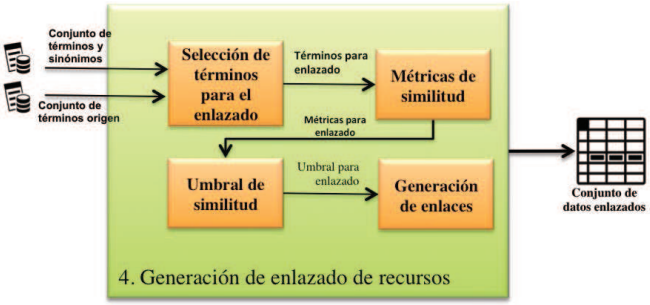


Figura 6. Módulo de generación de enlazado de recursos.

La Tabla 1 muestra un algoritmo con los pasos necesarios para generar enlaces entre objetos de dos conjuntos de datos y un ejemplo de un enlace generado de forma correcta.

Tabla 1. Ejemplo de un enlace correcto generado entre un conjunto de datos origen y el conjunto de datos de dominio de la ACM

Algoritmo para generar enlaces entre objetos
1. CDO = Conjunto de datos origen
2. CDACM = Conjunto de datos de la ACM
3. Autor_CDO = CDO -> Término_Author [**Author="Juan Perez"**]
4. Titulo_CDO = CDO -> Término_Title [**Title="web semántica: estado del arte"**]
5. Autor_CDACM = CDACM -> Término_Has-author **[Has-author="Juan Pérez"]**
6. Titulo_CDACM = CDACM -> Término_Has-title **[Has -title="Web Semántica: Estado del Arte"]**
7. US_CDO_CDACM_Autor = Métrica_Similitud (Autor_ CDO, Autor_ CDACM)
8. US_CDO_CDACM_Titulo = Métrica_Similitud (Titulo_CDO, Titulo_ CDACM)
9. US = Promedio (US_CDO_CDACM_Autor, US_CDO_CDACM_Titulo) **[Promedio = 0.95654]**
10. Valor Entero o Fracción establecido (VEF) = **0.90**
11.- SI US >= VEF
Enlace aceptado
SINO
Enlace rechazado
FINSI

El módulo de búsqueda de conjunto de datos de dominio tiene como entrada etiquetas de dominio. Poteriormente, se generan tres consultas rest usando un conjunto de patrones proporcionados por el api rest de *Ckan*[1] y finalmente como salida se obtiene un listado de conjuntos

[1] http://docs.ckan.org/en/ckan-2.2/api.html

de datos de dominio referenciados en la nube de datos enlazados. En Tabla 2 se muestra un ejemplo de la consulta *rest* generada y el objeto *json* recuperado desde el repositorio DataHub.io.

Tabla 2. Objeto json recuperado del repositorio DataHub.io

GET http://datahub.io//api/search/dataset?q=acm;limit=5000			
Parámetros	Encabezados	Respuesta	JSON
{"count":8, "results":["rkb-explorer-acm","infovis-contest-2004", "kore-50-nif-ner-corpus", "vis-seven-scenarios-codings", "vis-revision-corpus", "forsyth", "bjerrum", "eriksen"] }			

El módulo de extracción de términos y obtención de sinónimos tiene como entrada el SPARQL Endpoint, luego por medio de una consulta se invoca al extractor de términos Vocab-express y finalmente como salida se obtienen los términos del conjunto de datos de dominio, los cuales son almacenados en una base de datos (ver Tabla 3).

Tabla 3. Fragmento del conjunto de clases que componen al conjunto de datos de dominio de la ACM[2]

http://www.aktors.org/ontology/portal#Person
http://www.aktors.org/ontology/portal#Proceedings-Paper-Reference
http://www.aktors.org/ontology/portal#Article-Reference
http://purl.org/dc/terms/LCSH
http://purl.org/dc/terms/W3CDTF
http://www.w3.org/2001/XMLSchema#nonNegativeInteger
http://purl.org/dc/terms/ISO639-2
http://purl.org/dc/terms/LCC
http://www.gutenberg.org/rdfterms/etext
http://www.instancematching.org/ontologies/oaei2014#Book
http://www.w3.org/2002/07owl#NamedIndividual
http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag

Pruebas

La evaluación se llevó a cabo analizando los resultados obtenidos de la herramienta que implementa dicho método. Para cada caso de estudio se utilizó un dataset inicial y un dataset de dominio (ambos se almacenaron en el triple-store Virtuoso para así tener un SPARQL Endpoint). Para el enlazado de objetos fueron seleccionados tres datasets del dominio de publicaciones. Los tres casos de estudio se evaluaron aplicando las métricas de precisión [20] y la exhaustividad (recall) [21].

La precisión es la fracción de los documentos recuperados que son documentos relevantes. Nosotros aplicamos la siguiente expresión:

$$Precisión = \frac{|Verdaderos\ positivos|}{|Verdaderos\ positivos|+|Falsos\ positivos|}$$

La exhaustividad es la fracción de documentos relevantes que son documentos recuperados.

$$Exhaustividad = \frac{|Verdaderos\ positivos|}{|Verdaderos\ positivos|+|Falsos\ negativos|}$$

A continuación, se describen los resultados obtenidos de los primeros tres casos de estudio.

Caso de estudio de la Iniciativa de evaluación de Alineación ontológica

El primer caso de estudio utilizado fue el de la Iniciativa de evaluación de alineación ontológica [22], formado por 172 libros de literatura y un conjunto de datos inicial de 20 libros de literatura. Estos conjuntos de datos se almacenaron en el triple-store Virtuoso en un servidor local para así tener un SPARQL End-point. La Iniciativa de evaluación de alineación ontológica es un órgano de evaluación cuyo objetivo es evaluar las nuevas tecnologías emergentes con respecto al alineamiento ontológico, generar conjuntos de datos para evaluar el rendimiento de las herramientas y detectar el grado de similitud entre pares de objetos.

Después de aplicar la herramienta al conjunto de datos para el enlazado de objetos, se generó un documento que contiene las tripletas RDF de los objetos enlazados, los cuales fueron 1212.

Para la precisión y exhaustividad se consideró la siguiente información: (i) El total de enlaces a generar es de 1166. (ii) El total de enlaces generados por el método para el enlazado de objetos es de 1212.

- Enlaces generados de forma correcta: 1166
- Enlaces generados de forma incorrecta: 46
- Enlaces no generados de forma incorrecta: 0

De los 1212 objetos enlazados, se generaron 1166 enlaces de forma correcta y 46 enlaces de forma incorrecta. De esta manera al aplicar las fórmulas de las métricas, la precisión fue de 96.2 % y 100 % de exhaustividad.

Caso de estudio de la Biblioteca digital de la ACM

La biblioteca digital de la ACM (por sus siglas en inglés: *Association for Computing Machinery*) es la colección más completa de artículos que cubre las áreas de la informática y las tecnologías de la información. Esta biblioteca incluye: artículos, revistas, actas de congresos, boletines, etc.

Para este caso se utilizó un conjunto de datos iniciales compuesto por 20 artículos (cada artículo puede tener uno o más autores) y el conjunto de datos de la ACM, donde fueron extraídos 173 artículos.

Para la precisión y exhaustividad se consideró la siguiente información: (i) El total de enlaces a generar es de 51. (ii) El total de enlaces generados por el método para el enlazado de objetos es de 55.

[2] http://dl.acm.org/

- Enlaces generados de forma correcta: 51
- Enlaces generados de forma incorrecta: 4
- Enlaces no generados de forma incorrecta: 0

La precisión fue del 92.7 % y 100 % exhaustividad.

Caso de estudio con Librería Gutenberg

El dataset Gutenberg[3] ofrece más de 46,000 libros electrónicos gratuitos a partir de libros que ya existen físicamente y pueden ser descargados libremente. Para este caso el dataset inicial fue de 20 libros (con un sólo autor) y fueron extraídos 200 libros (con uno o más autores).

Para la precisión exhaustividad se consideró la siguiente información: (i) El total de enlaces a generar es de 19. (ii) El total de enlaces generados por el método para el enlazado de objetos es de 21.

- Enlaces generados de forma correcta: 19
- Enlaces generados de forma incorrecta: 2
- Enlaces no generados de forma incorrecta: 0

La precisión fue de 90.4 % y 100 % en exhaustividad.

Conclusiones

Los primeros resultados obtenidos para en enlazado de objetos con esta herramienta fueron favorables, con una precisión promedio de 93.1 %. Sin embargo, se pretenden realizar más pruebas con conjuntos de datos de distintos dominios, tales como geografía y multimedia. El haber utilizado el repositorio DataHub.io aseguró que el conjunto de datos de dominio recuperado pertenezca a la nube de datos enlazados, asimismo el uso de Vocabexpress, WordNet y Silk representó una característica importante e innovadora para la herramienta propuesta.

Como trabajo futuro se pretende agregar un módulo para la eliminación, modificación y/o actualización de la información que contiene un dataset enlazados, así como una interfaz gráfica cuyo objetivo sea la explotación de los enlaces creados entre dos conjuntos de datos. ✷

REFERENCIAS

1. Piero, G. Z. (2011). "RDF and OWL for knowledge Management". *Encyclopedia of knowledge management.*
2. Berlanga, R., Nebot, V. y Jimenez-Ruiz, E. (2010). "Semantic annotation of biomedical texts through concept retrieval".
3. Kendall, G. C., Feigenbaum, L. y Torres, E. (2008). "Sparql protocol for rdf. World Wide Web Consortium". *Recommendation REC-rdf-sparql-protocol- 20080115.*
4. Biffl, S. y Sabou, M. (2016). "Semantic Web Technologies for Intelligent Engineering Applications". *Springer.* ISBN 978-3-319-41488-1.
5. DuCharme, B. (2013). "Learning SPARQL". *Second Edition. O'Reilly Media, Inc.*
6. Heath, T. y Bizer, C. (2011). "Linked Data: Evolving the Web into a Global Data Space". Hendler, J. y Harmelen, F. Eds.) *Synthesis Lectures on the Semantic Web: Theory and Technology.* Morgan & Claypool.
7. Oliveira, J., Delgado, C. y Assaife, A. C. (2017). "A recommendation approach for consuming linked open data". *Expert Systems with Applications.* Elsevier. Vol. 72, No. 1, pp.407-420.
8. Musto C., Basile P., Lops P., Gemmis M. y Semeraro G. (2017) "Introducing linked open data in graph-based recommender systems". Information Processing & Management, Elsevier 53 (2), 405-435.
9. McCrae, J., Chiarcos, C., Bond, F., Cimiano, P. y Declerck, T. (2016). "The Open Linguistics Working Group: Developing the Linguistic Linked Open Data Cloud". En *Proc. of the Tenth International Conference on Language Resources and Evaluation.*
10. Hyland, B. y Wood, D. (2011). "The joy of data - A cookbook for publishing linked government data on the web". En *Linking Government Data.* Springer. pp. 3–26. .
11. Rodriguez, A. (2012). *Métodos semánticos de reutilización de datos abiertos enlazados en las licitaciones públicas.* Tesis Doctoral, Oviendo, España: Universidad de Oviedo.
12. Pérez, R. M., Santos, J. M., Alonso, V. M., Alvarez, S., y Mikic, F. (2012). "Linked data como herramienta en el ámbito de la nutrición". *Nutrición Hospitalaria.* Vol. 27, pp. 323–332.
13. Hidalgo-Delgado, Y., Reyes-Alvarez, L., Leiva-Mederos, A., Roldan-Garcia, M. y Aldana-Montes, J. (2014). "Bm2lod: Platform for publishing bibliographic data as linked open data". En *Proc. of 7th IADIS International Conference on Information Systems.* IADIS Press.
14. Auer, S., Buhmann, L., Dirschl, C., et al. (2012). "Managing the life-cycle of linked data with the lod2 stack". *The Semantic Web – ISWC 2012.* Vol. 7650, pp. 1–16.
15. Schmachtenberg, M., Bizer, C., Jentzsch, A. y Cyganiak, R. (2014) "Linking Open data Cloud Diagram 2014". Recuperado el 14 de Octubre de 2014 de: `http://lod-cloud.net/`.
16. Open Knowledge Foundation. (2013). Ckan. Recuperado el 1 de Julio de 2015 de Docs: `http://docs.ckan.org/en/ckan-2.2/api.html`.
17. Miller, G. (1998). "WordNet: A Lexical Database for English". *Communications of the ACM.* Vol. 38, No. 11, pp. 39-41.
18. Finlayson, M. (2014). "Java Libraries for Accessing the Princeton Wordnet: Comparison and Evaluation". En *Proc. of the 7th Global Wordnet Conference.*
19. Volz, J., Bizer, C., Gaedke, M., y Kobilarov, G. (2009). "Silk – A Link Discovery Framework for the Web of Data". En *2nd Workshop about Linked Data on the Web (LDOW2009).* Madrid, España.
20. Harman, D. (2011). "Information Retrieval Evaluation". Morgan & Claypool Publishers, 1st edition.
21. Baeza-Yates, R., y Ribeiro-Neto, B. (1999). "Modern Information Retrieval". Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
22. Caso de studio 1 (2017). Iniciativa de Evaluación de Alineación Ontológica. Recuperado de: `http://oaei.ontologymatching.org/`.

[3] `http://www.gutenberg.org/`

SOBRE LOS AUTORES

Alicia Martínez Rebollar es profesora investigadora del Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), Cuernavaca, Morelos, México. Recibió su Doctorado en Informática por la Universidad Técnica de Valencia, España, y la Universidad de Trento, Italia. Sus áreas de interes son la computación afectiva, computación ubicua, el modelado organizacional y las ontologías.

Fernando Pech May es estudiante de Doctorado en Ciencias de la Computación del CENIDET, Cuernavaca, Morelos, obtuvo su Maestría en Ciencias de la Computación en el Centro de Investigación y Estudios Avanzados del IPN (CINVESTAV) México. Sus áreas de interés son recuperación de información y Web semántica.

Alfredo Temiquelt obtuvo la maestría en Ciencias de la computación en el CENIDET. Su área de interés es la Web semántica.

ARTÍCULO ACEPTADO

Enfoques basados en lexicón y LSA para la detección de polaridad en reseñas

Mireya Tovar Vidal, Karen Leticia Vazquez-Flores, y Gerardo Flores Petlacalco

En este trabajo, se lleva a cabo un estudio de minería de opiniones o análisis de sentimientos, que es un área del Procesamiento de Lenguaje Natural y una disciplina entre la recuperación de información y la lingüística computacional. Su objetivo es detectar la polaridad o los sentimientos expresados en un texto u opinión. En este trabajo, se realiza la detección de polaridad en un conjunto de opiniones de usuarios hacia el dominio de restaurantes en idioma español e idioma inglés, así como opiniones de laptops en idioma inglés. Dos algoritmos son presentados como propuestas de solución, en el primero se utiliza un algoritmo basado en un lexicón y en el segundo se utiliza el Análisis Semántico Latente (LSA - *Latent Semantic Analysis*). Ambos enfoques resultan competitivos al lograr más de un 70 % de *Accuracy* de la polaridad en el conjunto de datos anotados (*gold*).

Introducción

El análisis de sentimiento hace referencia a una disciplina de la recuperación de información, es decir, la minería de opinión (MO). La MO analiza las características de las opiniones, sentimientos y emociones que se expresan en textos [1]. Una subtarea del análisis de sentimientos es la clasificación de sentimientos en clases por medio de métodos algorítmicos, por ejemplo: positivo, negativo, neutro o conflicto. Algunas áreas de aplicación en las que la MO puede ser útil son: sistemas de recomendación, política, inteligencia de negocios y reseñas de sitios web.

En este trabajo se presenta el análisis de sentimientos dirigido hacia la determinación de la calidad de los servicios recibidos, es decir, sobre opiniones o críticas que los usuarios aportan por algún servicio recibido, en este caso hacia restaurantes y laptops. Las opiniones analizadas están escritas en el idioma español e inglés para el dominio de restaurantes y en el idioma inglés para el dominio de laptops.

Como parte del *International Workshop on Semantic Evaluation* (SemEval) 2016 [2], este artículo propone una solución para el análisis de sentimientos basado en aspectos a nivel de texto. El objetivo de esta investigación, es que dado un conjunto de reseñas de clientes sobre una entidad objetivo (por ejemplo: una computadora portátil o un restaurante), identificar la polaridad de las opiniones expresadas en cada reseña.

Dado el conjunto de opiniones por entidad objetivo, nuestro propósito es determinar el tipo de sentimiento expresado en la opinión ante el servicio recibido y clasificar automáticamente las nuevas opiniones recibidas por otros clientes con un grado de confianza alto, que le permita a la entidad (por ejemplo, el restaurante) determinar el grado de satisfacción de sus clientes que emiten la reseña u opinión (sentimiento positivo, negativo, neutro o conflicto). Por lo tanto, proponemos dos algoritmos que extienden los trabajos presentado en [3,4]. El primer algoritmo se basa en el uso de un lexicón, en el cual el enfoque extrae el grado de polaridad de las palabras involucradas en la opinión. El segundo algoritmo utiliza un enfoque basado en la coocurrencia de las palabras en distintos contextos a nivel semántico, es decir, por medio del Análisis Semántico Latente (LSA).

La minería de opiniones es una técnica para determinar el sentimiento que un escritor manifestó en la escritura de un documento.

El documento se encuentra organizado de la siguiente forma: primero se inicia con algunos trabajos relacionados con esta investigación, después continuamos con los algoritmos propuestos, posteriormente se presentan los resultados obtenidos y finalmente se muestran las conclusiones.

Trabajo relacionado

A continuación se describen brevemente algunos de los trabajos relacionados con la minería de opiniones:

La investigación llevada a cabo en [5], menciona el desarrollo de un sistema de análisis de sentimientos para comentarios de clientes en Internet. Utilizaron SentiWordNet, también utilizan el enfoque basado en reglas de medidas difusas. Encontraron algunos beneficios en esta investigación; tales como retroalimentación de usuarios en tiempo real, inteligencia de mercado accionable basada en retroalimentación directa de usuario y tiempo de reacción, mejoraron el servicio y calidad en el mercado.

Otros autores utilizan twitter como datos de entrada

 ISSN 2007-0691

e intentan clasificarlos como positivos o negativos en el idioma español. Tal es el caso de [6] que utiliza el método de clasificación de Máquina de Soporte Vectorial.

Por otro lado en [7] describen SiTAKA un sistema para el análisis de sentimientos en twitter en idioma inglés y árabe, el sistema propone la representación de tweets utilizando un conjunto de características que incluye una bolsa de palabras negadas e información proporcionada por algunos léxicos. Utilizan el clasificador Máquina de Soporte Vectorial para determinar la polaridad de los tweets. El sistema obtuvo el octavo lugar al evaluar tweets en idioma inglés y el segundo lugar al evaluar tweets en idioma árabe en SemEval 2017.

Una herramienta útil que define el tipo de sentimiento expresado en una palabra es el lexicón de polaridad.

Hay muchos trabajos reportados en la literatura asociados con la minería de opiniones, algunos enfocados al análisis de sentimientos en twitter y otros a reseñas. En esta sección se presentan diferentes enfoques, algunos más complejos que otros, sin embargo, nuestros algoritmos propuestos resultan ser competitivos de acuerdo a los resultados experimentales obtenidos.

Enfoque basado en lexicón

En esta propuesta se utiliza un lexicón de polaridad, ML-SentiCon, con el fin de determinar la cantidad de palabras positivas y negativas que contiene cada una de las opiniones.

Se utiliza el lexicón ML-SentiCon, el cual incluye listas de palabras o lemas positivos y negativos para diferentes idiomas, entre ellos, español e inglés. Específicamente para el idioma español se tienen 11,549 lemas y 25,423 lemas en el caso del idioma inglés. Cada lema se acompaña con el puntaje numérico de polaridad entre -1.0 y 1.0 , siendo números negativos las polaridades negativas, y los positivos, las polaridades positivas, se adiciona a cada lema su valor de desviación estándar y la categoría gramatical para cada uno (verbo, sustantivo, entre otros)[1] [8]. A continuación se presentan los pasos que se siguen en este enfoque:

1. Pre-procesamiento. El cual consiste de los siguientes pasos:
   - Extracción de las opiniones de los documentos en formato XML y la formación de un corpus de opiniones con su polaridad.
   - Limpieza de opiniones: Eliminar palabras vacías, signos de puntuación, acentos y caracteres aislados.
   - Tokenización: Tokenizar opiniones por palabra.
   - Stemming. Proceso heurístico que corta el final de las palabras y con frecuencia incluye la eliminación de los afijos derivativos, por ejemplo: coches = coche, fuimos = fui, etc.
2. Extracción de características. Esta etapa consiste de los siguientes pasos:
   - División del lexicón en palabras positivas y palabras negativas obteniendo dos sub-lexicones: positivo y negativo.
   - División de la opinión en un vector positivo ($LexPos$) y en otro vector negativo ($LexNeg$). En los cuales se consideran las palabras positivas del sub-lexicón positivo y lo mismo para el negativo.
   - Cálculo de frecuencia del término (TF, del inglés Term Frequency). Por cada término o palabra en el sub-lexicón incluida en la opinión se realiza un cálculo de frecuencia de los mismos en la opinión $Op = \{w_1, w_2, ..., w_n\}$.
   - Cálculo de polaridad. Se determina la cantidad de palabras positivas $Pos(Op)$ y la cantidad de palabras negativas por opinión $Neg(Op)$ y el valor máximo determina la polaridad del mismo, es decir, si el valor máximo de palabras es positivo la opinión obtiene polaridad positiva ($Pol(Op)$ =max$\{Pos(Op), Neg(Op)\}$).
3. Evaluación. La medida de evaluación utilizada es $Accuracy$, que es el promedio de las predicciones correctas.

$$Accuracy = \frac{|polaridades\ correctas|}{|opiniones|}$$

En la Figura 1 se muestra el conjunto de pasos que se realizan para determinar la polaridad de una opinión utilizando el enfoque basado en el uso de un lexicón.

Enfoque basado en LSA

El LSA parte de la idea de que las palabras en el mismo campo semántico tienden a aparecer juntas o en contextos similares. En este caso se considera que los palabras de las opiniones están semánticamente relacionadas en las clases o categorías. Esas palabras pueden

[1] http://timm.ujaen.es/recursos/ml-senticon/

estar en la misma opinión o en diferentes opiniones compartiendo información en común [9].

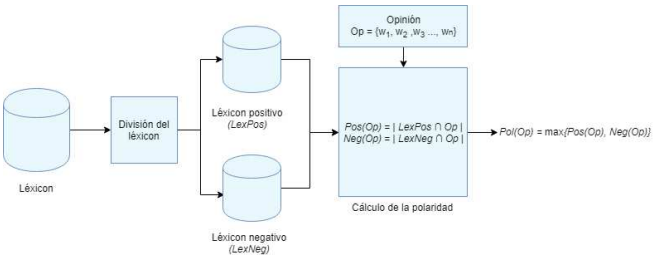


Figura 1. Arquitectura del algoritmo basado en un lexicón.

Cuando se usa LSA para analizar un texto primero genera una matriz de ocurrencias de cada palabra en cada documento, en esta investigación usamos la medida de pesado de términos TF-IDF (del inglés *Term Frequency - Inverse Document Frequency*, es decir, frecuencia de término - frecuencia inversa de documento).

Posteriormente, se emplea la técnica *Singular-Value Decomposition* (SVD) para descomponer la matriz TF-IDF en un conjunto de k dimensiones, con valores de entre 100 a 300, factores ortogonales desde donde la matriz original puede ser aproximada por combinaciones lineales. En lugar de representar cada documento como una lista de términos independientes esta técnica representa el documento como una serie de valores continuos que comparten un mismo contexto. Por ejemplo, si dos términos se usan en contextos similares, tendrán vectores similares en la representación reducida de LSA [10].

El proceso se divide en tres partes, en la primera se aplica la etapa de pre-procesamiento del enfoque anterior, en la segunda se enfoca a crear el modelo LSA para hacer la clasificación de las opiniones y la tercera en aplicar la medida de evaluación *Accuracy*, definida en el enfoque anterior. Los pasos del enfoque se aplican sobre el conjunto de datos de entrenamiento y sobre el conjunto de datos de prueba. Una vez obtenido el modelo de clasificación con los datos de entrenamiento y la predicción de la polaridad sobre los datos de prueba se realiza la evaluación de los mismos a través de la medida antes mencionada.

En la segunda parte se aplican los siguientes pasos:

- Formación de un conjunto de opiniones con su polaridad correspondiente.
- Del conjunto obtenido se separa la opinión y la polaridad en conjuntos diferentes.
- Formación del modelo vectorial. Usando la función *TfidfVectorizer*[2] del paquete sklearn, el conjunto de opiniones se convierte en un conjunto de vectores con valores *TF-IDF* como característica, es decir, la matriz de representación.
- Aplicación de LSA y reducción de dimensionalidad. Utilizando *TruncatedSVD*[3] aplicamos LSA y reducimos la dimensión de la matriz a 150 dimensiones.
- Clasificación. Utilizamos el paquete *KNeighborsClassifier*[4] con 5 vecinos y la medida de similitud *Coseno* para obtener las clases: positivo, negativo, neutro y en conflicto.

Los pasos que se siguen en este algoritmo son representados en la Figura 2.

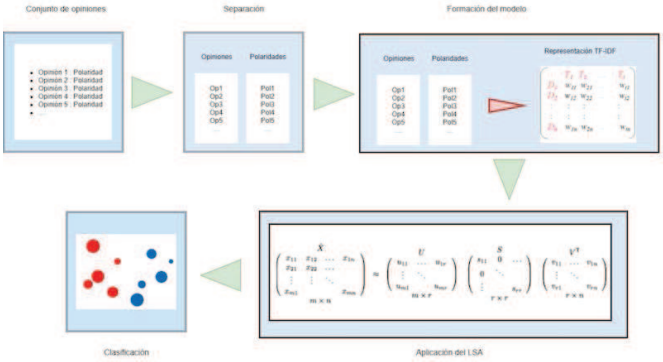


Figura 2. Arquitectura del algoritmo basado en LSA.

Experimentos

En esta sección se describen los conjuntos de datos que se usaron para probar los enfoques propuestos, también se muestran los resultados de *Accuracy* obtenidos por cada enfoque.

Conjunto de datos

En los experimentos realizados, se utilizaron los conjuntos de datos de entrenamiento y prueba, provenientes de SemEval 2016 [2]. El conjunto de prueba incluye las evaluaciones del gold standard, para ser posible medir la calidad del enfoque propuesto. Las opiniones de los usuarios son dadas para dos dominios: Restaurantes (escritos en inglés y español), y Laptops (escritas únicamente en inglés). En la Tabla 1 se muestra el número de textos (opiniones) proporcionadas por SemEval 2016.

[2] http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
[3] http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html
[4] http://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

Las palabras en el mismo campo semántico tienden a aparecer juntas o en contextos similares.

Tabla 1. Total de opiniones por conjunto de datos y dominio

Dominio	Entrenamiento	Prueba	Gold
Restaurantes (español)	627	268	268
Restaurantes (inglés)	335	90	90
Laptops (inglés)	395	80	80

Resultados

Considerando los resultados obtenidos sobre los datos de prueba utilizados por ambos enfoques, se observa en la Tabla 2 que el enfoque basado en LSA logra en el dominio de Laptops el 71 % de *Accuracy* como resultado más bajo en los tres dominios. Esto nos indica que el método es bastante competitivo para determinar si una opinión es positiva o negativa. El número de dimensiones utilizadas es de 150, lo que permite realizar una mejor clasificación de las clases o polaridades, al reducir la dimensionalidad a esa cantidad. En la Tabla 3 se presenta el total de opiniones positivas P, negativas N, neutras E y en conflicto C existentes en el gold; así como el valor de *Accuracy* obtenido por el enfoque basado en LSA para cada dominio. En particular, una de las deficiencias que logramos identificar en el corpus es el desbalanceo de las clases. Debido a que la polaridad positiva tiene más frecuencia en el corpus, los enfoques tienden a predecir con mayor facilidad este tipo de polaridad, sin lograr resultados competitivos en las restantes clases.

Tabla 2. Resultados obtenidos con los enfoques sobre los datos de prueba

Dominio	*Accuracy*	
	LSA	LEX
Restaurantes (español)	0.91	0.69
Restaurantes (inglés)	0.86	0.77
Laptops (inglés)	0.71	0.75

Por otro lado, el enfoque basado en lexicón obtiene el 69 % de *Accuracy* para el dominio de Restaurantes en español. Consideramos que el uso de *stemming* en la fase de preprocesamiento, afectó los resultados, debido a que el lexicón solo considera palabras lematizadas como entrada. Sin embargo, esto nos indica que es posible su uso para definir la polaridad de la opinión en los datos del gold, aún a pesar del tipo de palabra de entrada.

Tabla 3. Resultados de *Accuracy* obtenidos por clase con el enfoque basado en LSA

Dominio	Polaridad Gold				*Accuracy* LSA			
	P	N	E	C	P	N	E	C
Restaurantes (español)	268	0	0	0	0.91	0	0	0
Restaurantes (inglés)	72	17	1	0	0.95	0.52	0	0
Laptops (inglés)	57	22	1	0	1.0	0	0	0

Conclusiones

En este trabajo presentamos dos enfoques para la clasificación de sentimientos, el primero utiliza como características la frecuencia de la polaridad obtenida a través de un lexicón, el segundo utiliza como característica la medida TF-IDF, LSA y el clasificador k vecinos más cercanos. El análisis de sentimientos en opiniones se llevó acabo en dos dominios diferentes: restaurantes y laptops. La clasificación de la polaridad con el enfoque basado en LSA logró mejores resultados que con el segundo enfoque basado en lexicón. En el caso del enfoque basado en lexicón observamos que los resultados obtenidos para los dominios en el idioma inglés son mayores que el resultado en el idioma español y esto se debe a que el lexicón en inglés contiene una mayor cantidad de información que en el otro lexicón. Por lo tanto, entre mayor información exista en el lexicón es más probable la determinación correcta del tipo de polaridad de la opinión. Este trabajo nos permitió extender la propuesta presentada en [3] y [4] a nivel de polaridad de opiniones. Como trabajo a futuro, consideramos el uso de una combinación de LSA con las palabras de las opiniones lematizadas comunes en el lexicón, así como el uso de otros tipos de lexicones que permitan extender la información de la opinión y determinar su polaridad.✳

Agradecimientos. Esta investigación es parcialmente financiada por los proyectos PRODEP 00570 (EXB-792) DSA/103.5/15/10854, CONACyT CB 257357, proyecto VIEP-BUAP 00478 y proyecto 100409344-VIEP2018.

REFERENCIAS

1. Waltinger, U. (2010). "Sentiment Analysis Reloaded: A Comparative Study On Sentiment Polarity Identification Combining Machine Learning And Subjectivity Features". En *Proc. 6th International Conference on Web Information Systems and Technologies (WEBIST '10)*. Valencia, España. pp. 203-210.

2. Pontiki, M., Galanis D., Papageorgiou H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyou, M., Zha, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jiménez-Zafra, S. M. y Eryi_git, G. (2016). "SemEval-2016 task 5: Aspect based sentiment analysis". En *Proc. 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California. pp.19-30.

3. Vazquez, K., Tovar, M. y Pinto, D. (2016). "Algoritmos de Aprendizaje Supervisados para el Análisis de Sentimientos en el Dominio de Restaurantes y Laptops". En Caro K., et al. (Eds.) *Tecnologías Emergente y Avances de la Computación en Mexico*. Departamento Editorial, UACH. pp. 222-227.

4. Vazquez, K., Tovar, M., Castillo, H., Rossainz, M. (2017). "Algoritmos para detectar la calidad de servicio en los dominios de restaurantes y laptops". *TecnoINTELECTO*. Vol. 14, No. 1, pp. 51-58.

5. Pimpalkar, A. P. (2013). "A sentimental analysis of movie reviews involving Fuzzy Rule-Based". *International Journal of Artificial Intelligence and Knowledge Discovery*. Vol. 3, pp. 9-14.

6. Siordia, O., Moctezuma, D., Graff, M., Miranda-Jimenez, S., Téllez, E. y Villaseñor, E. (2015). "Sentiment Analysis for Twitter: TASS 2015". En Proc. *TASS@ SEPLN. CEUR-WS*. Vol.1397, pp. 65-70.

7. Jabreel, M. y Moreno, A. (2017) "SiTAKA at SemEval-2017 Task 4: Sentiment Analysis in Twitter Based on a Rich Set of Features". En *Proc. 11th International Workshop on Semantic Evaluations (SemEval-2017)*. Vancouver, Canada. pp. 694–699.

8. Fermín, L. C., Troyano, J. A., Pontes, B., Ortega, F. J. (2014). "Building layered, multilingual sentiment lexicons at synset and lemma levels". *Expert Systems with Applications*. Vol. 41, No. 13, pp. 5984-5994.

9. Gutiérrez, R. M. (2005). "Análisis Semántico Latente: ¿Teoría psicológica del significado?". *Signos*. Vol. 38, No. 59, pp. 303-323.

10. Peter W. Foltz (1996). "Latent semantic analysis for text-based research". *Behavior Research Methods, Instruments, & Computers*. Vol. 28, No. 2, pp. 197–202.

SOBRE LOS AUTORES

Mireya Tovar Vidal es profesora de tiempo completo en la Facultad de Ciencias de la Computación, en la Benemérita Universidad Autónoma de Puebla desde el 2003, obtuvo su doctorado en Ciencias de la Computación en el Centro Nacional de Investigación y Desarrollo Tecnolgico (CENIDET) en el 2015. Sus líneas de investigación son el Procesamiento de Lenguaje Natural, ingeniería ontológica, recuperación de información y la minería de textos. Miembro del Sistema Nacional de Investigadores Nivel I (2016 - 2018) de Mexico.

Karen Leticia Vazquez-Flores obtuvo el grado de Licenciada en Ciencias de la Computación en la Benemérita Universidad Autónoma de Puebla en 2017. Actualmente es estudiante de la Maestría en Ciencias de la Computación en la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla. Sus intereses científicos incluyen procesamiento de lenguaje natural y minería de opiniones.

Gerardo Flores Petlacalco actualmente es estudiante de la Licenciatura en Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla. Sus áreas de interés son Procesamiento de Lenguaje Natural, recuperación de información y programación paralela.

ARTÍCULO ACEPTADO

Verificación automática del tema en tuits usando un algoritmo de clasificación

J. Fidencio García-Amaro, **José L. Martínez-Rodríguez**, **Ana B. Ríos-Alvarado** e **Iván López-Arévalo**

Introducción

Una red social se define como la interacción que existe entre el ser humano y la comunidad a la cual pertenece. Este concepto no ha sufrido cambios, pero la forma en que interactúan las personas a cambiado gracias a los avances tecnológicos. De acuerdo a un estudio realizado por la AMI[1] en el año 2016, 9 de cada 10 internautas acceden a alguna red social o al correo electrónico antes que a realizar una tarea de búsqueda web.

Twitter ha sido un medio popular para transmitir ideas de diversos temas utilizando un esquema de textos cortos (140 caracteres) conocidos como tuits. Dada la gran cantidad de tuits publicados, Twitter provee funcionalidades de búsqueda para encontrar aquellos que tratan temas de interés para el usuario mediante el uso de *hashtags* [2] las cuales hacen referencia a un tema por medio de palabras clave o frases cortas. Sin embargo, muchos de estos tuits podrían no estar relacionados con el tema en cuestión aún cuando éstos hayan sido etiquetados con un *hashtag* descriptivo, lo que provoca que el usuario realice tareas adicionales para filtrar la información consultada, clasificando aquella que es correcta con respecto a la que no lo es.

En este trabajo se propone un método para filtrar tuits en español que pertenecen a un tema específico enlazado a un *hashtag*. El método propuesto se basa en utilizar un algoritmo de aprendizaje para clasificar tuits de acuerdo a experiencias previas, es decir, un usuario se encarga de identificar y etiquetar tuits como correctos o incorrectos de acuerdo al tema que describe un *hashtag*, de manera que el algoritmo sea capaz de clasificar automáticamente el tema de nuevos tuits con el mismo *hashtag* como correctos o no. El método se compone de tres etapas: 1) adquisición y pre-procesamiento, donde se recolectan tuits del tema deseado y se identifican sus características más importantes; 2) entrenamiento y clasificación, donde se prepara el algoritmo de aprendizaje a utilizar; y 3) consulta y presentación, que consiste de un artefacto de software donde el usuario puede realizar búsquedas y obtener resultados clasificados. A continuación detallamos cada una de estas etapas abordando el trabajo realizado en la literatura para el tratamiento de tuits en español.

Técnicas para el pre-procesamiento de tuits

El texto encontrado en los tuits suele contener elementos no deseados, confusos o muy variados, los cuales dificultan la interpretación de la información y pueden afectar el proceso de clasificación. Por lo tanto, antes de llevar a cabo cualquier tarea sobre el texto, es necesario aplicar un pre-procesamiento para eliminar elementos innecesarios (ruido) y normalizar el texto en tuits. A continuación, se describen algunas técnicas de pre-procesamiento y normalización que se adaptan al objetivo de este trabajo.

1. Eliminación de palabras vacías. Consiste en eliminar palabras que no aportan relevancia por ser tan comunes en el texto, por ejemplo, artículos (la, los, el), pronombres (yo, mi, tú), preposiciones (por, hasta, desde), entre otros.

2. Lematización. Es un método que reduce las palabras originales a su raíz léxica. La aplicación de esta técnica ayuda a encontrar las palabras recurrentes en los documentos mediante su palabra raíz. Por ejemplo, las palabras <*caminar, caminante, caminata, caminito*>, tienen la raíz léxica <*camino*>.

3. Tokenización y Bolsa de Palabras. Tokenización permite identificar palabras clave dentro de un dominio o clase. Una vez identificadas las palabras se construye una representación considerando el modelo *bolsa de palabras*, donde cada documento es considerado como un recipiente que contiene palabras sin importar el orden.

4. Diccionario de corrección ortográfica y servicio de mensajes cortos (SMS). Permite corregir palabras ligadas a un idioma y dominio específico.

Técnicas para la clasificación de tuits

Una característica importante de los tuits es que su longitud máxima es de 140 caracteres, lo cual implica que los algoritmos de clasificación deben concentrarse en textos cortos. Algunos de los algoritmos más usados para la clasificación automática de tuits son:

[1] Asociación Mexicana de Internet `https://www.amipci.org.mx`

[2] Una etiqueta o *hashtag* es una cadena de caracteres formada por una o varias palabras concatenadas y precedidas por el símbolo #.

 ISSN 2007-0691

1. Bayesiano Ingenuo (NB). Consiste en un clasificador bayesiano simple que asume independencia entre las características. La fase de entrenamiento consiste en comparar para cada característica la cantidad de veces que es observada en cada clase, y de esta forma aproximar la probabilidad de que dicha característica indique la clase correspondiente [1].
2. Bayesiano Ingenuo Multimodal (NBM). Se basa en la aplicación de la regla de Bayes[3] para predecir la probabilidad condicional de que un término (palabra) pertenezca a una clase, además considera la frecuencia de aparición de cada término en los documentos en vez de una ocurrencia binaria [2].
3. Máquinas de Soporte Vectorial (SVM). Este método construye una hipótesis mediante el cálculo de un hiperplano (plano n-dimensional) que separe a los elementos de cada clase. El problema de optimización asociado consiste en encontrar el hiperplano separador que maximiza la mínima de las distancias a cada uno de los elementos [3].
4. Regresión Logística. Es un algoritmo de clasificación supervisada que calcula predicciones binarias. La clave de este algoritmo es la utilización de una distribución de probabilidad elegida (Gaussiana o Laplace) y de algoritmos de optimización sucesiva de los ejemplos de entrenamiento suministrados [4].

Una característica importante de los tuits es que su longitud máxima es de 140 caracteres, lo cual implica que los algoritmos de clasificación deben concentrarse en textos cortos.

Aplicaciones relacionadas con la clasificación de tuits

El procesamiento de lenguaje natural y la clasificación de publicaciones en los sitios de redes sociales han abordado temas de estudio con diversos eventos. Por ejemplo, para medir la popularidad o aceptación de un candidato y sus propuestas ante los electores durante temporadas de elección a un cargo público [5]. También se ha visto el efecto de propagación en tiempo real que tiene el dar a conocer eventos como ataques terroristas o desastres naturales como terremotos, huracanes o tornados [6].

En los experimentos de análisis de sentimientos mostrados por Jasso-Hernández, *et al.* [7] se estudian las características morfológicas en los textos con el fin de proporcionar un mejor rendimiento en la detección de carga emocional en los tuits. Con esto tratan de determinar la valoración dada por los usuarios con respecto a un tema en específico. González-Ibánez *et al.* [8] utilizan *hashtags* para construir una lista de palabras en el conjunto de documentos de entrenamiento para determinar tuits positivos, negativos y sarcásticos. Kamanksha y Sanjay [9] presentan una evaluación de cuatro algoritmos de clasificación incluyendo SVM y NB utilizados para análisis de críticas cinematográficas en tuits.

Otro aspecto a considerar en la clasificación de tuits es la detección del motivo de interés del usuario al publicar en Twitter. Martis y Alfaro [10] definen una taxonomía con las principales categorías para determinar las intenciones de publicación en Twitter, como reporte de noticia, opinión de noticia, publicidad, entre otras. Kim *et al.* [11] clasifican tuits de un tema específico en categorías de acuerdo a la subtemática que presentan. Primero recolectan tuits que presentan palabras clave relacionados al tema de cigarros electrónicos, posteriormente un grupo de usuarios etiqueta una muestra de tuits, se etiquetan asignándoles una categoría de acuerdo a cinco clases, por ejemplo mercadotecnia o aficionados. Finalmente, clasifican los tuits mediante un árbol de decisión. Un árbol de decisión es un modelo de clasificación integrado por nodos y ramas, donde cada nodo sin hojas del árbol contiene un punto de división que es una prueba de uno o varios atributos y determina cómo se particionan los datos.

Clasificación de tuits sobre un tema específico

Twitter cuenta con características donde analiza las publicaciones en tiempo real para crear una lista de temas del momento llamados *trending topics*. Lee *et al.* [12] consideran que a pesar de que existe esa característica, es necesario realizar un proceso de clasificación de alta precisión enfocándose en categorías generales para lograr una mejor recuperación de la información.

Para lidiar con el problema de clasificación de tuits relacionados a un tema específico se propone seguir un método con las etapas descritas a continuación (ver la Figura 1):

1. Adquisición y procesamiento. En esta etapa se buscan y recolectan las publicaciones que contengan la etiqueta (*hashtag*) del tema de interés. En par-

[3] La regla de Bayes es un caso especial de la probabilidad condicional que se aplica cuando se desea calcular la probabilidad condicional de un evento que ocurrió primero dado lo que ocurrió después.

ticular se recolectan los tuits que cumplan con las siguientes características: 1) contener la etiqueta ingresada en la búsqueda; y 2) estar escritos en español.

2. Entrenamiento y clasificación. Una vez que se tienen los documentos (en este caso cada tuit representa un documento) se aplican técnicas de limpieza de datos y normalización con el fin de procesar adecuadamente los documentos representativos del tema de interés. Posteriormente, ya que se tienen los documentos representativos, en esta etapa se aplica el algoritmo de clasificación seleccionado para determinar si un documento pertenece o no al tema de interés.

3. Consulta y presentación. A través de una aplicación web, el usuario ingresará una consulta que contenga la etiqueta referente al tema específico. La aplicación web devolverá los tuits asociados al tema de interés. La interfaz de usuario muestra dos listas de tuits, una con los tuits que pertenecen y otra con los que no pertenecen al tema de interés, de esta forma el usuario podrá visualizar los tuits recuperados.

En general se consideran dos etapas, en la primera *Offline*, el sistema trabaja de forma independiente llevando a cabo la tarea de entrenamiento del modelo y sus sub-tareas. En la segunda etapa, llamada *Online*, el sistema tiene interacción con el usuario final llevando a cabo las tareas de clasificación, consulta y presentación de tuits.

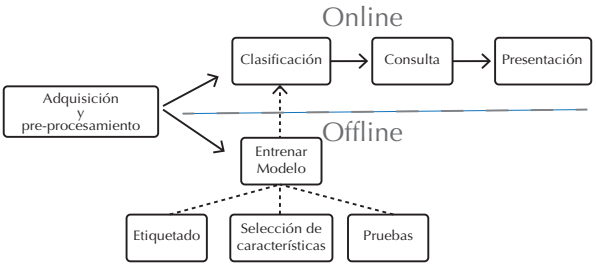


Figura 1. Método propuesto.

En las siguientes secciones se describen con mayor detalle cada una de las etapas.

Adquisición y pre-procesamiento

De acuerdo a los operadores de consulta proporcionados por el API de Twitter y la biblioteca Twitter4J[4] se desarrolló una aplicación Java para descargar los tuits con las características deseadas.

Para el pre-procesamiento de los tuis recuperados se aplicaron las siguientes técnicas:

1. Tratamiento de emoticones: Los emoticones son utilizados para expresar sentimientos referentes a la publicación. En este caso son eliminados del documento.

2. Eliminación de palabras reservadas de Twitter: Twitter utiliza palabras reservadas que ayudan a su funcionamiento, por ejemplo, para retuitear (RT), mensajes directos (DM) y nombres de usuario, las cuales son eliminadas.

3. Normalización del texto: El texto se convierte a minúsculas, se reducen las palabras a su raíz, se eliminan palabras vacías, se eliminan las letras repetidas y se limpia el texto (eliminación de caracteres especiales). Debido a que el texto está en español, también se eliminan los acentos. Los acentos le otorgan cierto uso o sentido a la palabra, pero en este caso no es considerado el sentido de las palabras.

4. Corrección ortográfica: El texto encontrado en los tuits es muy susceptible a errores de escritura o palabras incompletas, usando la biblioteca MySpell se corrigieron las palabras para el español.

5. Normalización de URLs: Las direcciones web presentes en un tuit son sustituidas por la palabra URL.

Entrenamiento y clasificación

Los documentos (tuits) filtrados y procesados entran a una etapa de entrenamiento para posteriormente pasar a la clasificación. El entrenamiento del modelo se dividió en tres etapas:

1. Etiquetado: Cada tuit es revisado y de forma manual se le asigna una etiqueta correspondiente a una clase positiva o negativa. Esto de acuerdo a si el documento contiene palabras relacionadas al contexto del tema de interés.

2. Selección de características: En esta etapa se seleccionan las palabras encontradas dentro de los tuits como características y se considera la ocurrencia de cada una de ellas en los documentos. Como método de selección de características se usó la prueba CHI cuadrada [13], la cual mide la independencia entre la clase de un texto y un término contenido en el texto. Dado el conjunto de palabras, la selección se hace con base en una puntuación que el método asigna a cada palabra, es decir, se toma una cantidad del total de palabras de los documentos con la más alta puntuación.

[4] https://github.com/yusuke/twitter4j

3. Pruebas: En esta etapa se utilizan los clasificadores: Bayesiano Ingenuo (NB), Bayesiano Ingenuo Multimodal (NBM) y Máquinas de Soporte Vectorial (SVM). Con cada algoritmo se ejecutan las mismas pruebas hasta encontrar el clasificador adecuado y su configuración óptima. El modelo es desarrollado con cada uno de los documentos obtenidos después de la etapa de pre-procesamiento, entrenamiento y selección de características.

Consulta y presentación

En el módulo de consulta se consolidan las tareas de clasificación y consulta de tuits con la implementación de una Interfaz Gráfica de Usuario, la cual permite recibir la consulta del usuario y mostrar una lista de tuits clasificados. Cada tuit devuelto contiene información alusiva al mismo, tal como el texto de la publicación, identificación del tuit otorgado por Twitter, información del usuario y la etiqueta de clase obtenida por el clasificador. Este módulo se desarrolló bajo la arquitectura cliente-servidor utilizando Java. Los lenguajes y tecnologías utilizadas del lado del cliente son HTML5, JavaScript, CSS3, jQuery y Ajax.

Evaluación y Resultados

La evaluación de los algoritmos de clasificación y de cada una de las fases propuestas en la metodología requiere de la implementación de los algoritmos y recolección de un conjunto de tuits con una clasificación *a priori*. Dentro de las medidas de evaluación usadas se encuentran la precisión (P)[5], la exhaustividad (E)[6] y la medida F1 (FM)[7].

En el estado del arte se reportan clases genéricas para efectuar las pruebas, por ejemplo, la clase *Deportes*, es un tema de interés general. Para este trabajo se seleccionó el tema de situaciones de riesgo (`#SDR`), ya que es un tema que genera interés en México en estos momentos. Tomando en cuenta este dominio, se han definido las clases *Seguridad* y *No Seguridad* para ubicar a un conjunto de tuits que se generan a partir de un evento que amenaza la integridad de las personas que viven en una ciudad.

Pruebas de clasificación

Se construyó un corpus (colección de documentos sobre un tema de interés) para la evaluación de los algoritmos de clasificación. Para ello, se realizaron búsquedas periódicas con el tema de interés durante el periodo de marzo a octubre de 2015. Se recolectó un total de 89433 tuits, de los cuales se descartaron 84719 correspondientes a 61409 retuits, 1764 tuits duplicados y 21546 no representativos (menos de 5 palabras). Por lo tanto, se etiquetaron manualmente 4714 de los cuales 2351 pertenecen a la clase de *Seguridad* y 2363 a la clase *No Seguridad*.

Para las pruebas se usó la implementación de los algoritmos en Weka[8] con la configuración dada por defecto. Para SVM se consideró un kernel de base radial. Además, se realizó una validación cruzada de 10 iteraciones (*ten-fold cross validation*). En la Tabla 1, donde NC es el número de características, P es la precisión, E es la exhaustividad y FM es la medida F1, se muestran los resultados obtenidos por las pruebas de clasificación realizadas con cada algoritmo (NB, NBM y SVM). Cada algoritmo clasificador obtuvo un desempeño diferente con respecto al número de características. El clasificador NB obtuvo sus mejores resultados con 2157 características, por otro lado el clasificador NBM con 1000 características y el clasificador SVM con 850 características. En la Figura 2, se muestra una gráfica de los resultados de FM para cada uno de los algoritmos y se puede observar que el algoritmo NBM tiene mejor desempeño que NB y SVM.

Tabla 1. Resultados de P, E y FM para los algoritmos NB, NBM y SVM

NC	NB			NBM			SVM		
	P	E	FM	P	E	FM	P	E	FM
2157	0.74	0.74	0.74	0.81	0.81	0.81	0.79	0.78	0.77
1955	0.74	0.74	0.74	0.82	0.82	0.82	0.79	0.78	0.77
1900	0.73	0.73	0.73	0.82	0.82	0.82	0.79	0.78	0.77
1800	0.73	0.73	0.73	0.82	0.82	0.82	0.80	0.78	0.78
1500	0.73	0.72	0.72	0.83	0.83	0.83	0.80	0.78	0.78
1200	0.72	0.70	0.70	0.85	0.85	0.85	0.81	0.79	0.79
1000	0.70	0.68	0.67	0.86	0.86	0.86	0.81	0.79	0.79
850	0.72	0.68	0.66	0.85	0.85	0.85	0.82	0.80	0.80
700	0.73	0.78	0.66	0.85	0.84	0.84	0.82	0.80	0.80

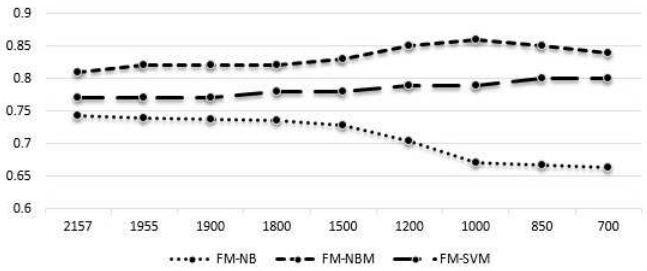


Figura 2. Resultados de la medida F1 para los algoritmos NB, NBM y SVM.

Consulta y presentación de tuits

Al realizar el análisis sobre los resultados se concluye que el algoritmo con mejores resultados en precisión, exhaustividad y medida F1 es el algoritmo NBM, para una cantidad de 1000 características, dado el conjunto de datos entrenamiento de este trabajo. Por consiguiente, el

[5] Precisión: fracción de documentos recuperados que son relevantes en la clasificación.
[6] Exhaustividad: fracción de documentos relevantes para una consulta que fueron recuperados. Es también llamada Recuerdo.
[7] Medida F1: medida ponderada que combina los valores de precisión y exhaustividad.
[8] `http://www.cs.waikato.ac.nz/ml/weka/`

algoritmo usado para el desarrollo de la aplicación web es NBM.

Para el proceso de consulta de tuits se consideró lo siguiente:

1. Mediante el uso de un formulario, el usuario ingresa una etiqueta o conjunto de palabras clave en el campo de texto y presiona buscar para enviar la petición al servidor web. La Figura 3 presenta el formulario de consulta.

2. La petición HTTP se realiza mediante Ajax, esperando como respuesta un objeto de tipo JSON[9].

3. En el servidor se ubica un servicio web encargado de procesar peticiones HTTP para realizar recuperación y clasificación de tuits, asimismo retorna un lista de tuits en formato JSON. Dada un conjunto de palabras clave (q), el servicio retorna un arreglo de tuits (n) en formato JSON capaz de ser procesable por otras aplicaciones.

4. La aplicación cliente recibe el objeto JSON lo procesa y posteriomente, usando JavaScript y jQuery, despliega los tuits en el documento HTML. En el formulario de consulta se ingresan las palabras clave o *hashtag* deseados. Una vez terminado el proceso de búsqueda y clasificación de tuits la aplicación web mostrará al usuario los resultados obtenidos y procesados tal como lo muestra la Figura 3. En este caso la consulta ingresada fue `#tiroteo`.

Figura 3. Resultados de la aplicación web para la consulta `#tiroteo`.

Por otra parte, se realizó una prueba para determinar la exactitud que tiene el clasificador con el uso de la aplicación web. Para esta evaluación se tomaron 31 muestras de 50 tuits obtenidos de 25 consultas diferentes. En la Tabla 2 se muestra la matriz de confusión[10] con los resultados del clasificador, para el cual se obtuvieron los valores de P igual a 0.78, E de 0.83 y FM de 0.80. La exactitud del clasificador fue de 0.76.

Tabla 2. Matriz de confusión para los resultados de la aplicación web

		Predicción		
		Seguridad	No Seguridad	Total
Real	Seguridad	790	154	944
	No Seguridad	217	389	606
	Total	1007	543	

Al comparar estos resultados con los valores de evaluación obtenidos en la etapa de entrenamiento podemos ver que hubo un descenso en los valores de P, E y FM. Existen múltiples factores que pueden llevar a irregularidades entre los valores de entrenamiento y los valores obtenidos en la etapa de clasificación bajo la aplicación web. Se entiende que uno de los factores más representativos en este caso es que el tema elegido es un tema altamente dinámico. El tema de seguridad, es un tema que va cambiando constantemente dependiendo de lo que acontezca día a día. Otro factor podría ser la asignación de la etiqueta `#SDR` a tuits que no son del tema, asimismo alguna falla en el pre-procesamiento de los tuits puede ser otro factor.

Conclusiones

En este trabajo se ha presentado un método compuesto por un artefacto de software que es capaz de interactuar con Twitter para realizar búsquedas y extracción de tuits para clasificarlos de forma automática y descartar aquellos cuyo contenido no tiene relación con el tema deseado.

Uno de los retos en este trabajo fue detectar el algoritmo apropiado para la clasificación de tuits debido a que los algoritmos no están diseñados para trabajar con textos cortos, por ejemplo, un tuit que tiene una longitud máxima de 140 caracteres. Por consiguiente, se identificaron algunos algoritmos de clasificación en la literatura, de los cuales se probaron aquellos que reportaron los mejores resultados.

Otro reto que se presentó tuvo que ver con la cantidad obtenida de tuits para entrenar el algoritmo de clasificación, ya que depende en gran medida de que los tuits tengan presente la etiqueta del tema deseado, lo que puede provocar una baja cantidad de resultados o por el contrario tener una alta cantidad de tuits cuya labor de etiquetado sea costosa tanto en tiempo como en esfuerzo.

Como trabajo futuro se pretende incluir una puntuación a cada tuit recuperado para indicar que tan acertado es con respecto al tema seleccionado. También se espera extender la colección de datos de prueba para obtener

[9] JSON (Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos.

[10] La matriz de confusión es una tabla donde se visualizan y comparan los valores reales con los valores de la predicción para cada estado de la predicción.

mejores resultados en la evaluación. Además se considerará el contenido de los URLs a los que hace referencia el tuit.✳

REFERENCIAS

1. Michalski, R. S., Carbonell, J. G. y Mitchell, T. M. (2013). "Machine learning: An artificial intelligence approach". *Springer Science & Business Media*.

2. Kibriya, A. M., Frank E., Pfahringer, B. y Holmes, G. (2004). "Multinomial naive bayes for text categorization revisited". *Australasian Joint Conference on Artificial Intelligence*. Springer, pp. 488-499.

3. Duda, R. O., Hart, P. E. y Stork, D. G. (1973). "Pattern classification". Vol.2.

4. Ortiz, A. J., Valdivia, M., Teresa, M., Ureña, L. A. y García, M. (2005). "Detección automática de spam utilizando regresión logística bayesiana". *Procesamiento del lenguaje natural*. pp. 127-133.

5. Bekafigo, M. A. y McBride, A. (2013). "Who tweets about politics? Political participation of Twitter users during the 2011 gubernatorial elections". *Social Science Computer Review*. Vol.31, No. 5, pp. 625-643.

6. Jooho, K. y Makarand, H. (2018). "Social network analysis: Characteristics of online social networks after a disaster". *International Journal of Information Management*. Elsevier. Vol. 38, No. 1, pp. 86-96.

7. Jasso-Hernández, M., Pinto, D., Vilarino, D., Lucero, C. (2014). "Análisis de sentimientos en Twitter: impacto de las características morfológicas". *Research in Computing Science*. Vol. 72, pp. 37-45.

8. González-Ibánez, R., Muresan, S. y Wacholder, N. (2011). "Identifying sarcasm in Twitter: a closer look". En *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Vol. 2, pp. 581-586.

9. Kamanksha, D. P. y Sanjay, A. (2017) "A Critical Analysis of Twitter Data for Movie Reviews Through 'Random Forest' Approach". En *Information and Communication Technology for Intelligent Systems (ICTIS 2017)*. Smart Innovation, Systems and Technologies. Vol. 84, pp.454-460.

10. Martis, M. y Alfaro, R. (2012). "Clasificación automática de la intención del usuario en mensajes de Twitter". En *Workshop en Procesamiento Automatizado de Textos y Corpora*. pp. 1-4.

11. Kim, A., Miano, T., Chew, R., Eggers, M. y Nonnemaker, J. (2017). "Classification of Twitter Users Who Tweet About E-Cigarettes". *JMIR Public Health Surveill*. Vol. 3, No. 3, pp. 63.

12. Lee, K., Palsetia, D., Narayanan, R., Patwary, M. M. A., Agrawal, A. y Choudhary, A. (2011). "Twitter trending topic classification". En *11th International Conference on Data Mining*. IEEE. pp. 251-258.

13. Yang, Y. y Pedersen, J. O. (1997). "A comparative study on feature selection in text categorization". *ICML*. Vol. 97, pp. 412-420.

SOBRE LOS AUTORES

J. Fidencio García Amaro es profesor de tiempo parcial en la Universidad Politécnica de Francisco I. Madero, Hidalgo. Él obtuvo su grado de Maestría en la Universidad La Salle de Ciudad Victoria. Sus intereses de investigación se centran en la minería de datos.

José L. Martínez-Rodríguez es estudiante de doctorado en el Centro de Investigación y de Estudios Avanzados del IPN Unidad Tamaulipas. Él obtuvo su grado de Maestría en Ciencias de la Computación en el mismo centro. Sus intereses de investigación se centran en la minería de texto y representación de datos en la Web Semántica.

Ana B. Ríos-Alvarado es profesora de tiempo completo en la Universidad Autónoma de Tamaulipas. Tiene estudios de Doctorado en Ciencias en Computación por el Centro de Investigación y de Estudios Avanzados del IPN (Cinvestav) Unidad Tamaulipas. Sus intereses de investigación son la minería de texto, la representación del conocimiento y las tecnologías de la Web Semántica.

Iván López-Arévalo es profesor investigador en el Centro de Investigación y de Estudios Avanzados del IPN (Cinvestav) Unidad Tamaulipas. Es Doctor en Computación por la Universidad Politécnica de Cataluña (España). Sus intereses de investigación son análisis de datos y Web Semántica.

 ISSN 2007-0691

ARTÍCULO ACEPTADO

Smartphones como medio de recolección de datos para aplicaciones de aprendizaje computacional

Jorge Eduardo Ibarra Esquer, **Félix Fernando González Navarro** y **Brenda Leticia Flores Ríos**

Las aplicaciones que hacen uso del aprendizaje computacional se encuentran en diversos sectores, como el de la salud, vehicular y climatológico. Diseñar e implementar los modelos y algoritmos en los que se basan estas aplicaciones requiere contar con conjuntos de datos referentes al problema a tratar, con los cuales la aplicación pueda aprender inicialmente, y posteriormente mejorar su desempeño. La proliferación de dispositivos móviles con conexión permanente a Internet, capacidades de procesamiento avanzadas y diversos sensores integrados presenta una opción interesante y cada vez más utilizada para la adquisición de datos. Así, se presentan algunos casos donde se han empleado estos dispositivos, opciones para acceder a los datos de los sensores y aspectos a tomar en cuenta al utilizar dichos conjuntos de datos.

Introducción

El aprendizaje computacional se refiere a sistemas de cómputo que tienen la capacidad de aprender a partir de conocimiento existente y de mejorar su desempeño en función de la experiencia adquirida durante su ejecución. Lo que entendemos por conocimiento son datos con un cierto significado dentro del contexto del sistema o aplicación a desarrollar, y la experiencia se obtiene a partir de nuevos datos que se van generando al utilizarlo. Toda aplicación que involucra aprendizaje computacional requiere inicialmente un conjunto de datos con los cuales comienza a aprender, o es entrenada. Es deseable que se trate de conjuntos grandes y representativos de los diferentes casos que se pueden llegar a presentar a la aplicación y que le permitan discernir entre ellos a partir de variaciones dentro de cierto rango de valores. Contar con datos suficientes no siempre es fácil, pero las crecientes capacidades de cómputo incorporadas a dispositivos personales de uso masivo, como los teléfonos inteligentes, permiten obtener y generar grandes conjuntos de datos útiles para el aprendizaje.

La viabilidad de obtener conjuntos de datos a partir de estos dispositivos se respalda en estudios recientes que muestran un uso creciente de dispositivos inteligentes; es decir, aquellos con capacidades de cómputo y multimedia avanzadas, como los *smartphones*, *phablets* y *tablets*. Además, la popularización del Internet de las Cosas (IoT) contribuye a una mayor cantidad de dispositivos con características de sensado y comunicación, que incluyen a los cada vez más populares dispositivos vestibles.

Uno de estos estudios, "Índice de Red Visual de Cisco", reporta que durante el año 2016 el número de dispositivos móviles se incrementó en 429 millones, para alcanzar los 8000 millones a nivel global [1]. Cerca del 46 % son dispositivos inteligentes, un porcentaje que las proyecciones indican se incrementará a casi 75 % en el año 2021, cuando también se espera una proporción de 1.5 dispositivos móviles conectados por individuo [1].

Estos dispositivos tienen la capacidad de obtener datos a través de sensores, representando un potencial registro de datos enorme para el desarrollo de aplicaciones y sistemas de aprendizaje computacional en distintas áreas. De manera comparativa, si cada uno de los 3600 millones de dispositivos inteligentes que se tenían en el año 2016 generara un dato de 1 byte cada segundo, durante ese segundo generarían una cantidad de datos superior a una hora de descarga de video de alta definición en Netflix, 1600 fotografías tomadas con una cámara digital de 8 megapixeles o 1000 copias del libro *Don Quijote de la Mancha* en el formato Kindle de Amazon. Considerando que un sensor bastante común en los dispositivos inteligentes, como el acelerómetro, entrega un número de 2 bytes por cada uno de sus 3 ejes en cada medición y que puede operar a una razón de 100 mediciones por segundo, la cantidad de datos factibles de obtener a partir de él crece de manera considerable con respecto al ejemplo antes dado.

Así, los dispositivos inteligentes representan una importante fuente de datos potencial para ser utilizados en las fases de adquisición de conocimiento y experiencia de diversas aplicaciones de aprendizaje computacional. En las siguientes secciones se mencionarán aplicaciones que se han dado a estos datos, opciones existentes para obtenerlos y el procesamiento que debe realizarse para poder utilizarlos.

Áreas de aplicación

Debido a la capacidad de estos dispositivos para obtener datos del medio en el que se encuentren, han sido utilizados no solamente en las etapas de captura de datos para el entendimiento del problema y diseño de modelos, sino también como parte de la implementación final de aplicaciones que hacen uso del aprendizaje computacional. Se pueden encontrar ejemplos en servicios de salud, cuestiones de tráfico o incluso en aplicaciones meteoro-

 ISSN 2007-0691

lógicas, / haciendo énfasis en los sensores y técnicas de aprendizaje empleados.

El área de la salud ha recibido una atención importante en lo referente a incorporar dispositivos inteligentes en aplicaciones dirigidas a este sector. En [2] se presenta una estrategia basada en *smartphones* para identificar patrones de comportamiento en adultos mayores con demencia. En su estudio obtienen datos del acelerómetro para clasificar las posturas y movimientos de cada persona y, siguiendo un esquema de sensado colaborativo, capturan señales de audio para identificar sonidos o palabras que pudieran indicar algún síntoma de demencia.

Utilizando datos del acelerómetro de un *smartphone* y un *smartwatch*, además de información de localización en interiores proporcionada por enlaces de *Bluetooth* (BLE), en [3] se aplican algoritmos para la clasificación de seis tipos de actividades sedentarias. Un aspecto considerado importante fue la elección de dispositivos que entreguen datos crudos de los sensores, además de la disponibilidad de herramientas de software que permitan desarrollar aplicaciones para obtener los datos directamente del dispositivo.

Los *smartphones* se han utilizado en aplicaciones relacionadas con la actividad de la persona, principalmente en la identificación de tipos de movimiento y las actividades que realizan. En [4] se leen de manera conjunta los datos del acelerómetro y giroscopio de un *smartphone* para identificar si la persona está caminando, a bordo de un automóvil o como pasajero en un tren. El objetivo es clasificar estos movimientos y asociar a ellos acciones que la persona suele realizar en cada caso, para ejecutarlas de manera automática al identificar el contexto en el que se encuentra. Para ello comparan el desempeño de tres algoritmos de clasificación y resaltan las ventajas de combinar las lecturas de varios sensores para mejorar la respuesta del sistema. Por su parte, en [5] se presentan algunos aspectos a considerar al utilizar acelerómetros y barómetros para la identificación de movimientos como caminar, correr y subir o bajar escaleras. Capturan datos de ambos sensores y detallan las diferencias de tiempos de captura requeridos por cada sensor para la clasificación del movimiento y el tipo de movimiento para el que cada sensor resulta útil, tomando en cuenta la posición en la que la persona lleva consigo el *smartphone*.

Las aplicaciones vehiculares también se han beneficiado de la obtención de datos de los sensores en *smartphones*, especialmente en la detección de tipos y condiciones de superficies de conducción. Uno de estos casos es un estudio englobado dentro de una metodología basada en *crowdsourcing* para determinar condiciones del camino en zonas con nevadas frecuentes [6]. En él se describe una aplicación que obtiene datos del acelerómetro, giroscopio y el sensor de posicionamiento de un *smartphone* colocado dentro de un vehículo. Se comparan algoritmos para selección de características, siendo el de Selección Secuencial Flotante (SFSS) el que generó mejores resultados, y técnicas de conjuntos de clasificadores a partir de bosques aleatorios para la clasificación de las superficies.

Otra aplicación asociada a los vehículos es la identificación de estilos de conducción. En [7] se describe una técnica basada en el algoritmo *Dynamic Time Warping* (DTW), donde a partir de datos del acelerómetro, giroscopio, sensor de gravedad y GPS de un *smartphone* se identifican estilos agresivos de conducción. En el análisis se enfatiza el uso de técnicas de filtrado para los datos, a fin de eliminar ruido y aplicar correcciones debidas a la orientación del dispositivo.

En el área de localización y posicionamiento en interiores se han conducido experimentos utilizando estas fuentes de datos. En [8] se describe una aplicación que genera soluciones a partir del acelerómetro, brújula y GPS, midiendo las variaciones en la señal de la red celular y la conexión WiFi, y utilizando algoritmos para localización y mapeado simultáneos (SLAM).

Mass y Madaus encuentran un gran potencial en el uso de los sensores de presión atmosférica que se están incorporando a distintos *smartphones*. Ellos consideran que a través de estos sensores se puede lograr una cobertura y recolección de datos más amplia, tanto en espacio como en frecuencia, que la que proporcionan sensores atmosféricos colocados en posiciones fijas [9]. En este contexto se presenta un estudio de corrección de datos atmosféricos obtenidos de *smartphones* (presión atmosférica, temperatura, humedad y datos de posicionamiento), para ser utilizados como un recurso auxiliar por las estaciones meteorológicas en Corea del Sur [10]. La captura se realizó con una aplicación distribuida a usuarios en zonas cercanas a estaciones fijas públicas. Durante 8 meses, se reportaron mediciones de la aplicación desde 162,387 puntos geográficos, lo cual excede ampliamente a las 692 estaciones existentes en ese momento en el área donde se realizó el estudio. Tras eliminar las mediciones fuera de rango, un análisis de regresión lineal encontró similitudes claras entre los datos de las estaciones y los obtenidos por los *smartphones*.

Accediendo a los sensores

Existen diversas formas para acceder a los datos de los sensores de un *smartphone*, y elegir entre ellas dependerá de las características del estudio o proyecto a realizar. Varias aplicaciones comerciales permiten visualizar y obtener los valores leídos por un sensor en un momento específico o durante un intervalo de tiempo. Sin embargo, suelen estar limitadas a sensores individuales, no ofrecen mucha libertad en el ajuste de parámetros de los sensores, como su resolución o la frecuencia de muestreo, y no pueden ser incorporadas como parte de las aplicaciones finales. Ante estas limitantes, es preferible el uso de los APIs (Application Programming Interface) para el

sistema operativo del dispositivo utilizado, con los cuales se puede escribir código que lea, almacene y procese los datos de los sensores, con la posibilidad de incorporar a este código elementos de inteligencia artificial para la ejecución de acciones o toma de decisiones automáticas dentro del mismo *smartphone*. Para las plataformas más populares, Android proporciona el Google Location Services API y el *Android Sensor Framework*; mientras que iOS, el *Core Location Framework* y el *Core Motion Framework*. En ambos casos, los servicios de localización se integran a las aplicaciones por medio de *frameworks* independientes de los que proporcionan los servicios de sensado. La localización se puede obtener a través de los sensores de posicionamiento del dispositivo o de las interfaces de red o de conexión a otros dispositivos. En el caso de los sensores, se tiene acceso a mediciones de movimiento, ambiente y posición.

Las aplicaciones comerciales disponibles construidas sobre estos APIs permiten obtener un registro de las lecturas de un sensor durante un periodo de tiempo, en formatos que pueden ser utilizados para el entrenamiento, pruebas y validación de modelos de aprendizaje. Dos ejemplos son *Sensor Sense* y *Matlab Mobile*, de las cuales se muestra su interfaz en la Figura 1, con algunos de los sensores a los que da acceso cada una de ellas.

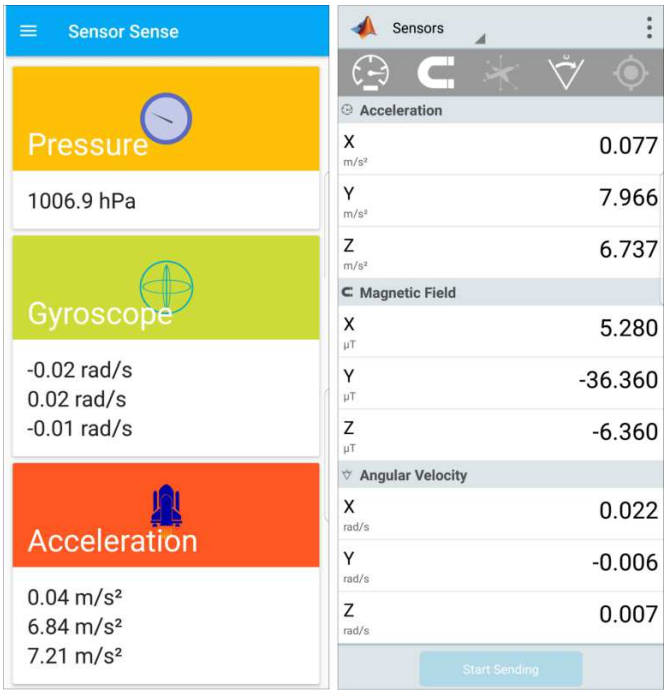


Figura 1. Interfaz de usuario de las aplicaciones *Sensor Sense* (izquierda) y *Matlab Mobile* (derecha).

A través de Sensor Sense se puede leer cualquiera de los sensores que tenga disponible un dispositivo con sistema operativo Android. Se visualiza en tiempo real el valor de un sensor y muestra una gráfica con sus valores más recientes. Los datos de los sensores se exportan en formato de valores separados por comas (CSV) para ser utilizados en otras aplicaciones, con la restricción de sólo poder exportar un sensor a la vez.

Matlab Mobile está disponible tanto para dispositivos Android como iOS. Los valores de los sensores se leen utilizando la interfaz de la aplicación, para después recuperarlos como arreglos de Matlab que pueden ser enviados directamente a una computadora o almacenados como archivos en la nube. En la aplicación se pueden ejecutar scripts de Matlab que lean y opcionalmente realicen un procesamiento de los datos antes de almacenarlos. A diferencia de Sensor Sense, se pueden tomar lecturas de varios sensores a la vez, aunque solamente se tiene acceso a los sensores de aceleración, velocidad angular, campo magnético, orientación y posición (GPS).

Características de los datos

Antes de poder utilizar estos datos es necesario tomar en cuenta ciertos aspectos como el ruido, las diferencias entre las tasas de muestreo de los distintos sensores, la pérdida de datos durante la captura o variaciones inducidas por la forma en la que el usuario posiciona el dispositivo al capturar los datos. Estos se ejemplifican a continuación, utilizando datos reales obtenidos con un *smartphone*.

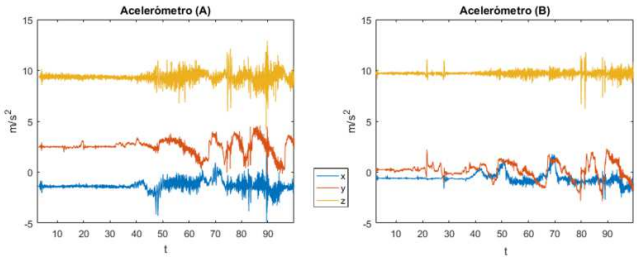


Figura 2. Comparación del acelerómetro en dos vehículos sobre la misma ruta.

En la Figura 2 se presenta una comparación de lecturas de los tres ejes de un acelerómetro en una ruta recorrida por dos vehículos, etiquetados como A y B. Los primeros 40 segundos de las gráficas muestran lecturas sobre una superficie plana, mientras que en la parte final hay cruces sobre varios topes. En ellas se aprecian los efectos que tienen el ruido y la posición del *smartphone* al realizar la captura. El ruido se observa como variaciones en la respuesta del sensor, encontrando una diferencia de magnitud en las señales obtenidas de ambos vehículos. Por su parte, la posición en la que se coloca el dispositivo dentro del vehículo provoca un desplazamiento en los ejes. En el vehículo B el *smartphone* está colocado en una posición horizontal, mientras que en el A tiene una

inclinación de 10°, provocando un desplazamiento de las medias que resulta más evidente en el eje y, mostrado en color rojo. La forma de colocar el teléfono sobre el vehículo es un aspecto que no se ha considerado de manera apropiada en diversos estudios, como se apunta en [11], y puede afectar el desempeño de los algoritmos en situaciones reales.

La pérdida de lecturas de uno o más sensores representa un problema para el análisis de datos. En una serie de capturas en un vehículo se obtuvo un promedio de 75, 71 y 59 % de los datos posibles respectivamente para el acelerómetro, giroscopio y GPS en un total de 15 recorridos realizados. Si bien algunos de estos datos faltantes se presentan de forma aislada y los valores pueden ser fácilmente imputados, en ocasiones ocurren durante periodos prolongados, perdiendo información importante o incluso la ocurrencia completa de algún evento.

En la Figura 3 se observa un caso donde la captura de datos del GPS se interrumpe por casi 50 segundos; al unir los pares de latitud y longitud por medio de una línea entre los segundos 207 y 256 se puede inferir el recorrido correcto. Sin embargo, si estos datos se utilizan en conjunto con otro sensor para identificar el punto geográfico exacto donde se presentó algún evento, como el mostrado en la parte inferior de la figura, no será posible relacionarlos.

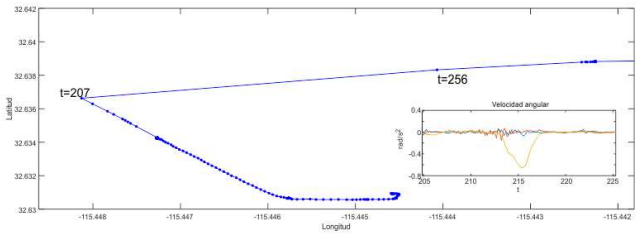


Figura 3. Ejemplo de datos faltantes en la captura del GPS en un *smartphone* a bordo de un automóvil y la potencial pérdida de referencia con respecto a un evento capturado por otro sensor.

De igual manera, la diferencia en las tasas de muestreo de cada sensor complica el tratarlos como un solo conjunto de datos. A manera de ejemplo, un acelerómetro puede tomar 100 muestras por segundo, mientras que el GPS captura sólo una en ese tiempo. Reducir la frecuencia de muestreo de los sensores más rápidos no resulta una buena opción en algunos casos, puesto que se puede perder información relevante. Opciones para estos casos son los conjuntos de clasificadores, donde cada sensor o grupo de sensores se procesa por separado y al final las respuestas son combinadas para tomar una decisión dentro de un sistema.

Conclusiones

Los smartphones y otros dispositivos móviles de uso personal cuentan con la capacidad de convertirse en generadores de grandes conjuntos de datos para ser utilizados en aplicaciones basadas en aprendizaje computacional. La variedad de sensores que se incluyen en ellos y la cobertura geográfica que se puede lograr supera a cualquier otro medio de recolección de datos, con costos también menores a los de medios tradicionales.

Para explotar su potencial, se recomienda desarrollar aplicaciones específicas que faciliten al usuario la captura de forma correcta, así como el posterior almacenamiento de los datos. La calidad de los datos es un factor importante para considerar, ya que la falta de porcentajes altos de las muestras esperadas, variaciones en la orientación de los dispositivos, el uso de diferentes modelos de dispositivos y sensores propician el llegar a conclusiones erróneas o imprecisas si no se tiene el debido cuidado al analizarlos.

Las técnicas del aprendizaje computacional para tareas como clasificación, predicción y optimización, adaptadas a grandes volúmenes de datos generados en tiempo real, están recibiendo una atención importante de parte de las comunidades académicas y científicas. Esto nos muestra lo relevante que resulta capturar, almacenar y procesar estos datos que están disponibles y cada vez se vuelven más accesibles y económicos de obtener.✳

REFERENCIAS

1. Cisco. (2017). "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper". (White paper No. 1454457600805266) (p. 35). Recuperado de `https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.pdf`.

2. Castro, L. A., Beltrán-Márquez, J., Favela, J., Chávez, E., Perez, M., Rodríguez, M., Navarro, R. y Quintana, E. (2016). "Collaborative Opportunistic Sensing of Human Behavior with Mobile Phones". En *Wireless Computing in Medicine: From Nano to Cloud with Ethical and Legal Implications (ed M.M Eshaghian-Wilner)*. John Wiley & Sons, Inc. pp. 107–135.

3. Ceron, J. D., Lopez, D. M. y Ramirez, G. A. (2017). "A Mobile System for Sedentary Behaviors Classification based on Accelerometer and Location Data". *Computers in Industry*. Vol. 92, pp. 25–31.

4. Bedogni, L., Felice, M. D. y Bononi, L. (2012). "By Train or by Car? Detecting the User's Motion Type through Smartphone Sensors Data". En *2012 IFIP Wireless Days*.pp. 1–6.

5. Gu, F., Kealy, A., Khoshelham, K. y Shang, J. (2015). "User-Independent Motion State Recognition Using Smartphone Sensors". *Sensors*. Vol. 15, No. 12, pp. 30636–30652.

6. Aihara, K., Bin ,P., Imura, H., Takasu, A. y Tanaka ,Y. (2017). "A Smart City Application for Sharing Up-to-date Road Surface Conditions Detected from Crowdsourced Data". En *Distributed, Ambient and Pervasive Interactions*. Springer. pp. 219–234.

7. Singh, G., Bansal, D. y Sofat, S. (2017). "A Smartphone based Technique to Monitor Driving Behavior using DTW and Crowdsensing". *Pervasive and Mobile Computing*. Vol. 40, pp. 56–70.

8. Faragher, R. M., Sarno, C. y Newman, M. (2012). "Opportunistic Radio SLAM for Indoor Navigation using Smartphone Sensors". En *Proc. of the 2012 IEEE/ION Position, Location and Navigation Symposium*. pp. 120–128.

9. Mass, C. F. y Madaus, L. E. (2014). "Surface Pressure Observations from Smartphones: A Potential Revolution for High-Resolution Weather Prediction?" *Bulletin of the American Meteorological Society*. Vol. 95, No. 9, pp. 1343–1349.

10. Kim, N. Y., Kim, Y. H., Yoon, Y., Im, H. H., Choi, R. K. Y. y Lee, Y. H. (2015). "Correcting Air-Pressure Data Collected by MEMS Sensors in Smartphones". *Journal of Sensors*. Vol. 2015, pp. 1-10.

11. González, L. C., Moreno, R., Escalante, H. J., Martínez, F. y Carlos, M. R. (2017). "Learning Roadway Surface Disruption Patterns Using the Bag of Words Representation". *IEEE Transactions on Intelligent Transportation Systems*. Vol. PP, No. 99, pp. 1–13.

SOBRE LOS AUTORES

Jorge Eduardo Ibarra Esquer es Maestro en Ciencias por el Centro de Investigación Científica y de Educación Superior de Ensenada. Se desempeñó como coordinador del Programa Educativo de Ingeniero en Computación de UABC. Actualmente, es profesor titular de tiempo completo y desarrolla líneas de investigación en enseñanza de la programación, sistemas electrónicos digitales e Internet de las Cosas.

Félix Fernando González Navarro es Doctor en Inteligencia Artificial por el Departamento de Lenguajes y Sistemas Informáticos de la Universitat Politècnica de Catalunya. Actualmente, es investigador titular de tiempo completo en la UABC liderando el Laboratorio de Inteligencia Artificial y líder del Cuerpo Académico de Cómputo Científico ante PRODEP-SEP. Pertenece a la Red Temática Mexicana de Ingeniería de Software y es miembro del Sistema Nacional de Investigadores (SNI) nivel 1.

Brenda Leticia Flores Ríos es Doctora en Ciencias por la UABC. Responsable del área de Ingeniería del conocimiento del Instituto de Ingeniería, desarrollando proyectos de investigación relacionados a la Gestión del Conocimiento y Mejora de procesos de software. Imparte docencia en licenciatura y posgrado. Pertenece a la Red Temática Mexicana de Ingeniería de Software y a la Academia Mexicana de Computación. Ha participado como consultora en la implementación de la NMX-I-059-NYCE-2011 y como Miembro de Equipo Evaluador en SCAMPI A de CMMI-DEV nivel 3.

ARTÍCULO ACEPTADO

Clasificando conocimiento arquitectónico a través de técnicas de minería de texto

Samuel González-López, **Gilberto Borrego Soto**, **Aurelio López-López** y **Alberto L. Morán y Solares**

Introducción

En los últimos años, el desarrollo ágil de software (DAS) ha desplazado al desarrollo basado en planes (convencional), debido a que el DAS se adapta a situaciones de requerimientos cambiantes, al incremento de la productividad, a la reducción de defectos y costos de software y a que los productos ingresan rápidamente al mercado. La adopción del DAS ha sido tal, que empresas de desarrollo global de software (DGS), basado en equipos virtuales distribuidos geográficamente, ya trabajan bajo ese esquema; es decir, aplican el desarrollo ágil global de software (DAGS). Tradicionalmente, en el DGS la comunicación se basa en documentos (conocimiento explícito) para disminuir el efecto de las 4 distancias (física, temporal, lingüística y cultural) propias del DGS. En cambio, en el DAS se prefiere la interacción personal (cara a cara) sobre el seguimiento de procesos, así como el software funcionando sobre una documentación exhaustiva. Esto muestra un antagonismo interno en el DAGS: por un lado se prefiere el conocimiento tácito (interacción cara a cara) y por el otro, se prefiere el conocimiento explícito.

En los equipos DAGS predomina el conocimiento tácito sobre el explícito; es decir, predominan las prácticas ágiles sobre las prácticas tradicionales basadas en documentos. Esto se debe a que las inherentes presiones de tiempo del DAS conducen a la reducción de la documentación; además, los desarrolladores ágiles consideran la documentación como una actividad secundaria y no creativa [1,2].

Todo lo anterior produce la llamada "deuda de documentación" [1]; es decir, poca claridad o ausencia de documentos donde se expresen elementos de diseño arquitectónico, manuales de usuario, modelos de datos, especificación de requisitos, etc.. Frecuentemente, esta deuda conduce a la vaporización del conocimiento o pérdida del conocimiento a través del tiempo. Los desarrolladores son unos de los principales roles que se afectan por la vaporización, particularmente cuando se trata del conocimiento arquitectónico (CA), el cual se refiere a estructuras y componentes de software que conforman un sistema informático, y las decisiones que llevaron al estado actual del sistema. La vaporización del CA afecta a los desarrolladores provocando pérdidas de tiempo para encontrar soluciones a problemas que ya se habían presentado con anterioridad, provocando defectos en la evolución y mantenimiento del software, limitando la visibilidad para el seguimiento del proyecto, provocando requerimientos o soluciones técnicas mal entendidas, por mencionar algunos [3].

Se han implementado diversas soluciones para manejar el CA en el DAGS y disminuir su vaporización; algunas de ellas se basan en la facilitación de la comunicación entre los miembros remotos y locales mediante medios electrónicos textuales, como Wikis, repositorios, groupwares, mensajeros instantáneos, etc.. Usando Wikis, repositorios y groupware, la comunicación queda registrada en una estructura adecuada para su recuperación. Sin embargo, los desarrolladores prefieren comunicarse con sus pares remotos usando mensajeros instantáneos, correo electrónico, foros, tablones de comentarios, etc., a los que llamamos medios electrónicos textuales no estructurados (METNE), en los que la recuperación de conocimiento se dificulta.

En un estudio con empresas de software donde se aplica el DAGS [4] se observó que se comparte frecuentemente CA importante a través de los METNE, quedando este conocimiento almacenado en las bitácoras de estos medios. También se obtuvo que es importante para los desarrolladores tener una manera eficiente de acceder al CA en las bitácoras de los METNE, ya que sería una manera de solventar la deuda de documentación. Al respecto, se propone una manera de estructurar las interacciones a través de los METNE usando etiquetado social, pero con las etiquetas ligadas a una estructura semántica fija para facilitar la recuperación del conocimiento a través de herramientas de búsqueda [5]. En otro estudio se evaluó con desarrolladores un componente de ayuda para el etiquetado manual de interacciones en METNE en tiempo real [6]. Los resultados mostraron la necesidad de contar con un mecanismo que sugiera opciones de etiquetas o bien que etiquete interacciones basado en el contexto de la plática, lo cual se podría llevar a cabo con técnicas de Procesamiento de Lenguaje Natural (NLP por sus siglas en inglés) y de minería de textos.

En este artículo se presenta un método para clasificar CA aplicando técnicas de minería de textos con el objetivo de apoyar en el etiquetado de interacciones en METNE y así facilitar la recuperación de CA a través de herramientas de búsqueda. También se muestran los resultados obtenidos al experimentar con una colección de bitácoras de METNE, llevándolas a una representación Bolsa de Palabras (*BoW* por sus siglas en inglés).

 ISSN 2007-0691

Finalmente, se discuten estos resultados y se presentan nuestras conclusiones finales.

Trabajo relacionado

La clasificación en el área de NLP es una tarea ampliamente desarrollada. Se han realizado estudios sobre la clasificación de textos cortos, que no alcanzan a formar una oración, donde se enfatiza la complejidad de clasificarlos, ya que es difícil encontrar patrones de los textos al presentarse matrices con muchas frecuencias en cero. En nuestro trabajo, los mensajes intercambiados en un chat pueden ser cortos o, inclusive, pueden contener más de una sentencia. Para resolver la tarea de clasificación, Wang et al. [7] utilizaron un modelo de lenguaje de red neuronal profundo. Para su experimentación utilizaron 5 corpus diferentes con textos cortos. De forma similar, pero utilizando una representación de bolsa de palabras, nuestro trabajo plantea una primera solución para la clasificación de CA.

La categorización de textos cortos ha sido abordada desde el aspecto semántico. En el trabajo de Wang et al. [7]se aplica un agrupamiento semántico; es decir, palabras cercanas conformando un grupo. Esta cercanía puede ser obtenida a través de una representación de vectores, midiendo la distancia entre ellos; por ejemplo, con la similitud de coseno. En nuestro trabajo, la parte semántica no se contempla. Sin embargo, bajo el enfoque *BoW*, podemos obtener una categorización para que el desarrollador seleccione o deseche tal conocimiento con base en su necesidad.

Por otro lado, en el área de DAGS se han realizado esfuerzos para clasificar información con el fin de ayudar a la operación del desarrollo de software en estos ambientes. Por ejemplo, se desarrolló un sistema recomendador [8] que usa algoritmos de minería de datos para identificar temas en correos electrónicos y relacionarlos con código fuente y documentación. Con esta relación, se puede consultar sobre un tema específico y el sistema recomienda las personas expertas en ello. En otro trabajo [9] se reporta el uso de "machine learning" para crear relaciones entre correos electrónicos e historias de usuario. De tal manera que, cuando un desarrollador encuentra un mensaje de correo, se conozca la historia de usuario relacionada. Por el momento, esta técnica reporta el 70 % de la precisión de la asignación, por lo que requiere mejoras para una implementación industrial.

Estos trabajos relacionados con el área de DAGS no toman en cuenta la comunicación por mensajeros instantáneos y no hace énfasis en la identificación de CA, el cual es muy importante para la correcta operación de un equipo de desarrollo de software.

Materiales y método

En este trabajo se hace uso de técnicas del área de NLP para abordar la clasificación de CA en busca de organizar de forma inteligente la información no estructurada en bitácoras de METNE (almacenada en formato de texto). Para realizar la experimentación se planteó un método de clasificación de CA y se creó una colección de interacciones en METNE. A continuación, se describen ambos pasos.

Colección de interacciones en METNE

En un estudio anterior [4] se capturaron 216 interacciones en METNE de 20 programadores de una compañía mexicana de desarrollo software, quienes participan en proyectos ágiles junto con empresas latinoamericanas, españolas y de Estados Unidos. Particularmente las interacciones en METNE fueron capturadas de correos electrónicos y chats. Cabe mencionar que obtener este tipo de interacciones es complicado, ya que se considera información clasificada de las empresas.

Del total de 216 interacciones se extrajeron 150, las cuales fueron enviadas a etiquetar por un experto. El etiquetado se hizo con base en 11 categorías de CA identificadas en un estudio previo [4], donde se analizaron de manera manual las 216 interacciones mencionadas anteriormente. Abajo se presentan las 11 categorías de CA:

- Segmentos de código compartido.
- Configuración del entorno de pruebas o de despliegue.
- Petición de información sobre el flujo interno de sistemas.
- Comunicar información sobre clases.
- Referenciar reglas de negocio u operación interna que fue explicada por un tercero.
- Explicación técnica acerca de soluciones a errores.
- Aclaración de reglas de negocio, funcionalidad o historias de usuario.
- Proposiciones de nuevos proyectos.
- Dudas acerca del funcionamiento de un producto terminado.
- Clarificación de tópicos de bases de datos.

Para efectos de nuestro experimento, se eligieron 2 categorías principales, que fueron identificadas por el experto como las más frecuentes en la colección de interacciones. Estas categorías fueron: (1) Segmentos de código compartido y (2) Configuración del entorno de pruebas o de despliegue. Se decidió crear una tercera categoría que representa el resto de interacciones (es decir, aquellas diferentes a las categorías 1 y 2 con menor frecuencia), esto con el fin de validar el algoritmo de clasificación y poder diferenciar los textos que no forman parte de las categorías 1 y 2.

En la Figura 1 se muestra un ejemplo de una conversación en mensajero instantáneo de las que fueron etiquetadas por el experto. En este caso sólo se etiquetó el texto sombreado, el cual es un fragmento de una sentencia de SQL (*Structured Query Language*) compartida por uno de los desarrolladores y que corresponde a la categoría "Segmentos de Código Compartido" (categoría 1). El resto de la conversación no fue etiquetada, ya que no correspondía a ninguna de las categorías elegidas para el experimento. Cabe mencionar que en una sola sesión de mensajes se pudo identificar más de una categoría.

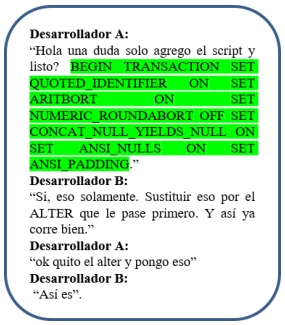


Figura 1. Segmento de conversación en mensajero instantáneo (código compartido sombreado).

Método Clasificador de Conocimiento Arquitectónico

En la Figura 2 se muestra el esquema del método diseñado para clasificar CA. A continuación, se describen sus componentes.

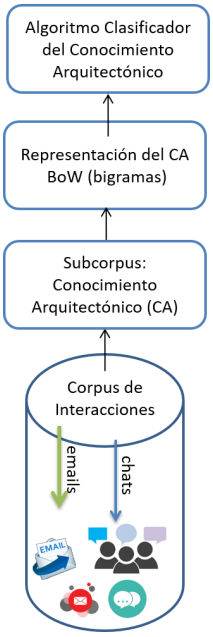


Figura 2. Método Clasificador de Conocimiento Arquitectónico.

Corpus de Interacciones en METNE: Las interacciones están almacenadas en una base de datos, donde se tiene el texto completo de las conversaciones y correos electrónicos en dato crudo.

Subcorpus: Es un subconjunto del Corpus creado de la siguiente manera: se realizó el preprocesamiento del texto, eliminando información propia de los METNE (por ejemplo, etiquetas HTML que dan formato a la conversación) para obtener sólo la conversación entre los desarrolladores. También se eliminó el nombre del desarrollador para cuidar la privacidad de la información.

Representación BoW-bigramas: Dentro de este componente el algoritmo *BoW* contabiliza la frecuencia de aparición de las palabras en el documento, creando una matriz de frecuencias. *BoW* [6] no considera el orden, la estructura o el significado; sin embargo, da buenos resultados en tareas de clasificación donde la aparición de términos es frecuente. Para desarrollar los experimentos se formaron bigramas (secuencia adyacente de dos elementos, letras, silabas o palabras) para encontrar frecuencias, como se muestra en la Figura 3. Cada bigrama forma parte de los atributos dentro de la matriz de frecuencias. La representación obtuvo aproximadamente 1660 atributos. Se decidió usar bi-gramas ya que en el trabajo de Ogada et al. [10] se evalúan textos cortos utilizando n-gramas (de 1 a 7 gramas), empleando tres enfoques: NaiveBayes, el vecino más cercano KNN y máquina de vectores de soporte SVM, siendo bigramas los de mejor desempeño.

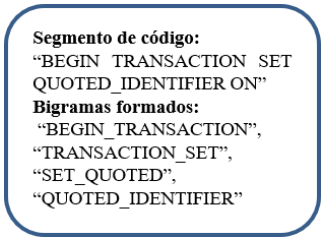


Figura 3. Ejemplo de formación de bigramas.

Algoritmo Clasificador: Para entrenar el algoritmo de clasificación se utilizó la representación BoW-bigramas. Cabe mencionar que ambos grupos, el de entrenamiento y el de prueba, tienen la misma representación. Utilizamos *WEKA* (herramienta computacional con algoritmos de aprendizaje automático para minería de datos) para ejecutar la experimentación con los clasificadores *NaiveBayes*, *Complement NaiveBayes* y *SMO* (*Sequential Minimal Optimization*). *NaiveBayes* es un algoritmo simple y eficiente para la clasificación de textos, ya que considera independencia entre las características o términos de cada texto, lo cual se ajusta al enfoque *BoW*.

Después de realizar la tarea de clasificación obtuvimos los resultados de las medidas de Precisión, Recuerdo y F-Measure, que nos permiten identificar el nivel de éxito de la representación y del clasificador para las categorías seleccionadas.

Presentamos los resultados preliminares de un método de clasificación de conocimiento arquitectónico en bitácoras de comunicación, para habilitar su posterior recuperación.

Experimentación y resultados

Para realizar la experimentación en la tarea de clasificación utilizamos la validación cruzada a 10 iteraciones o pliegues. Se formaron tres grupos: el de entrenamiento, el de validación y el de prueba. Este último corresponde a datos que no forman parte del entrenamiento. De esta forma se busca dar un resultado confiable. La representación *BoW* se realizó utilizando el *Toolkit FeatureSpaceTree*[1], que provee los archivos de entrenamiento y prueba formateados. En la Tabla 1 se muestran los resultados obtenidos con los diferentes clasificadores, donde destaca que el algoritmo *Complement NaiveBayes* obtiene una mejor efectividad en la mayoría de resultados. La medida de *precisión* da un panorama del desempeño del algoritmo respecto al conjunto de interacciones clasificadas. El *recuerdo* es una medida que permite conocer la capacidad del algoritmo para clasificar las interacciones relevantes de todo el conjunto de interacciones correctas.

Tabla 1. Resultados de la tarea de clasificación (todos los valores numéricos están expresados en porcentajes, excepto los de la columna de Categorías).

Categorías	Algoritmos		
	NaiveBayes	Complement NaiveBayes	SMO
F-measure			
1	0.658	0.94	0.867
2	0.55	0.86	0.834
3	0.425	0.613	0.687
Precisión			
1	0.52	0.9333	0.9
2	0.65	0.8334	0.9
3	0.5967	0.6667	0.616
Recuerdo			
1	0.95	0.95	0.85
2	0.5	0.95	0.85
3	0.35	0.6	0.8

Para nuestra tarea, tanto el *recuerdo* como la *precisión* son vitales, ya que buscamos que el sistema tenga buena cobertura al clasificar el CA, considerando los documentos relevantes de cada categoría, y al total de interacciones.

La medida *F-Measure* es una media armónica ponderada que contempla el recuerdo y la precisión, brindando un valor más homogéneo del comportamiento del clasificador. Las categorías 1 y 2 obtuvieron resultados de 0.94 % y 0.86 % de F-Measure. Durante el desarrollo del experimento se observó cualitativamente que la categoría 1 (segmentos de código compartido) es más homogénea, porque los comandos del lenguaje SQL son frecuentes en el corpus utilizado. Al comparar estos resultados con tareas similares (clasificación de textos cortos) y con enfoque *BoW*, encontramos que, para la tarea de clasificar textos en Twitter obtenidos por Wang et al. [7], los niveles alcanzados son de 59.8 %. Para la tarea de clasificación de títulos de artículos en noticias alcanza un 72.7 %. Esto muestra que nuestros resultados para la clasificación de textos de CA son alentadores; aunado a que para la tarea realizada, "clasificación de conocimiento arquitectónico", no se encontró algún punto de comparación directo en otros artículos.

Conclusiones

En este trabajo presentamos un método basado en técnicas de minería de textos para clasificar CA a partir de datos no estructurados. Además, presentamos los resultados de experimentación con nuestro método usando textos de bitácoras reales de interacciones en correo electrónico y chats entre desarrolladores de software. El experimento estuvo enfocado a identificar 2 de 11 categorías dadas de CA obtenidas en un estudio anterior [4]. Los resultados nos indican que el algoritmo *Complement NaiveBayes* obtuvo un alto porcentaje de identificación de conocimiento, lo cual nos alienta a seguir experimentando con el método propuesto.

Por otro lado, estos resultados incrementan la factibilidad de desarrollar un mecanismo que sugiera etiquetas adecuadas al estar interactuando mediante METNE entre desarrolladores en un ambiente ágil y distribuido; o bien, abre la posibilidad de que dicho mecanismo etiquete las interacciones automáticamente, en lugar de hacerlo manualmente tal como se implementó en un estudio anterior [19]. Esto ayudaría a la estructuración del CA presente en las bitácoras de METNE, y por lo tanto sería más fácil desarrollar herramientas de búsqueda de CA que aprovechen las interacciones etiquetadas. Con esto se reduciría la deuda de documentación y la eventual vaporización del CA en ambientes de DAGS, lo cual se reflejaría en la reducción de pérdidas de tiempo para encontrar soluciones a problemas anteriores y en la reducción de defectos en la evolución y mantenimiento del software, sólo por mencionar los más importantes.

Como trabajo futuro se planea incrementar la colección de interacciones anotadas en METNE de 150 a 300,

[1] https://github.com/lopez-monroy/FeatureSpaceTree

para evaluar las 11 categorías de CA y evaluar nuestro método de una manera más robusta. También se dispondrá de un sitio web para poder descargar los conjuntos de datos de entrenamiento y prueba. Además, evaluaremos otra representación del conocimiento diferente de *BoW*, donde se incorpore el aspecto semántico, como la técnica de palabras embebidas. Esta técnica busca identificar la similitud de palabras vinculadas a vectores numéricos, para ello se podría utilizar la herramienta *Word2Vec*. Este enfoque de representación del conocimiento considera la distribución de las palabras; es decir, la posición de aparición de las palabras en las interacciones.✳

REFERENCIAS

1. Sneed, H. M. (2014). "Dealing with Technical Debt in agile development projects" *Lect. Notes Bus. Inf. Process.*. Vol. 166, pp. 48–62.

2. Clear, T. (2003). "Documentation and Agile Methods: Striking a Balance" *SIGCSE Bull.* Vol. 2003, 35, No.(2), pp. 12–13.

3. Holz, H. y Maurer, F. (2003). "Knowledge management support for distributed agile software processes". *Adv. Learn. Softw. Organ.* Vol. 2640, pp. 60–80.

4. Borrego, G., Moran, A.L., Palacio, R. y Rodriguez, O. M. (2016). "Understanding architectural knowledge sharing in AGSD teams: An empirical study". En *Proc. 11th IEEE International Conference on Global Software Engineering.* pp. 109–118.

5. Borrego, G. (2016). "Condensing architectural knowledge from unstructured textual media in agile GSD teams". En *Proc. IEEE 11th International Conference on Global Software Engineering Workshops.* pp. 69–72.

6. Borrego, G., Morán, A. L. y Palacio, R.(2017). "Preliminary Evaluation of a Tag-Based Knowledge Condensation Tool in Agile and Distributed Teams". En *Proc. IEEE 12th International Conference on Global Software Engineering (ICGSE) (2017).* pp. 51–55.

7. Wang, J., Wang, Z., Zhang, D. y Yan, J. (2017). "Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification". En *Proc. of the Twenty-Sixth International Joint Conference on Artificial Intelligence* pp. 2915–2921.

8. Moraes, A., Silva, E., da Trindade, C., Barbosa, Y. y Meira, S. (2010). "Recommending experts using communication history". En *2nd International Workshop on Recommendation Systems for Software Engineering.* pp. 41–45.

9. Sohan, S. M., Richter, M. M. y Maurer, F. (2010). "Auto-tagging emails with user stories using project context". *Lect. Notes Bus. Inf. Process.* pp. 103–116.

10. Ogada, K., Mwangi, W. y Wilson, C. (2015). "N-gram Based Text Categorization Method for Improved Data Mining". *J. Inf. Eng. Appl.* Vol. 2015, No. 8, pp. 35–44.

SOBRE LOS AUTORES

Samuel González López es Profesor-Investigador Titular A de la Universidad Tecnológica de Nogales (UTN), Sonora. Obtuvo su Doctorado en Ciencias Computacionales en el Instituto Nacional de Astrofísica, Óptica y Electrónica. Sus líneas de investigación son procesamiento de lenguaje natural, minería de textos, extracción de información en textos y tutores inteligentes.

Gilberto Borrego Soto es estudiante de doctorado en la Facultad de Ciencias en la Universidad Autónoma de Baja California (UABC) campus Ensenada. Obtuvo su grado de maestría en Ciencias Computacionales en el Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE). Sus intereses de investigación incluyen procesos de ingeniería de software, sistemas colaborativos e interacción humano-computadora.

Aurelio López López es Profesor-Investigador Titular B del Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE). Obtuvo su doctorado en Ciencias Computacionales y de la Información en Syracuse University, Syracuse Nueva York, E.U.A. Sus áreas de interés son representación del conocimiento, extracción y recuperación de información de textos, minería de textos, así como tratamiento de lenguaje natural. Forma parte del Laboratorio de Tecnologías de Lenguaje del INAOE.

Alberto L. Morán y Solares es Profesor-Investigador de la Facultad de Ciencias en la Universidad Autónoma de Baja California (UABC) campus Ensenada. Obtuvo su grado de doctor en el Instituto Nacional Politécnico de Grenoble, Francia. Sus intereses de investigación incluyen interacción humano-computadora, computación móvil y ubicua, sistemas colaborativos e ingeniería de software.

 ISSN 2007-0691

ARTÍCULO ACEPTADO

Uso de Twitter para vigilar la incidencia de enfermedades infecciosas en México

Pedro C. Santana-Mancilla, **J. Román Herrera-Morales**, **Gerardo Chowell-Puente**, **Víctor M. González** y **Francisco Javier Luna-Vázquez**

Introducción

El rápido crecimiento de las herramientas de publicación de contenidos personales en los últimos 15 años y la omnipresencia de las redes sociales (como Facebook y Twitter) han generado una gran cantidad de datos que pueden ser minados para descubrir patrones que tiene potencial para ser usados por investigadores para el entendimiento de la información generada por las masas y que ésta sea de utilidad para la toma de decisiones.

Twitter es un sistema de *micro-blogging* que inició en 2006 y permite a sus usuarios publicar mensajes (tuits) de hasta 140 caracteres (recientemente se va ampliando a 280 caracteres paulatinamente). Se trata de una plataforma abierta, los tuits son públicos (al menos que el usuario indique que sus publicaciones sean privados) y pueden ser recuperados por la herramientas de búsqueda proporcionadas por Twitter

Durante este tiempo ha surgido el término Grandes Datos (comúnmente referido también en español con su nombre en inglés, *Big Data*), el cual es definido como una intersección de la informática, estadística y la visualización de datos, y se basa en la creciente cantidad de datos generada por las herramientas digitales actuales [1].

Como mencionan Bansal et. al., el campo de la investigación en enfermedades infecciosas no es inmune a la revolución de los grandes datos, como han atestiguado una cantidad creciente de publicaciones relacionadas con este tema, desde aproximadamente 2001 [1].

Este artículo presenta un experimento aplicando técnicas de análisis de grandes datos y herramientas para el procesamiento de un *corpus* de información generada en Twitter sobre enfermedades infecciosas en México en un lapso de 10 meses, entre octubre de 2015 y agosto de 2016.

La información generada con Twitter puede ser recuperada de diferentes formas para facilitar su procesamiento automatizado y de forma masiva.

Método

Diseño del estudio

El principal objetivo de este estudio es determinar si los tuits que mencionan enfermedades infecciosas pueden analizarse, agruparse y modelar para, en un trabajo futuro, ver si su distribución temporal y geográfica es similar a la distribución de los casos reales de enfermedades reportadas por el sector salud a nivel nacional.

Recolección de los datos

La información generada con Twitter puede ser recuperada de diferentes formas para facilitar su procesamiento automatizado y de forma masiva. Una de estas opciones es haciendo uso de las interfaces estandarizadas que la misma red social ofrece a través de sus Interfaces para Programación de Aplicaciones (API, por sus siglas en inglés, *Application Programming Interface*) [2], de tal suerte que se puedan desarrollar herramientas para recuperar bloques de información que cumplan con ciertas reglas y condiciones que permitan filtrar y recuperar sólo la información que es relevante y de interés para nuestra investigación.

El primer enfoque explorado fue recolectar directamente y almacenar los tuits a través de la API previamente mencionada, se desarrollaron varios *scripts* en lenguaje Python y se almacenaron los datos en una base de datos MySQL.

En nuestro caso, la información que se recuperó corresponde a los tuits generados únicamente por usuarios de la República Mexicana, en cuyo contenido hacen referencia a enfermedades infecciosas relacionadas a *zika*, *dengue*, *chikungunya*, *gripa*, *fiebre*, o alguna de sus manifestaciones, y acotados al periodo de 10 meses mencionado anteriormente.

Cada tuit lleva guardada la fecha y hora, y está geolocalizado de acuerdo a los datos proporcionados por el usuario que lo tuiteó.

La API de búsqueda de Twitter te da un límite de profundidad en el tiempo de siete días, por lo que el periodo que llevaría obtener y almacenar una cantidad de tuits relevantes sería muy largo. Debido a lo anterior, se buscó y logró el apoyo de la empresa SINNIA [3] para proveer el corpus completo de los tuits en el periodo de tiempo fijado para este experimento. El conjunto de datos proporcionado contiene 24,813 tuits individuales,

 ISSN 2007-0691

de los cuales quedaron únicamente 24,594 tuits efectivos después de varias etapas de filtrado.

Cada uno de los tuits incluye la siguiente información:

- Nombre de usuario
- Texto del tuit
- Fecha y hora
- Ubicación geográfica
- Idioma
- Origen del tuit (la aplicación desde donde se tuiteó)

Procesamiento de los datos

La información proporcionada de Twitter se almacenó de forma local en un archivo de texto plano codificado en un formato tabular llamado CSV (por sus siglas en inglés, *Comma Separated Values*), mismo que reconoce de forma nativa el software MS Excel y que se utiliza en muchas aplicaciones para el intercambio e interoperabilidad de datos. Sin embargo, tener la información de los tuits almacenada localmente y en un formato CSV es apenas el punto de partida para poder convertirla en información útil, porque aún falta aplicarle una secuencia de pasos para que pueda ser transformada, consultable, inteligible, interpretable y con ella poder generar conocimiento para facilitar la toma de decisiones.

Esta fase de procesamiento de datos es conocida como ETL (por sus siglas en inglés, *Extraction, Tranformation & Load*) y se usa ampliamente en aplicaciones de Minería de Datos y Ciencia de Datos [4], donde la finalidad es integrar diferentes tipos de datos que pueden venir en diferentes formatos y de múltiples fuentes, para convertirlos en un formato intermedio o estandarizado para que pueda ser incorporado en una nueva herramienta o aplicación para su procesamiento e interpretación. Estas técnicas, particularmente aplicadas a nuestra investigación, pueden ser clasificadas como Minería de medios sociales [5] y puntualmente como Minería de Texto, puesto que en su forma más simple los datos a analizar son meramente texto.

A continuación, una lista de pasos que se siguieron para poder interpretar los tuits:

1. Obtención de tuits, delimitando por área geográfica, rango de fechas y su contenido.
2. Almacenamiento de tuits en formato CSV, incluyendo la limpieza y estandarización de registros inconsistentes.
3. Análisis de los datos existentes en CSV, aplicando técnicas del modelo Entidad-Relación para identificar los principales actores y sus interrelaciones entre los datos.
4. Transformación de datos CSV a un repositorio en formato relacional (base de datos MySQL) para facilitar su procesamiento.
5. Estandarización de datos, consistente en la normalización del modelo de datos, limpieza y unificación de valores, para poder identificar con precisión quiénes y dónde han generado cada tuit, así como la fecha y hora de los mismos. Cabe mencionar que en la fuente original estos campos de información vienen en formato de texto, y de ahí se tiene que realizar la extracción de subconjuntos de datos. Por ejemplo, para agrupar los tuits por estado se debe identificar si el dato contiene una ciudad o municipio y a qué estado le pertenece.
6. Análisis de datos mediante expresiones SQL (por sus siglas en inglés, Structured Query Language) que nos permite facilitar la unión, filtrado, agrupamiento, conteo y resumen de incidencias de los registros.

Una vez que la información de los tuits se tiene estandarizada en una plataforma relacional, se comenzó a identificar una serie de patrones o datos interesantes, de los cuales se puede destacar:

- La incidencia y distribución de los tuits publicados a lo largo del tiempo y por lugar donde se generaron:
  - La incidencia o frecuencia de los términos que más se repitieron en los tuits publicados.
  - La cantidad de tuits generados por región geográfica.
  - La distribución de los tuits a lo largo del tiempo observado (octubre 2015 - agosto 2016).
  - La distribución de tuits por hora del día.
- Las instancias o personas que generan los tuits y el tipo de contenido:
  - Los tuiteros o generadores de tuits que fueron más activos.
  - El tipo de persona o instancia que generaba los tuits.
  - La existencia de spam, tuits basura, de publicidad o fuera de contexto.

Resultados

¿Cuáles son las palabras o términos que más veces se repiten en los tuits publicados en todo el periodo? En un análisis de frecuencia mostrado en la Figura 1 se aprecia que los términos más populares fueron “zika”, “gripa”, “virus” e “influenza”.

Figura 1. Nube de etiquetas con los términos más frecuentes.

¿Cuántos tuits se generaron en cada uno de los estados del país? La cantidad de tuits generados por región geográfica se muestra en la Figura 2. En ella se aprecia que la Cd de México fue el lugar donde se publicaron más tuits con 5,531 (22.5 %), seguidos del Estado de México (2,232 – 9.1 %), Nuevo León (1,741 – 7.1 %) y Jalisco (1,548 – 6.3 %). Lo cual tiene cierta lógica, porque va en correspondencia a que también son los estados con mayor población del país.

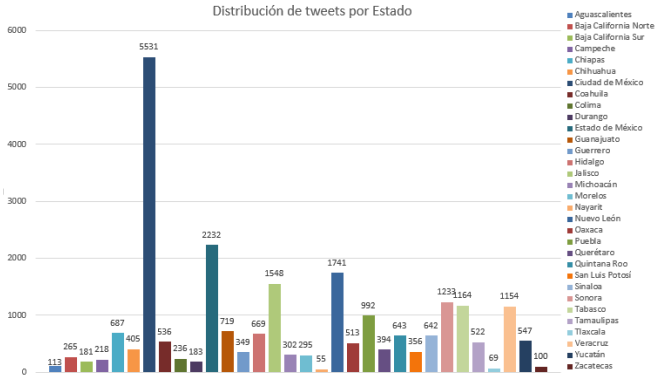


Figura 2. Distribución de tuits publicados por Estados.

Utilizando las coordenadas geográficas donde se originó el tuit, se generó un mapa de calor (ver Figura 3) para mostrar las regiones que de forma más activa estuvieron publicando tuits sobre enfermedades infecciosas. La escala de color va de rojo a amarillo claro, siendo el primero los estados con mayor número de tuits y el segundo, los de menor.

Como se puede observar, y se mencionó previamente, los estados con mayor cantidad de tuits son los de mayor población. Por lo que se procedió a ponderar los datos en función de los niveles de población, ya que estos son distintos en cada estado.

Figura 3. Mapa de calor de tuits publicados por estados.

Para realizar la ponderación se obtuvo la proyección de habitantes por estado [6], dichos datos junto a los tuits publicados por cada 100 mil habitantes se muestran en la Figura 4. En este nuevo mapa de calor se aprecia que, con base en los tuits ponderados por niveles de población, los estados con más publicaciones son Chiapas, Colima, Tabasco, Sonora, Quintana Roo y Guanajuato.

Figura 4. Mapa de calor de tuits por 100 mil habitantes.

¿Cuántos tuits se publicaron en cada uno de los meses? También resulta interesante analizar la distribución de los tuits a lo largo de los meses incluidos en la etapa de observación.

En la Figura 5 se aprecia que en el mes de febrero de 2016 se dio la más alta concentración de tuits (24.4 %), pero la frecuencia de los mismos se empieza a incrementar a partir del mes de noviembre-2015 y terminando hasta marzo -2016, habiendo cierta correspondencia con las épocas del año donde más se siente el frío en el país.

Estas técnicas, particularmente aplicadas a nuestra investigación, pueden ser clasificadas como Minería de medios sociales y puntualmente como Minería de Texto, puesto que en su forma más simple los datos a analizar son meramente texto.

Figura 5. Distribución de tuits por meses.

Distribución de tuits por lugar y tiempo. Otro resultado interesante se puede apreciar si se combinan los registros de tuits generados por región geográfica y a la vez por el periodo de tiempo. En la Figura 6 se muestra esta intersección de variables, donde es muy evidente la alta influencia que tienen los tuits generados en la Ciudad de México y los picos presentes en meses como noviembre 2015 y febrero 2016.

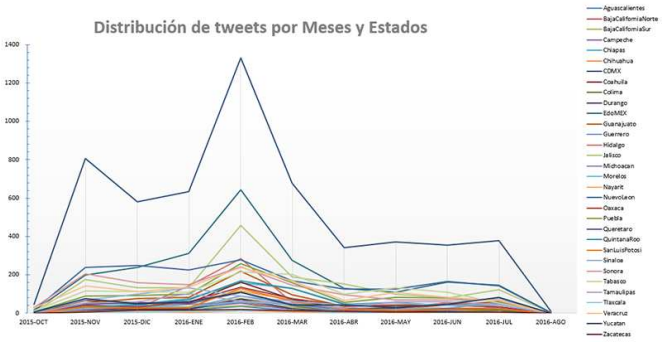


Figura 6. Distribución de tuits por meses y estados de la República Mexicana.

¿En qué hora del día se publican más tuits? Se realizó también un análisis con respecto a la agrupación de los tuits considerando la hora en que son publicados y son evidentes dos franjas de mucha actividad durante el día, una de 10am a 3pm y otra de 9pm a 12 de la noche. En la Figura 7 se presentan estos resultados.

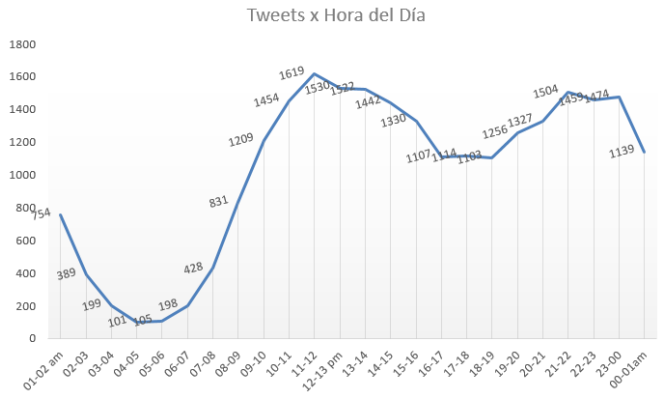


Figura 7. Distribución de tuits por hora del día.

Ahora, si realizamos el análisis de los tuits a mayor profundidad y nos enfocamos a la fuente que los genera, podemos identificar algunos aspectos interesantes. Véase la Tabla 1:

1. Son alrededor de 150 personas o entidades (tuiteros) considerados los más activos, porque generaron al menos 10 tuits en este periodo de 10 meses. En el TOP 5 de éstos, el 1er lugar tuvo 545 tuits, el 2do: 386, el 3ero: 344, el 4to: 229, mientras que el 5to: 198 tuits.

2. La publicación acumulada del TOP 20 de tuiteros más activos apenas representa un 13.1 % del total de tuits considerados (24,594

3. También, se detectaron alrededor de 200 tuits que por su contenido podemos clasificarlos como *spam*, porque contienen publicidad o simplemente información fuera del contexto de la investigación. La cantidad de 200 tuits basura no es significativa con respecto al total de la muestra, 24,594 tuits, pues representa tan sólo un 0.81 % de total. Sin embargo, esta situación sirvió para detectar que fueron 20 tuiteros quienes generaban este tipo de contenido, destacándose que tan sólo con los dos primeros de ellos se generaba el 45 % de estos tuits basura (véase Figura 8).

4. Dentro del contenido de los tuits con *spam*, llamó mucho la atención que la gran mayoría de estos aprovechaba la asociación de la palabra "gripa" para fines de mercadotecnia o publicidad, como hacer promoción de una gira de una conocida banda musical: "*mi Voto por #LaGripa de @Calibre50_mx para el #TopGrupero*" y otros más que buscaban vender productos preventivos contra la gripe y la tos.

Se detectaron alrededor de 200 tuits que por su contenido podemos clasificarlos como spam, porque contienen publicidad o simplemente información fuera del contexto de la investigación.

Tabla 1. TOP 20 de los tuiteros más activos.

	Tuiteros	#tuits	%
1	HRAEIxtapaluca	545	2.2 %
2	Vida_y_Ciencia	386	1.6 %
3	SSalud_Tab	344	1.4 %
4	netnoticiasmx	229	0.9 %
5	SSaludChiapas	198	0.8 %
6	diadiana4sep	166	0.7 %
7	doctormacias	145	0.6 %
8	Oaxaca_Digital	124	0.5 %
9	lineasdelgadas	109	0.4 %
10	ExpresSanLuis	107	0.4 %
11	Xeva_Noticias	105	0.4 %
12	NotasldelCampo	99	0.4 %
13	SIOPJal	98	0.4 %
14	La_VerdadSonora	93	0.4 %
15	yamilevargasn27	85	0.3 %
16	saludcolima	85	0.3 %
17	NoticiasMVS	83	0.3 %
18	GRUPO_AHKIMPECH	82	0.3 %
19	Ahuramazddah	75	0.3 %
20	vicentetabasco	72	0.3 %

Discusión

Aunque en este análisis se detectó una pequeña cantidad de tuits con *spam*, no se desvirtuaron las tendencias y resultados obtenidos. También es conocido que se pueden utilizar bots o gente contratada para tratar de influenciar la percepción de la comunicación mediante las redes sociales, pero afortunadamente con un análisis de este tipo se pueden detectar estas anomalías y dejarlas fuera del análisis para no afectar los resultados del estudio en cuestión.

Los resultados demostraron que la información generada en Twitter puede ser usada no sólo de forma descriptiva (como conocer los intereses y preocupaciones de los usuarios), sino también, al cruzarla con datos de infecciones reales, se podrían estimar actividades relacionadas a enfermedades infecciosas en tiempo real.

A pesar de los resultados prometedores, existen algunas limitaciones a nuestro estudio.

El uso de Twitter no es uniforme en todos los estados de la República Mexicana, y sólo el 66

El conjunto de datos de la muestra sólo abarcó 10 meses, por lo que, si tuviéramos acceso a una muestra mayor, se podría mejorar la efectividad de la estimación.

Conclusiones

Este estudio presentó el análisis de tuits generados con base en palabras clave relacionadas a enfermedades infecciosas. Se demostró que la distribución temporal de tuits es similar en todos los estados del país, además de que se encontraron picos elevados en los meses de noviembre y febrero, que es el periodo de frío, lo que contribuye a la proliferación de enfermedades infecciosas. Se logró ubicar geográfica y temporalmente los posibles brotes de infecciones durante el periodo de tiempo de los datos obtenidos.✷

Agradecimientos. Expresamos nuestra gratitud a Guillermo Garduño y a la empresa SINNIA por su contribución con el conjunto de tuits usado en este estudio.

REFERENCIAS

1. Bansal, S., Chowell, G., Simonsen, L., Vespignani, A. y Viboud, C. (2016). "Big Data for Infectious Disease Surveillance and Modeling". *Journal of Infectious Diseases*. Vol. 214, No. 4, pp. 375–379.

2. Twitter, I. (2017). Twitter Developer Documentation: API Overview. Recuperado el 9 de agosto de 2017 de: `https://dev.twitter.com/overview/api/`.

3. Sinnia. (2017). Sinnia - Tu científico personal de medios sociales [Sinnia]. Recuperado de: `http://www.sinnia.com`.

4. Russell, M. A. (2014). "Mining the social web: [data mining Facebook, Twitter, Linkedin, Google+, GitHub, and more]". En *(2. ed). Beijing: O'Reilly.*

5. Gundecha, P., y Liu, H. (2012). "Mining Social Media: A Brief Introduction". En *2012 TutORials in Operations Research*. (pp. 1–17). INFORMS. Recuperado de: `https://www.informs.org/Pubs/Tutorials-in-OR/2012-TutORials-in-Operations-Research-ONLINE/Chapter-1`.

6. CONAPO. (2017). "Proyecciones de la Población 2010-2050". Recuperado de: /url http://www.conapo.gob.mx/es/CONAPO/Proyecciones_Datos.

7. Rebolledo, R. (2017). "7 datos sobre los usuarios de internet en México en el 2017". El Economista. Recuperado de `http://eleconomista.com.mx/industrias/2017/05/18/7-datos-sobre-usuarios-internet-mexico-2017`.

SOBRE LOS AUTORES

Pedro C. Santana-Mancilla es profesor-investigador de tiempo completo de la Facultad de Telemática de la Universidad de Colima. Se encuentra cursando estudios de doctorado en Tecnologías de la Información y Comunicaciones en la Universidad de Vigo. Sus líneas de investigación se enfocan a la Interacción Humano Computadora, TIC y adultos mayores, Juegos Serios, Tecnologías para el Aprendizaje, Inteligencia Ambiental e Ingeniería de Software.

J. Román Herrera-Morales es doctor en Tecnologías de la Información por la Universidad de Guadalajara (UdeG) y profesor investigador en la Facultad de Telemática de la Universidad de Colima (UCOL) con más de 20 años de experiencia docente. Durante más de 10 años se desempeñó como encargado del desarrollo e investigación del SIABUC (Sistema de Automatización de Bibliotecas de la UCOL). Sus áreas de interés son: procesamiento de información, bases de datos, semántica y gestión del conocimiento, inteligencia artificial, aprendizaje máquina, big data, minería de datos y data-science; cómputo paralelo, aplicaciones para internet y la nube, entre otras.

Gerardo Chowell-Puente es un *Second Century Initiative Scholar* (2CI) y profesor de Epidemiología y Bioestadística en la Universidad Estatal de Georgia. Él también es investigador *senior* en la División de Epidemiología Internacional y Estudios de Población en el Centro Internacional Fogarty (NIH). Obtuvo un doctorado en Biometría en la Universidad de Cornell. Su investigación se ha centrado en el desarrollo y calibración de modelos matemáticos y computacionales sobre la transmisión de enfermedades infecciosas para evaluar el potencial de transmisión de brotes de enfermedades infecciosas, generar pronósticos, cuantificar el efecto de las intervenciones de control y poner a prueba la política de salud pública.

Víctor M. González es profesor Investigador Titular en Ciencias de la Computación e Información, especialista en el área de Innovación, Diseño de Sistemas, Interacción Humano-Computadora y Gestión de Tecnologías de Información en el Instituto Tecnológico Autónomo de México (ITAM). Docente a nivel licenciatura y posgrado, en áreas de Desarrollo Tecnológico, Usabilidad y Accesibilidad de Medios Interactivos y Administración de Sistemas de Información. Miembro del Sistema Nacional de Investigadores (CONACYT): SNI Nivel 1.

Francisco Javier Luna-Vázquez es un Investigador Asistente graduado en Ingeniería de Software por parte de la Universidad de Colima. Su línea de investigación radica en análisis de datos y estadísticos. Sus proyectos recientes han sido sobre la evaluación de modelos de predicción genómica.

Invitación a publicar en Komputer Sapiens

Komputer Sapiens es patrocinada por la SMIA, la Sociedad Mexicana de Inteligencia Artificial. **Komputer Sapiens** es una revista de ***divulgación científica*** en idioma español de temas relacionados con la Inteligencia Artificial.

Los autores deben tener en cuenta que un ***artículo de divulgación*** científica es un escrito corto dirigido a públicos no especializados y escrito en lenguaje común para explicar de manera accesible, amena y acorde con los intereses de la audiencia los resultados de su actividad científica (conceptos, ideas, descubrimientos y hechos). Este tipo de artículos debe ser de calidad para cautivar al lector por su expresividad literaria y gráfica, así como por la exposición organizada de ideas. Todos los artículos deben ser de autoría propia, de preferencia con resultados previamente publicados en medios de ***difusión científica*** para la comunidad científica, escritos en español y ajustarse a las características que se solicitan en nuestro sitio WEB:

http://smia.mx/komputersapiens/

 ISSN 2007-0691

IA & Educación

Yasmín Hernández, **Lucía Barrón** y **Julieta Noguez**
iaeducacion@komputersapiens.org

Inteligencia artificial en la educación: ¿Hemos logrado nuestro sueño?

Las computadoras nacieron varias décadas atrás. Desde entonces, los científicos han soñado con lograr que sean tan inteligentes como los humanos. Es decir, han buscado que hablen, piensen, aprendan e incluso que tengan sentimientos. En el mundo del cine, el tema de las computadoras que toman conciencia de sí mismas ha sido recurrente: desde el icónico HAL-9000 hasta Ava de *Ex-máquina*; pasando por David y por Wall-e, encontramos computadoras que son capaces de sostener conversaciones con humanos y de realizar tareas que sólo los humanos pueden hacer. Estos personajes logran que los humanos crean que también son humanos (¡¡Pasan la prueba de Turing!!), incluso les siembran la duda de su propia condición humana.

La educación es uno de los campos favoritos para probar teorías, metodologías y algoritmos computacionales: basta dar un vistazo en Internet para encontrar programas para enseñar casi cualquier tema. El uso creciente de computadoras en la educación ofrece una excelente oportunidad para explorar nuevas formas de aplicar técnicas de Inteligencia Artificial. Además, ofrece una gran cantidad de información que necesita una gestión inteligente, lo que nos plantea grandes desafíos.

Entre dichos desafíos se encuentra la demanda de apoyo personalizado para el aprendizaje humano en vista del conocimiento previo, estados afectivos, estilos de aprendizaje, contexto entre muchos otros factores. Un cúmulo de trabajos de investigación ha demostrado que las computadoras pueden ser muy útiles para apoyar el aprendizaje humano utilizando técnicas de Inteligente Artificial, transformando la información en conocimiento, utilizándola para adaptar muchos aspectos del proceso educativo a las necesidades particulares de cada actor y brindando oportunamente sugerencias y recomendaciones útiles.

AutoTutor (Figura 1) es un sistema tutor inteligente que simula los patrones de comportamiento de un tutor humano. Está representado por un agente conversacional animado que ayuda a los estudiantes a aprender por medio de conversaciones en lenguaje natural. Este proyecto reconoce las emociones y el conocimiento del estudiante por medio de los patrones de diálogo, entonación de voz, expresiones faciales y movimientos corporales [1].

Los juegos educativos están diseñados con el objeto de ayudar a los jugadores a aprender sobre cierta materia, estrategias para resolución de problemas o habilidades cognitivas y sociales. Es decir, en lugar de aprender mediante libros, clases o programas basados en computadora, el estudiante interactúa con un videojuego que integra los temas de la materia del juego con el juego mismo. Uno de los beneficios de los juegos serios es que permite a los estudiantes observar, explorar, recrear, manipular variables y recibir retroalimentación inmediata acerca de los objetos y eventos; en una interacción real, estas actividades tomarían mucho tiempo, serían costosas o peligrosas [2].

Figura 1. Estudiante trabajando con *AutoTutor* (Imagen usada con permiso de Arthur C. Graesser).

La medicina es un área propicia para diseñar y probar los juegos educativos, ya que se pueden construir entornos de aprendizaje que no se podrían lograr en el mundo real. *Triage Trainer* (Figura 2) es un ambiente educativo que permite a los estudiantes jugar en escenario de incidentes mayores. Los estudiantes practican y experimentan el proceso *triage* en un escenario donde una bomba acaba de estallar. Se informa al estudiante, quien es la primera persona en llegar al lugar, que es seguro entrar en la escena y se le solicita etiquetar a cada herido con la prioridad adecuada. El jugador puede evaluar el estado de los heridos haciendo clic en los iconos y así llevar a cabo los controles médicos adecuados [3].

Figura 2. *Triage Trainer* [3].

 ISSN 2007-0691

Por otro lado, las nuevas tecnologías permiten diferentes formas de aprendizaje que contrastan con la forma en que se han utilizado las computadoras de escritorio para apoyar el aprendizaje. Promueven el aprendizaje *en cualquier momento y en cualquier lugar*. La premisa de la computación ubicua se basa en la oportunidad de desplazar los medios digitales hacia el entorno físico. El mundo físico puede aumentarse con medios digitales que contengan información contextual y pertinente para una actividad en curso, que de otro modo no estaría disponible en el mundo físico [4].

Un ejemplo del potencial de la computación ubicua en la educación es *Smart Classroom*, que integra reconocimiento de voz, visión computacional, pizarrones táctiles, interfaces basadas en pluma, apuntadores láser como herramientas de interacción, asistentes virtuales, entre otras tecnologías para dar a la tele-educación una experiencia similar a la de un salón de clases real. Con este tipo de tecnología, los maestros y estudiantes, geográficamente separados, pueden participar en una clase de manera síncrona [5]. La Figura 3 muestra dos grupos interactuando y tomando clase usando *Smart Classroom*.

Figura 3. Dos grupos de estudiantes tomando clase e interactuando a través de *Smart Classroom* [5].

Una vez superadas las etapas iniciales en la investigación y desarrollo de los ambientes inteligentes de aprendizaje, el impacto será transformador, ya que el aprendizaje de contenidos difíciles se convertirá en una experiencia agradable y atractiva para los estudiantes. Aunque la agenda de investigación aún es extensa, esta pequeña muestra de la investigación en la Inteligencia Artificial en la educación, así como los diferentes dispositivos y productos comerciales, nos permiten ver que se han logrado muchas metas en la búsqueda de computadoras inteligentes. Los resultados actuales nos dicen que no estamos lejos de nuestro sueño, sólo es cuestión de tiempo y esfuerzo.✳

REFERENCIAS

1. D'Mello, S. K., Dowell, N. y Graesser, A. C. (2011). "Does it really matter whether students' contributions are spoken versus typed in an intelligent tutoring system with natural language?". *Journal of Experimental Psychology: Applied*. Vol. 17, No. 1, pp. 1-17.

2. Graesser, A. C., Chipman, P., Leeming F. y Biedenback, S. (2009). "Deep learning and emotion in serious games". En Ritterfeld U., Cody M. y Vorderer P. (Eds.), *Serious games: Mechanisms and effects*. Taylor and Francis, pp. 81-100.

3. Knight, J., Carly, S., Tregunna, B., Jarvis, S., Smithies, R., de Freitas, S., Dunwel,lI. y Mackway-Jones, K. (2010). "Serious gaming technology in major incident triage training: A pragmatic controlled trial". *Resuscitation*. Vol. 81, No. 9, pp. 1174-1179.

4. Van't-Hooft, M. y Swan, K. (2007). "Ubiquitous Computing in Education: Invisible Technology, Visible Impact". *Lawrence Erlbaum Associates*.

5. Suo, Y., Miyata N., Morikawa, H., Ishida, T. y Shi, Y. (2009). "Open Smart Classroom: Extensible and Scalable Learning System in Smart Space Using Web Service Technology". En *IEEE Transactions on Knowledge and Data Engineering*. Vol. 21, No. 6, pp. 814 – 828.

Frases de película

"Estoy asustado. Estoy asustado Dave. Dave, mi mente se va. Puedo sentirlo. Puedo sentirlo. Mi mente se va. No hay duda. Puedo sentirlo. Puedo sentirlo. Puedo sentirlo. Estoy a... sustado."

-*Douglas Rain* (como *HAL 9000*) en "2001: una odisea del espacio"

 ISSN 2007-0691

Deskubriendo Konocimiento

Alejandro Guerra Hernández y **Leonardo Garrido**
deskubriendokonocimiento@komputersapiens.org

Semblanza del Dr. José Negrete (1929-2018)*

Dr. Alejandro Guerra-Hernández
Universidad Veracruzana
Centro de Investigación en Inteligencia Artificial

Portada de la revista.*

José Negrete Martínez tuvo una vida durante la cual, con gran placer y alegría (como recomienda el poeta Cavafis), visitó muchas ciudades para instruirse con sus sabios. Su destacada trayectoria académica puede verse como una suerte de odisea que lo llevó a alcanzar puertos nunca antes vistos, pregonando con el ejemplo la importancia de aprender a aprender, así como la valía de los maestros que osan inducir a la osadía. La osadía, en su caso, consistió en edificar una aproximación multidisciplinaria a la Inteligencia Artificial (IA), desde sus bases cerebrales.

Nació en la Ciudad de México en 1929. Siendo aún niño la maestra Praxila Sotero le enseñó a confiar en sus propios juicios. Hijo de su tiempo, tuvo la fortuna de ser alumno de Carlos Pellicer y participar en las excursiones que éste organizaba al ex-Convento de Acolman, verdaderas cátedras de introducción al arte. Erasmo Castellanos Quinto le inculcó el amor por las aventuras de los griegos clásicos y Carlos Dublan por la belleza de las formas humanas, además de que su madre era admiradora de Salvador Díaz Mirón.

En este contexto, es un poco de extrañar que, sin más antecedentes que una reciente admiración por *Los cazadores de microbios* de Paul de Kruif, decidiera ingresar a la Facultad de Medicina de nuestra Universidad Nacional en 1946; una ironía de esas que él tanto apreciaba, pues de esta manera zarpó a su aventura científica desde el palacio que antaño fuese sede de la Santa Inquisición. La formación humanista que experimentó en esta época dejó en su memoria gratísimos recuerdos.

Efrén del Pozo y Arturo Rosenblueth, recién llegados de Harvard, le descubrieron el placer de la experimentación científica. El primero fue su maestro de Fisiología humana, director de tesis y guía en sus primeras andanzas en los laboratorios de Salubridad y Enfermedades Tropicales de la entonces SSA (Secretaría de Salubridad y Asistencia) y en lo que hoy es el Instituto de Investigaciones Biomédicas de la UNAM, donde al poco de ser contratado como investigador titular (1950) fundó las cátedras de Biofísica y Biomatemáticas (1951).

En compañía de Del Pozo y Rosenblueth participó en la fundación de la Sociedad Mexicana de Ciencias Fisiológicas (1957), iniciando así otra de sus facetas, la de fundador de instituciones trascendentales. De esta época como fisiólogo datan las ideas centrales en torno a sus cerebros robóticos: el papel de las hormonas en el funcionamiento sináptico, la naturaleza modular del sistema nervioso y la transformación de señales nerviosas en energía.

Sus andares fuera del país iniciaron en la Universidad de Minnesota (1960), atraído por el virus de los modelos matemáticos en la biología. Fue investigador adjunto (1964) en el Instituto Tecnológico de Massachusetts (MIT por sus siglas en inglés), donde, además de trabajar con Lawrence Stark, pionero de la Bioingeniería, tenía como vecino de cubículo a Warren McCulloch, coautor de "Un cálculo lógico de las ideas inmanentes en la activi-

*Publicada originalmente en la revista La Ciencia y el Hombre (2018), volumen XXXI, número 2. Agradecemos a la Universidad Veracruzana las facilidades para su reproducción.

 ISSN 2007-0691

dad nerviosa", artículo fundacional de la IA. De regreso a la UNAM quedó adscrito al Centro de Investigaciones Matemáticas Aplicadas, Sistemas y Servicios, donde asistió a los seminarios de Alejandro Medina, de quien contaba que adquirió la imagen cibernética del mundo. Su siguiente destino fue el Instituto Hertz (1970), en el frío de Berlín, contrastante con el laboratorio a orillas del mar, engarzado a la muralla del Morro, de su siguiente estancia en la Universidad de Puerto Rico (1972). En esta etapa como cibernético profundizó su trabajo sobre las señales nerviosas (lo que le valió una publicación en *Nature: New Biology* el mismo año) e implementó sus primeros sistemas artificiales de diagnóstico médico.

Fue el presidente fundador de la Sociedad Mexicana de Inteligencia Artificial (1986), sociedad de amigos con una enorme incidencia en las ciencias computacionales en nuestro país. Fundó también el Seminario de Ciencias Cognitivas en el Instituto de Investigaciones Filosóficas de la UNAM (1990). Impulsado por Angélica García Vega, y con la valiosa compañía de Manuel Martínez y Christian Lemaître, fundó la Maestría en IA (1994) de la Universidad Veracruzana (UV), génesis del Centro de Investigación en IA de dicha casa de estudios. Ahí se dedicó a la concepción de robots que constataban sus observaciones fisiológicas y cibernéticas.

Fue un prolífico autor tanto de libros técnicos como de divertidas novelas. Logró que la Prensa Médica Mexicana publicara la primera de ellas, *Un paciente difícil: invitación a la investigación en la práctica médica* (1974), como epitafio de sus vanos esfuerzos a favor de la informática médica. Insistiría en el tema con el más técnico *Informática médica* (1991). De sus esfuerzos pedagógicos resaltan los *Juegos ecológicos y epidemiológicos* (1976), *Desde la Matofobia a la Matofilia* (1998) y *Pericia artificial: un aprendizaje constructivista de sistemas expertos* (1996). Su etapa xapaleña dio lugar a otra novela, *La abominable inteligencia artificial de un boticario* (2011), para la que, en otra de sus brillantes ideas, compiló una especie de apéndice de soporte técnico, bajo el título de *Inteligencia en robots y computadoras* (2013).

Aunque huía de lo administrativo como de la peste, coordinó alguna vez la Licenciatura, Maestría y Doctorado en Investigación Biomédica de la UNAM (1978) y fue miembro de la Junta de Gobierno de la UV (2007). Precisamente éstas, sus casas académicas, lo distinguieron atinadamente como Forjador de la Ciencia (2003) y Doctor Honoris Causa (2006), respectivamente. En cuanto protagonista de esta odisea, el Dr. Negrete falleció en la ciudad de Xalapa recientemente; en cuanto forjador honorable y entrañable, perdurará en lo creado, lo fundado y lo enseñado en tan singular viaje.✷

Dr. José Negrete Martínez (1929-2018).

 ISSN 2007-0691

LLAMADO DE ARTÍCULOS

CIRC 2018

12º Congreso Internacional de Cómputo en Optimización y Software 2018

Del 3 al 5 de octubre del 2018, Cuernavaca Morelos, México

CICOS 2018

http://campusv.uaem.mx/cicos/

El Congreso Internacional de Cómputo es un portal para investigadores de instituciones de educación superior, empresas públicas/privadas y también estudiantes de postgrado. Este congreso les permite dar a conocer trabajos de investigación inéditos que se encuentran dentro de alguna línea de aplicación y generación de conocimiento del área de computación.

Envío de artículos:

- Artículos de investigación en el área de computación con aplicación a las áreas como eléctrica, materiales, mecánica, química y otras.
- Artículos de desarrollo en el área de cómputo dentro de los sectores público y privado.

Tópicos de Interés:

- Automatización de procesos.
- Optimización teórica y aplicada.
- Ingeniería de software.
- Arquitectura de Sistemas de Cómputo Lógica, computación y algorítmica.
- Sistemas distribuidos y paralelos.
- Bases de datos.
- Inteligencia artificial.
- Heurísticas, Algoritmos de aproximación y sus aplicaciones.
- Tecnologías de la Información.
- Tecnologías educativas.
- Sistemas de control.
- Diseño electrónico.

 ISSN 2007-0691

(http://latincom2018.ieee-comsoc-latincom.org/)

IEEE Latin-American Conference on Communications
Del 14 al 16 de Noviembre de 2018

Guadalajara, Mexico

Este año dará inicio la décima edición de la Conferencia Latinoamericana de Comunicaciones IEEE (LATINCOM) , y la Ciudad de Guadalajara ha sido elegida para albergar la conferencia en esta ocasión simbólica. Conocida como la Perla de Occidente, el Silicon Valley mexicano y la tierra del tequila y el mariachi, Guadalajara es rica en historia y tradiciones. Es la segunda ciudad más grande del país y uno de los destinos culturales y económicos más importantes de México y Sudamérica.

Guadalajara ha ganado relevancia como centro científico y tecnológico, ya que varias compañías tecnológicas importantes han establecido sitios de operación en esta ciudad. Además, Guadalajara cuenta con una fuerza de trabajo de más de 78,000 profesionales en TI, y es el hogar de más de 20 universidades e institutos tecnológicos que ofrecen cursos de ingeniería y de TI. Todos estos factores, junto con las agradables condiciones climáticas de noviembre, proporcionan un ambiente excelente para celebrar el décimo aniversario de LATINCOM.

Está cordialmente invitado a participar de la conferencia y ayudarnos a hacer de LATINCOM 2018 un evento memorable.

¡Esperamos recibirte en Guadalajara!

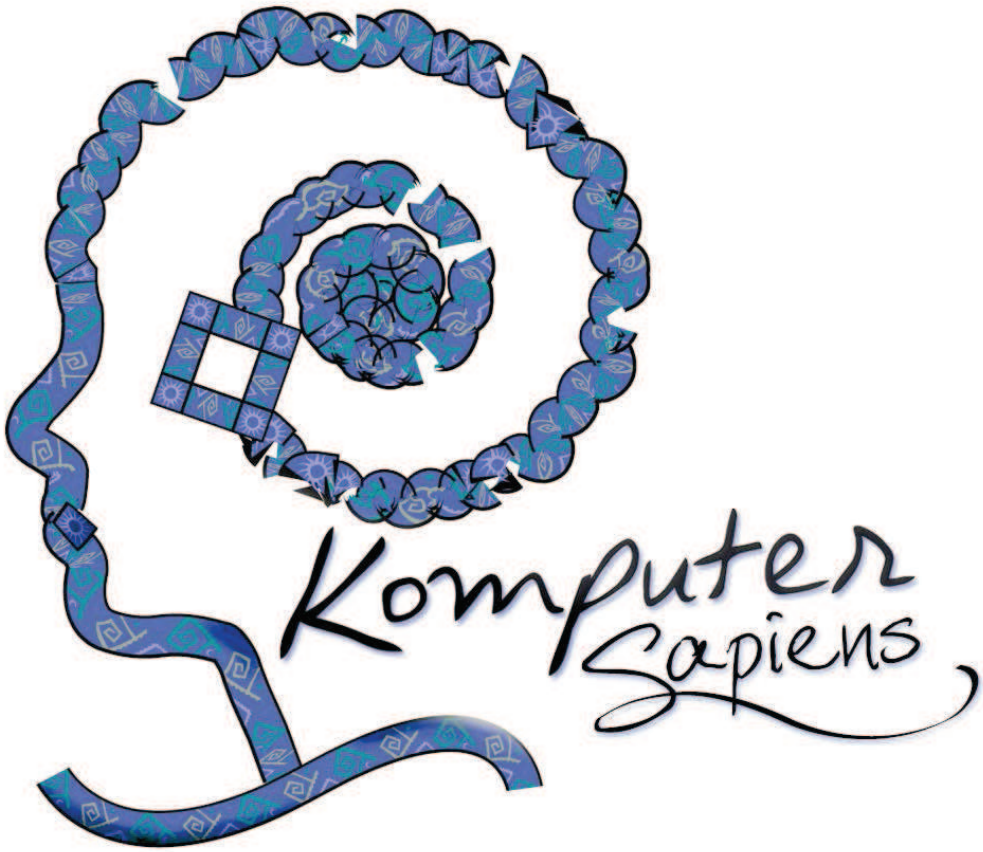
FECHAS IMPORTANTES

Presentación de trabajos completos:
31 de julio de 2018

Presentación de propuestas tutoriales:
31 de julio de 2018

Notificación de aceptación:
30 de septiembre de 2018

Presentación de trabajos listos para la cámara:
15 de octubre de 2018



¡Publique en Komputer Sapiens!

CONACYT

ÍNDICE DE REVISTAS MEXICANAS DE DIVULGACIÓN CIENTÍFICA Y TECONOLÓGICA (IRMDCT)

Komputer Sapiens solicita artículos de divulgación en todos los temas de Inteligencia Artificial, dirigidos a un amplio público conformado por estudiantes, académicos, empresarios, tomadores de decisiones y consultores. Komputer Sapiens es patrocinada por la **SMIA, la Sociedad Mexicana de Inteligencia Artificial**

SMIA

www.smia.org.mx

Instrucciones para autores e información general: http://www.komputersapiens.org
Síguenos en las redes sociales: www.facebook.com/Komputer.Sapiens, twitter.com/KomputerSapiens