PROPUESTA DE UN SISTEMA PARA LA DETECCIÓN AUTOMÁTICA DE COMPETENCIA Y MONITOREO DE PRODUCTOS EN MERCADOLIBRE

Ing. Gerardo Alberto López Vega [1], MTI. Juan Miguel Hernández Bravo [2], Dr. José Antonio Montero Valverde [3], Dra. Miriam Martínez Arroyo[4]

Resumen **--- El presente trabajo expone el desarrollo de un sistema inteligente capaz de detectar y hacer un seguimiento de manera automática de la competencia de una empresa al evaluar las características de vendedores y productos publicados en la plataforma de comercio electrónico MercadoLibre utilizando un modelo de clasificación y la técnica de minería de datos Naive Bayes. El sistema plantea la integración con el API de desarrollo de MercadoLibre, mediante la cual el usuario pueda acceder a los datos publicados, realizar búsquedas y dar seguimiento a los productos de sus competidores que se encuentren publicados en la plataforma, los datos registrados servirán como entrada para el modelo de clasificación que evaluará las características. Se plantea que el sistema analice el precio del producto, la reputación del vendedor, tipo de publicación del producto, forma de pago y tipo de envío.**

Palabras clave **--- clasificación, detección, seguimiento, competencia, mercadolibre.**

Introducción

De acuerdo con la Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares (ENDUTIH) 2018, en los últimos años ha ido al alza el interés de los mexicanos por el comercio electrónico, y actualmente en el país se calcula que cerca de 20 por ciento de las personas que tienen acceso a Internet lo usa para ordenar o comparar productos, esa cifra tuvo un incremento de 3.0 por ciento respecto a 2017, cuando fue cerca de 17 por ciento (ENDUTIH, 2018).

Según Linio en su informe Índice Mundial de Comercio Electrónico, las ventas de comercio electrónico en América Latina superan los 57,000 millones de dólares. Brasil y México se encuentran a la cabeza de los países con mayores ventas en el comercio electrónico en América Latina. México ha conseguido un alcance del 65%, detrás de Brasil que se consolido con el 71% (Linio, 2018).

De acuerdo con la consultora ComScore, el comercio electrónico en América Latina se encuentra dominado por MercadoLibre, con 56.3 millones de visitantes al mes, seguido por Amazon con 22.4 millones de visitantes (ComScore & EMarketer, 2018). En México MercadoLibre alcanza los 20 millones de visitas mensuales (Mercadolibre, 2017).

Según datos de MercadoLibre el aumento de vendedores en la plataforma ha tenido un comportamiento exponencial en los últimos años, el cual ha aumentado un 7% en 2018 llegando a alcanzar 10.8 millones de vendedores activos (MercadoLibre, 2018).

El mundo moderno es sumamente competitivo y exigente, la industria actual requiere de soluciones integrales de automatización y control cada vez más avanzadas que les permitan a las empresas optimizar su productividad y los recursos de modo que puedan competir en un entorno globalizado. Las PyMES cuentan con recursos limitados tanto humanos como financieros por lo que, para poder llegar a competir con empresas más grandes y posicionadas en el mercado, es necesario que cuenten con las herramientas que les permitan optimizar sus recursos y ser más eficientes en sus procesos. Las pequeñas y medianas empresas cumplen un importante papel en la economía de todos los países. Actualmente en México existen 4 millones de PyMES activas. Las cuales constituyen el 80% de las empresas, el 79% del empleo, y contribuyen con un 52% del PIB (Forbes, 2018).

Uno de los principales retos para las PyMES en México es la innovación tecnológica, de acuerdo a Forbes solo el 6% de las PyMES aprovecha las TIC's para mejorar su productividad (Forbes, 2018). Ante el creciente nivel de competencia, la búsqueda de mantenerse en el mercado y crecer es una tarea de vital importancia para las PyMES, por lo anterior es de gran importancia contar con las herramientas tecnológicas necesarias que le brinden información veraz y oportuna del estatus actual de la organización con la cual puedan tomar decisiones y desarrollar estrategias que les permitan crecer y tener un mejor aprovechamiento de sus recursos.

[1] Ing. Gerardo Alberto López Vega estudiante de Maestría en Sistemas Computacionales en el Instituto Tecnológico de Acapulco, Acapulco Guerrero. mcgalv@gmail.com
[2] MTI. Juan Miguel Hernández Bravo profesor del área de Maestría en Sistemas Computacionales del Instituto Tecnológico de Acapulco, Acapulco, Guerrero. jmhernan@yahoo.com
[3] Dr. José Antonio Montero Valverde profesor del área de Maestría en Sistemas Computacionales del Instituto Tecnológico de Acapulco, Acapulco, Guerrero. jamontero1@infinitummail.com
[4] Dra. Miriam Martínez Arroyo profesora del área de Maestría en Sistemas Computacionales del Instituto Tecnológico de Acapulco, Acapulco, Guerrero. miriamma_ds@hotmail.com

Técnicas de Minería de Datos

Según Russel y Norvig un sistema aprende cuando su desempeño mejora con la experiencia, es decir, cuando la habilidad no estaba presente entre sus rasgos de nacimiento (Russell & Norvig, 2009).

De acuerdo con Microsoft una técnica de minería de datos es un conjunto de heurísticas y cálculos que permiten crear un modelo a partir de datos. Para crear un modelo, el algoritmo analiza primero los datos proporcionados, en busca de tipos específicos de patrones o tendencias (Microsoft, 2018). El algoritmo usa los resultados de este análisis en un gran número de iteraciones para determinar los parámetros óptimos para crear el modelo de minería de datos. Las técnicas se separan en aquellas que siguen un enfoque de aprendizaje supervisado y en las que no. Algunas de las técnicas más representativas se observan en la Figura 1.

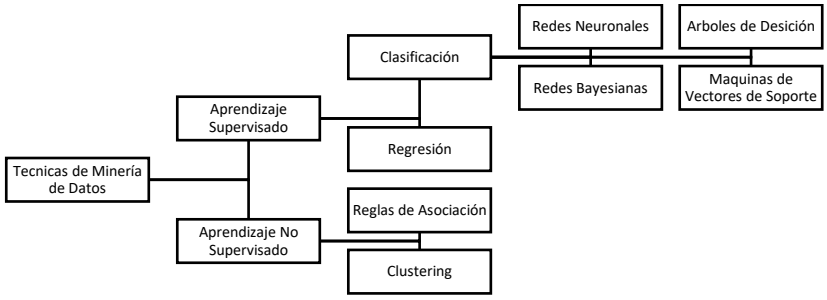

Figura 1. Técnicas de minería de datos más representativas

Modelo de Clasificación

En aprendizaje automático la clasificación es el problema de identificar a algo dentro de un conjunto de categorías (subpoblaciones) que pertenece a una nueva observación, sobre la base de un conjunto de datos que contiene observaciones (o instancias) cuya categoría de miembros es conocida, en la Figura 2 puede observarse un ejemplo de clasificación. La clasificación está considerada como un caso de aprendizaje supervisado, es decir, un aprendizaje en el que se dispone de un conjunto de observaciones correctamente identificadas. El procedimiento no supervisado se conoce como clustering, e implica agrupar los datos y categorías basadas en alguna medida de similitud o distancia inherente (Alpaydin, 2014). Un algoritmo que implementa la clasificación, se conoce como un clasificador, estos algoritmos predicen una o más variables discretas, basándose en los demás atributos del conjunto de datos.

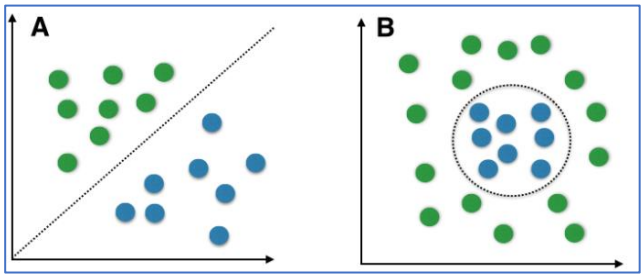

Figura 2. Ejemplo de identificación y clasificación de datos

Clasificador Naive Bayes

En teoría de la probabilidad y minería de datos, un clasificador Bayesiano es un clasificador probabilístico fundamentado en el teorema de Bayes (Han, 2011), se utiliza en el modelado de predicción y de exploración. Si una nueva tupla[5] va a ser clasificada, el Teorema de Bayes es utilizado para calcular la probabilidad de que pertenezca a la clase utilizando la ecuación (1) (Mozina, Demsar, Kattan, & Zupan, 2004). En la ecuación (1) la P denota probabilidad y la notación $P(T|C_i)$ representa la probabilidad condicional de T dado que se sabe que la clase es C_i. C_i es una clase que pertenece al conjunto de clases $\{C_1, C_2, C_3, \ldots\}$ para el conjunto de datos. La ecuación (1) es calculada para cada una de las clases y la clase con $P(C_i|T)$ más alta es considerada como instancia de la clase.

[5] Tupla En matemáticas, una tupla es una lista ordenada de elementos.

$$P(C_i|T) = \frac{P(T|C_i)P(C_i)}{P(T)} \quad (1)$$

Al calcular $P(C_i|T)$ para cada C_i el denominador $P(T)$ es constante en todas las clases por lo tanto puede ser eliminado de la ecuación. Así, la ecuación (2) se puede usar para encontrar la clase que tiene la mayor probabilidad. El símbolo ~ denota que el lado izquierdo es proporcional al lado derecho.

$$P(C_i|T) \sim P(T|C_i)P(C_i) \quad (2)$$

El clasificador Naive Bayes se basa en el supuesto de independencia condicional de clase, es decir, que los valores de los atributos de T son independientes entre sí. Como consecuencia, si T es la tupla n <*t1, t2, ... tn*>, entonces $P(T|C_i)$ de la ecuación (2) puede ser calculada utilizando la ecuación (3). La justificación de la ecuación (3) se basa en la teoría de probabilidad, donde la probabilidad conjunta de eventos independientes se puede calcular multiplicando las probabilidades de estos eventos. Por lo tanto, para calcular $P(C_i|T)$ basado en la ecuación (2) necesitamos calcular $P(C_i)$ y calcular $P(T|C_i)$ basados en Ecuación (3) y multiplica los dos resultados. Esto se realiza para cada clase C_i y la clase con el valor más alto es elegido como la clase para la nueva tupla T.

$$P(T|C_i) = \prod_{k=1}^{n} P(t_k|C_i) = P(t_1|C_i)\ x\ P(t_2|C_i)\ x\ P(t_3|C_i)\ x\ \ldots\ x\ P(t_n|C_i) \quad (3)$$

Descripción del Método

La metodología utilizada para este estudio está basada en el proceso KDD[6] para la obtención de conocimiento el cual ha sido adaptado para ajustarse a los objetivos y características del proyecto. El desarrollo del proyecto se encuentra divido principalmente en 6 etapas como puede observarse en la Figura 3, las cuales serán descritas a continuación.

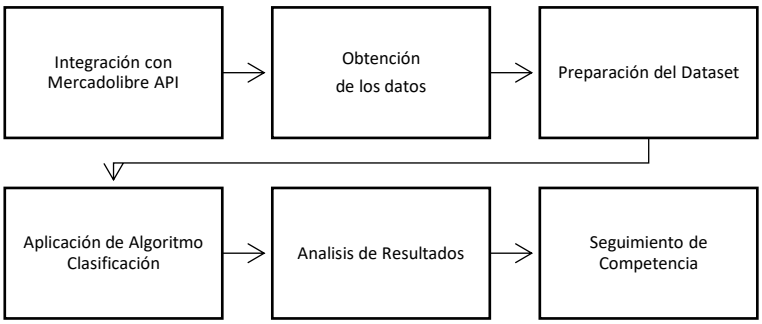

Figura 3 Metodología del proyecto

Integración con Mercadolibre API

En esta etapa se plantea el desarrollo de un sistema que permita establecer una conexión con la plataforma MercadoLibre, como puede verse en la Figura 4 la conexión se realiza a través del API[7] de desarrollo de la plataforma, esto con la finalidad de obtener los datos de vendedores y productos, así como también permitir al usuario buscar y darles seguimiento a los productos de su competencia. Se propone desarrollar el proyecto bajo un ambiente Web utilizando el lenguaje Python y Django debido a las características del proyecto, así como por las especificaciones del API.

[6] KDD (Knowledge Discovery in Databases) En la minería de datos es un campo de la estadística y las ciencias de la computación referido al proceso que intenta descubrir patrones en grandes volúmenes de conjuntos de datos.
[7] API (Application Programming Interface), es un conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Figura 4. Integración al API de MercadoLibre

Obtención de los datos

Durante esta etapa se pretende que el usuario utilice el sistema y lo alimente con los datos obtenidos de MercadoLibre, los cuales se encuentran codificados en formato JSON[8] como puede verse en la Figura 5, así como también, aquellos que se registren al buscar y seleccionar de forma manual la competencia para los productos del usuario, los cuales servirán como entrada para el modelo de clasificación.

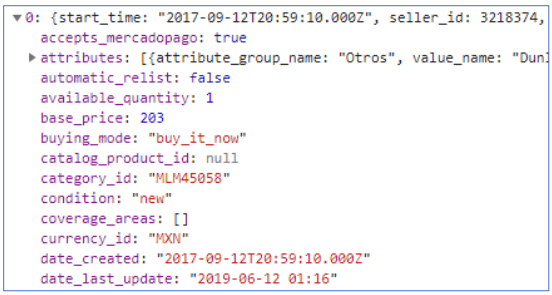

Figura 5. Datos obtenidos de MercadoLibre en formato JSON

Preparación del Dataset

En esta etapa, se plantea el procesamiento y limpieza de los datos registrados poder aplicar el algoritmo de clasificación, los datos deben de estar en un formato que pueda ser procesado por el algoritmo, por lo que primero hay transformarlos para poder utilizarlos. El Cuadro 1 muestra un conjunto de datos de entrenamiento que pueden usarse. El atributo de clase seleccionado es Competencia (C), que indica si el producto es competencia del producto del usuario. Los datos utilizados para determinar si es competencia son Calificación (R), Pago (P), Precio (B), Envío (E) y Tipo (T).

R tomará el valor de "roja" para vendedores con reputación negativa, "amarilla" para vendedores con reputación regular y "verde" para vendedores con excelente reputación. B será "menor" cuando el producto del otro vendedor sea más económico que el producto del usuario y "mayor" en caso contrario, T será "especial" para publicaciones con mayor exposición y "normal" para publicaciones normales. P será "si" en caso de que el vendedor acepte pagos diferidos y "no" en caso contrario, por último, E tomará el valor de "si" en caso de ofrecer envío gratis del producto y "no" en caso contrario. Por ejemplo, la primera instancia del Cuadro 1 tiene los siguientes valores para los diferentes atributos R = verde, P = si, B = menor, y E = si y T = especial, basados en estos datos, la decisión mostrada en la columna C es si, en otras palabras, el producto es clasificado como competencia por las características que lo hacen más competitivo que el producto del usuario. Esto es basado en la experiencia pasada hecha por el usuario.

Calificación (R)	Pago (P)	Precio (B)	Envío (E)	Tipo (T)	Competencia (C)
verde	si	menor	si	especial	si
amarilla	no	mayor	si	especial	no
rojo	no	mayor	no	especial	no
rojo	no	mayor	si	especial	no
amarillo	si	menor	si	normal	si
verde	no	menor	si	normal	si
verde	si	mayor	si	normal	si
amarillo	si	menor	no	normal	si
amarillo	no	mayor	no	especial	no

[8] JSON (Javascript Object Notation) es un formato de texto sencillo para el intercambio de datos.

verde	no	mayor	no	especial	no
verde	no	menor	si	especial	no
rojo	si	mayor	no	normal	si
amarillo	si	menor	si	normal	si
verde	no	menor	si	normal	si
verde	no	menor	no	normal	no
rojo	si	menor	si	normal	si
rojo	si	menor	no	normal	si

Cuadro 1. Conjunto de datos de entrenamiento para el modelo de clasificación

Aplicación del Algoritmo de Clasificación

En esta etapa se propone aplicar la técnica Naive Bayes para analizar nuevas tuplas agregadas al conjunto de datos, por ejemplo, considerando que los valores de los atributos de una nueva tupla son ("verde", "no", "mayor", "no", "especial") considerando el orden de las columnas mostradas en el Cuadro 1, el algoritmo debe de calcular la probabilidad de que este pertenezca a la clase Competencia, para ello se considera lo siguiente, existen 2 clases identificadas para la etiqueta de clase Competencia, C = si y C = no. Sustituyendo en la ecuación (2) para cada uno de ellas se obtienen las ecuaciones (4) y (5).

$$P(C = si \mid T) \sim P(T \mid C = si) * P(C = si) \quad (4)$$
$$P(C = no \mid T) \sim P(T \mid C = no) * P(C = no) \quad (5)$$

En el primer paso se calcula $P(C = si)$ y $P(C = no)$ utilizados en las ecuaciones (4) y (5).

$P(C = si) = 10/17 = 0.59$
$P(C = no) = 7/17 = 0.41$

En el segundo paso calculamos la probabilidad de $P(T \mid C = si)$ y $P(T \mid C = no)$ al sustituir los valores individuales de probabilidad por cada valor del atributo en la ecuación (3).

$P(R = verde|C = si) = 4/10 = 0.4$
$P(P = no|C = si) = 2/10 = 0.2$
$P(B = mayor|C = si) = 2/10 = 0.2$
$P(E = no|C = si) = 3/10 = 0.3$
$P(T = especial|C = si) = 1/10 = 0.1$
$P(T \mid C = si) = 0.4 * 0.2 * 0.2 * 0.3 * 0.1 = 0.00048$

$P(R = verde|C = no) = 3/7 = 0.429$
$P(P = no|C = no) = 7/7 = 1$
$P(B = mayor|C = no) = 5/7 = 0.714$
$P(E = no|C = no) = 4/7 = 0.571$
$P(T = especial|C = no) = 6/7 = 0.857$
$P(T \mid C = no) = 0.429 * 1 * 0.714 * 0.571 * 0.857 = 0.14988$

Como último paso se realiza la multiplicación representada por las ecuaciones (4) y (5) para obtener el resultado final. Dado que $P(C = no \mid T) > P(C = si \mid T)$ la clase de la nueva fila es identificada como C = no.

$P(C = si \mid T) \sim P(T \mid C = si) * P(C = si) = 0.0005 * 0.59 = 0.000282$
$P(C = no \mid T) \sim P(T \mid C = no) * P(C = no) = 0.1499 * 0.41 = 0.0616$

Análisis de Resultados

En esta etapa se analizan los resultados obtenidos por el algoritmo aplicado en la etapa anterior, para ello se propone el desarrollo de un módulo dentro del sistema que permita visualizar mediante un reporte los datos de los productos que hayan sido detectados por el algoritmo. Como puede verse en la Figura 6, el reporte debe mostrar el precio de los productos e indicar si son mayores o menores que el precio del producto del usuario. El usuario podrá determinar si el resultado de la detección ha sido correcto, y en caso contrario indicar en el sistema si el producto no se considera competencia, de modo que el modelo de clasificación vaya mejorando conforme se utilice.

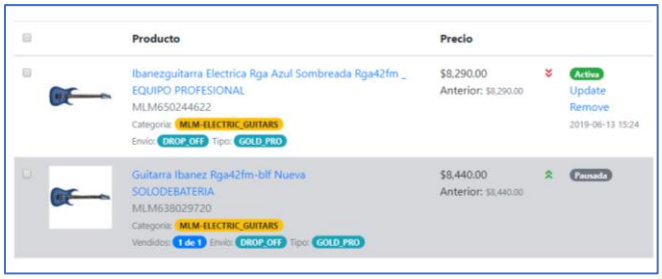

Figura 6. Reporte de Resultados

Seguimiento de Competencia

En esta etapa se plantea que el sistema realice el seguimiento automático de los productos que se hayan marcado como competencia por el algoritmo de clasificación. En esta parte, el sistema debe de hacer consultas cada cierto periodo de tiempo al API de MercadoLibre con el fin de obtener los datos actualizados de los productos y actualizar la información registrada en la base de datos del sistema, en caso de detectar cambios en las publicaciones de otros vendedores, el sistema generará una alerta para el usuario indicando por ejemplo un precio sugerido para su producto en caso que los productos de la competencia sean mejores, lo anteriormente mencionado se puede observar en la Figura 7 donde se puede ver un producto con diferentes precios, entre ellos el actual y el sugerido.

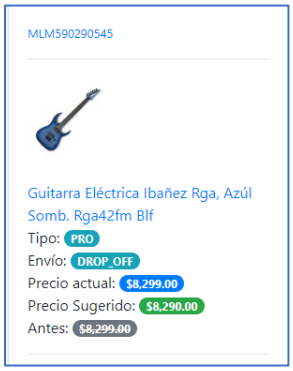

Figura 7. Seguimiento de Competencia y Precio Sugerido

Comentarios Finales

Conclusiones

Se ha mostrado en este trabajo una propuesta de desarrollo de un sistema que se integra con MercadoLibre una de las plataformas de comercio electrónico más importantes en México y América Latina, mediante su API de desarrollo mediante el cual se pretende automatizar el proceso de detección y seguimiento de competencia al integrar un algoritmo de clasificación mediante la técnica de minería de datos Naive Bayes. En el trabajo se analizaron los datos de entrenamiento que se pretenden utilizar para el modelo y se describió el fundamento probabilístico de la técnica propuesta. Así mismo se describen las etapas del método, las cuales se basaron en el proceso KDD por sus similitudes con el proyecto y las fases para su desarrollo.

Recomendaciones

En este trabajo se ha propuesto la técnica de clasificación Naive Bayes ya que resulta útil para generar rápidamente modelos de minería de datos que detecten las relaciones entre las columnas de entrada y las columnas de predicción. Este algoritmo resulta útil para realizar una exploración inicial de los datos y, más adelante, aplicar los resultados para crear modelos de minería de datos adicionales con otros algoritmos más complejos y precisos, por lo que se recomienda continuar el presente trabajo explorando otras técnicas de clasificación como árboles de decisión o redes neuronales y así, comparar los resultados obtenidos.

Referencias

Alpaydin, E. (2014). *Introduction to Machine Learning.* MIT Press.
ComScore, & EMarketer. (2018, Mayo). *Statista*. Retrieved from https://www.statista.com/statistics/321543/latin-america-online-retailer-visitors/
ENDUTIH. (2018). Retrieved from https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2019/OtrTemEcon/ENDUTIH_2018.pdf

Forbes. (2018, Enero 31). *Pymes Mexicanas, Un panorama para 2018*. Retrieved from https://www.forbes.com.mx/pymes-mexicanas-un-panorama-para-2018/

Han, J. (2011). *Data Mining: Concepts and Techniques, 3rd Ed.* Morgan Kaufmann.

Linio. (2018). *Índice Mundial de Comercio Electrónico*. Retrieved from https://www.linio.com.mx/sp/indice-ecommerce

Mercadolibre. (2017, Diciembre). Retrieved from http://www.mercadolibrepublicidad.com.mx/descargas/2017/ML_Mediakit2018_MX.pdf

MercadoLibre. (2018). Retrieved from https://www.marketplacepulse.com/stats/mercadolibre/mercado-libre-number-of-sellers-81

Microsoft. (2018, Abril 30). Retrieved from https://docs.microsoft.com/es-es/sql/analysis-services/data-mining/data-mining-algorithms-analysis-services-data-mining?view=sql-server-2017#choosing-the-right-algorithm

Mozina, M., Demsar, J., Kattan, M., & Zupan, B. (2004). Nomograms for Visualization of Naive Bayesian Classifier. *Springer, Berlin, Heidelberg*.

Russell, S., & Norvig, P. (2009). *Inteligencia Artificial: Un Enfoque Moderno.* Prentice Hall.