Clasificación de Transacciones usando kNN

Alejandro Murillo González
Universidad EAFIT
Medellin, Colombia
amurillog@eafit.edu.co

Juan Pablo Vidal Correa
Universidad EAFIT
Medellin, Colombia
jpvidalc@eafit.edu.co

Henry Giovanny Velasco Vera
Universidad EAFIT
Medellin, Colombia
hgvelascov@eafit.edu.co

ABSTRACT

We propose a K- Nearest Neighbors (k-NN) model to classify data sets composed of PSE transactions that do not contain an assigned category, which is necessary to identify the economic sector of the transaction. We found that the model is a well-suited solution for the data classification task, and its implementation would allow Bancolombia clients, that make use of the **Personas app**, to easily identify the nature of their expenses through the PSE system.

1 INTRODUCCIÓN

El desafío propuesto considera la clasificación de los datos no categorizados generados por pagos hechos vía el sistema de Pago Seguro en Línea (PSE), para afrontar esta problemática se plantea una solución que involucra el machine learning supervisado para generar un algoritmo capaz de clasificar este tipo de datos a partir de otros que ya tienen una categoría asignada.

Las bases de datos que se generan por pagos realizados en el sistema PSE poseen una categorización no estandarizada y rígida, enfocada en su mayoría en los servicios, dando como resultado referencias con poca precisión acerca de los servicios y/o productos a los cuales los clientes de Bancolombia están accediendo al momento de usar este sistema. Además, se presentan casos donde los datos no tienen una categoría asignada al tipo de producto que se comercializa y sólo poseen 3 referencias generadas en texto libre, las cuales varían entre cada transacción. Esto es un problema ya genera ambigüedad en la clasificación, al no tener un estándar definido. Este tipo de datos se generan cuando los intermediarios de la transacción no son clientes del banco y son representados por un valor Null en la base de datos. Por lo tanto, teniendo en cuenta la situación anterior, se planea crear un algoritmo capaz de categorizar los datos de los receptores no afiliadas al banco, recreando así, una clasificación similar a la utilizada por el Merchant Category Code (MCC), del cual los registros del sistema PSE carecen en general.

La solución implementa un algoritmo k-NN, el cual recibe los datos sin categoria, solo con los textos de referencia, y los clasifica en las clases que contienen los datos con una categoría asignada.

2 MARCO TEÓRICO

2.1 K- Nearest Neighbors

K- Nearest Neighbors es un método de clasificación supervisada que estima la probabilidad de que cierto elemento pertenezca a una determinada clase. El algoritmo almacena todos los casos posibles de un conjunto muestra, y clasifica nuevos casos basándose en las similitudes de esta muestra. Para el reconocimiento y clasificación de cada caso, el algoritmo k-NN se basa en un entrenamiento que consiste en recopilar los vectores y etiquetas características de las clases de los casos de ejemplo. En la clasificación, se calcula la distancia entre los vectores almacenados y el nuevo ejemplo (representado por un vector y con una clase desconocida), y se seleccionan los $\boldsymbol{k}$ ejemplos más cercanos. El ejemplo es clasificado con la clase que más se repite en los vectores elegidos [3].

La óptima selección de la $\boldsymbol{k}$ depende completamente de los datos, valores de $\boldsymbol{k}$ grandes reducen considerablemente el ruido, pero crean límites en clases que comparten similitudes. En el caso de este proyecto se seleccionó una $\boldsymbol{k}$ = 3, junto con un proceso de preprocesamiento en los datos para reducir el ruido, el cual puede degradar considerablemente la precisión del algoritmo.

Un ejemplo de cómo opera un algoritmo k-NN se muestra en la figura 1.

3 METODOLOGÍA

3.1 Manejo de datos

Al momento de analizar las bases de datos entregadas, se observa un total de 11829025 de datos, de los cuales 3297106 contienen una categoría asignada por el sistema y el resto de los datos (8531919) solo poseen un texto de referencia que indica la naturaleza de la transacción.

Los datos que están categorizados se usan para conformar los conjuntos de entrenamiento, validación y prueba del modelo. El conjunto de datos se divide como se muestra en la Tabla 1.

3.2 Asignación de categorías

Tomando como referencia los 3297106 datos, los cuales contienen información categórica (sector y subsector económico del receptor), se identificarón 77 subcategorías que se agruparon en 20 nuevas categorías. Éstas permiten entrenar un algoritmo de aprendizaje supervisado, de las cuales la solución tomará referencia para clasificar los 8531919 datos no categorizados. Las nuevas categorías se muestran en la Tabla 2.

El uso de estas etiquetas, le permiten al cliente un menú donde puede identificar fácilmente la naturaleza de sus gastos a través del sistema PSE.

3.3 Preprocesamiento

Se procesan los datos utilizando una función Tokenizer [1] que categoriza las partes de un String en un número fijo de palabras. En este caso , se fijó el máximo número de palabras en 5000. Esto significa que función Tokenizer identifica las 5000 palabras más frecuentes del set de entrenamiento y a cada una le asigna una etiqueta numérica, convirtiendo cada frase en un vector.

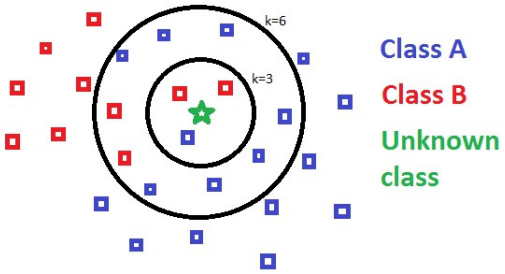


Figure 1: Ejemplo de cómo funciona un algortimo k-NN [4].

Table 1: Describe como estan divididos los conjuntos de entrenamiento, validación y prueba.

Tipo de conjunto	Número de datos
Conjunto de entrenamiento	1350000
Conjunto de validación	75000
Conjunto de prueba	75000

Table 2: Muestra las nuevas categorías asignadas.

Nuevas categorías				
1. Servicios bancarios	2. Gobierno	3. Valor agregado	4. Administración central	5. Educación
6. Electricidad	7. Tecnología y comunicaciones	8. Construcción	9. Servicios sociales	10. Servicios a empresas
11. Transporte	12. Salud	13. Comercio	14. Servicios financieros	15. Servicios a personas
16. Seguros	17. Comida	18. Turismo	19. Otros	20. DEFAULT LABEL

3.4 Investigación de algoritmos

Una vez las categorías están definidas, se procede a investigar y probar algoritmos que nos permitan categorizar datos. Entre los algoritmos analizados se encuentran k-Nearest Neighbor (k-NN), Decision Tree, Random Forest, Quadratic Discriminant Analysis, Support Vector Machine (SVM). Los tres primeros algoritmos entregaron resultados bastante prometedores en el ejercicio de validación . Sin embargo, se ha decidido utilizar el algoritmo k-NN por tener, entre los tres, un mejor desempeño respecto a la tasa de acierto y de precisión.

3.5 Creación del modelo

Empleando una arquitectura k-NN se construye un modelo que es capaz de clasificar datos que no contienen una categoría, a partir de su texto de referencia. El modelo es entrenado, validado y probado con una parte de los 3297106 datos preprocesados que contienen un categorÍa, como lo muestra la Tabla 1.

Este modelo fue desarrollado usando SkLearn y Keras compilado sobre Tensorflow v1.6.0. Las especificaciones de la maquina usada son:

- Procesador: Intel Core i7-6700HQ CPU @2.60GHz x 8
- GPU: GeForce GTX 1060/PCIe/SSE2
- Disco de memoria: 380 GB
- Memoria ram: 12 GB
- Sistema operativo: Ubuntu 16.04 LTS
- OS type: 64-bit

4 RESULTADOS

4.1 Validación de resultados

Para medir la precisión de los resultados arrojados por el modelo basado en un algoritmo k-NN, se utilizaron las métricas F1 score y R^2 [2]. Estos resultados se pueden ver en la Tabla 3 para los casos de entrenamiento, validación y prueba. La fórmula de estas medidas de precisión son:

Para F1 score:

$$F1 = 2 * (\frac{precision * recall}{precision + recall})$$

Donde *precision* es la fracción de datos a los que el algoritmo asignó correctamente una categoría entre los datos a los cuales el algoritmo determino, sea correcta o incorrectamente, una categoría. Por otro lado *recall* es la fracción de datos a los que el algoritmo asigna correctamente una categoría sobre el total de datos que eran de esa categoría.

Para R^2:

$$R^2 = 1 - (\frac{SS_{res}}{SS_{tot}})$$

Donde SS_{res} es la suma residual de cuadrados de los resultados y SS_{tot} es la varianza.

Teniendo en cuenta estas métricas se puede observar, que el algoritmo es bastante preciso en la categorización, ya que tanto para F1 - score como para R^2, entre más cerca de 1 este su valor, más precisos serán los resultados. A su vez, la medida de accuracy, la cual se refiere a la cercanía de una medida con respecto a su valor estándar, recalca la calidad y la precisión con la que el modelo clasifica correctamente a un dato no categorizado.

4.2 Clasificación de los datos sin categoría

Al utilizar el modelo con una muestra de 893405 datos perteneciente a los datos no categorizados, se obtuvierón las Figuras 2 y 3, las cuales muestran la frecuencia con la que se asigna la categoría a cada transacción y la distribución del dinero en cada categoría respectivamente. Esto permite observar datos interesantes, como el hecho de que aproximadamente el 80% del dinero que se maneja en transacciones PSE proviene de los servicios bancarios y servicios o productos provenientes de las tecnologías y comunicaciones. Estos resultados dejan en evidencia la procedencia de los servicios y productos a los cuales los clientes de Bancolombia están accediendo al momento de utilizar el sistema PSE.

4.3 Comparación con otros modelos

Se realizaron otros modelos a la par del algoritmo de k-NN para evaluar resultados y observar cual tenía un mejor rendimiento, entre estos se encuentra una SVM, la cual operando con 6619734 de registros no categorizados genero las gráficas 4 y 5, las cuales muestran la frecuencia con la que fue asignada cada categoría. Estos resultados difieren de los obtenidos con la k-NN, y en el caso de la gráfica 4, muestran que se está asignando un gran número de categorías ha DEFAULT LABEL, lo cual indica que el algoritmo no es muy preciso al momento de clasificar, mientras que la gráfica 5, la cual no tiene en cuenta esta categoría, sigue siendo diferente a los resultados obtenidos por el modelo k-NN, pero por las pruebas de precisión hechas con anterioridad, se puede demostrar que los resultados deberían asemejarse a los de la k-NN.

5 TRABAJO FUTURO

Una opción a contemplar es usar Word embeddings, los cuales permiten vincular palabras o frases a vectores de números reales, esto con el objetivo de lograr más generalización en las palabras con las que opera el algoritmo en su proceso de clasificar.

Por otro lado, los resultados que muestra el modelo permiten integrar satisfactoriamente al algoritmo a una aplicación de Personal Financial Managers (PFM), para que así las personas puedan tener a la mano las transacciones que han realizado vía PSE y la categoría de la misma, como se muestra en la Figura 2.

6 CONCLUSIONES

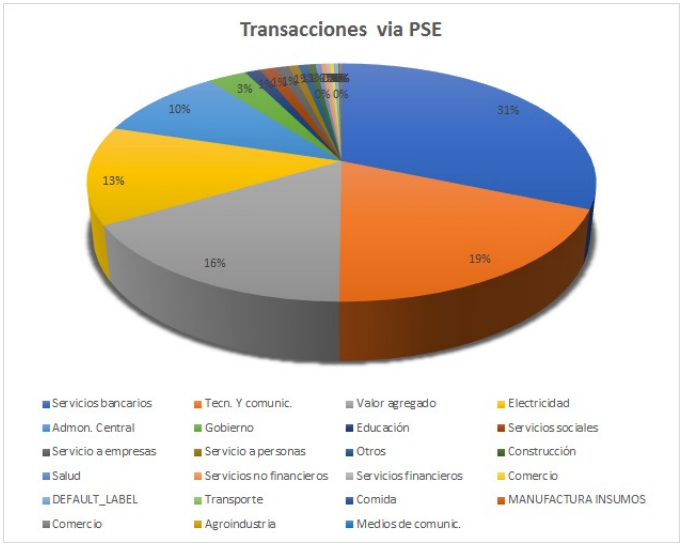
Como se aprecia en los resultados, el modelo generado es una solución viable para el desafío de clasificar conjuntos de datos pertenecientes a transacciones vía PSE. Esto permite replicar una clasificación similar a la utilizada por el MCC, pero aplicada a los servicios PSE, lo cual permitirá a los clientes tener un mayor control de sus transacciones, ya que se podría registrar la naturaleza de la mismas en este sistema. Por otro lado, este modelo también le brindaría al banco información sobre las transacciones que hacen sus usuarios por medio del servicio PSE.

REFERENCES

[1] Python Software Foundation. 2018. Tokenizer for Python source. (2018).
[2] Scikit learn develope. 2018. Sklearn.metrics. (2018).
[3] Yang Song, Jian Huang, Ding Zhou, Hongyuan Zha, and C Lee Giles. 2007. Iknn: Informative k-nearest neighbor pattern classification. In *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 248–264.
[4] Zeiselt. 2016. K-Nearest Neighbour(KNN) classification algorithm implementation in Python. (2016).

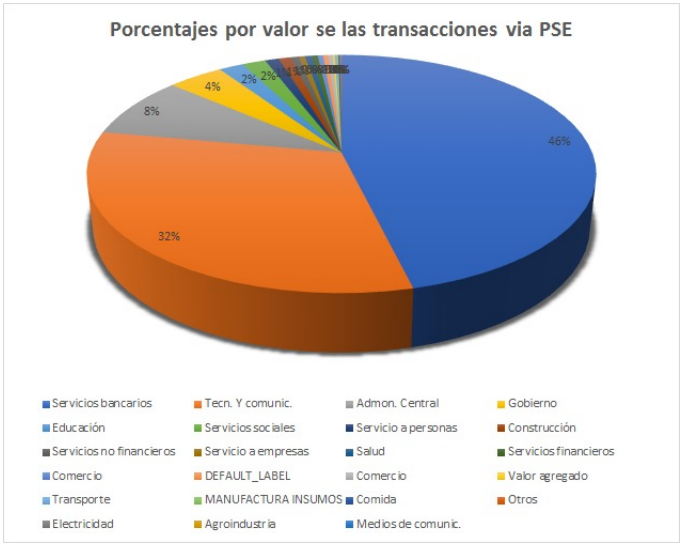
Table 3: Resume las métricas de los conjuntos de Entrenamiento, Validación y Prueba.

	Score	Accuracy	Recall	Precision	F1	R^2
Entrenamiento	0.991257	0.991257	0.991257	0.991208	0.991076	0.984354
Validación	0.990373	0.990373	0.990373	0.990838	0.990435	0.978653
Prueba	0.98808	0.98808	0.98808	0.989438	0.988381	0.974052



2.jpg

Figure 2: Frecuencia de la asginación de categorías.



1.jpg

Figure 3: Distribución del dinero en cada categoria.

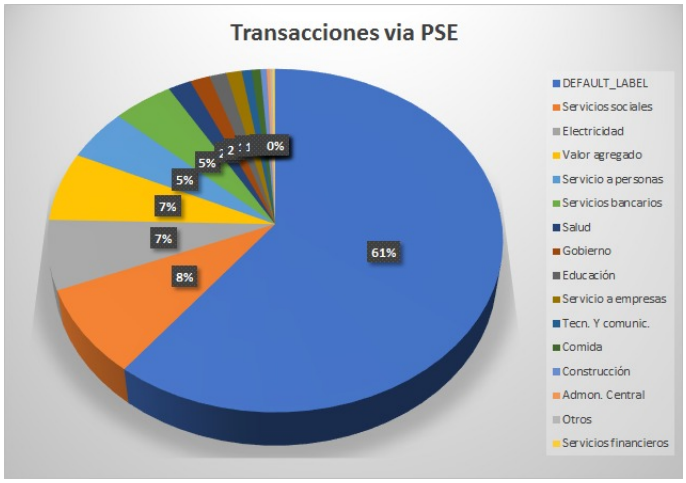
3.jpg 

Figure 4: Frecuencia de la asginación de categorías en modelo de SVM.

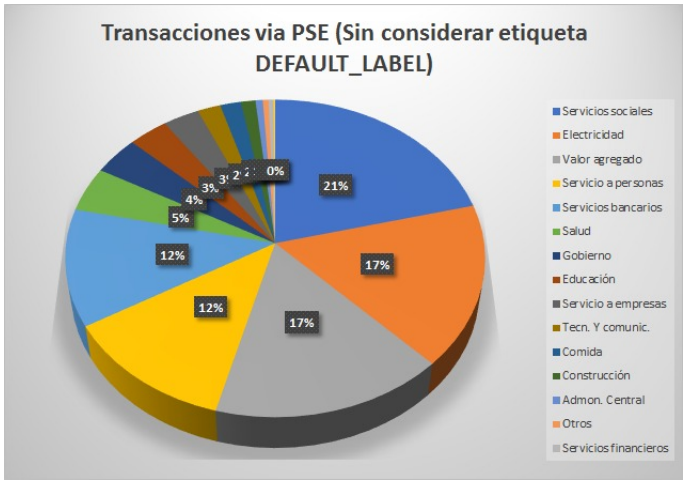
4.jpg 

Figure 5: Frecuencia de la asginación de categorías en modelo de SVM sin tener en cuenta DEFAULT LABEL.

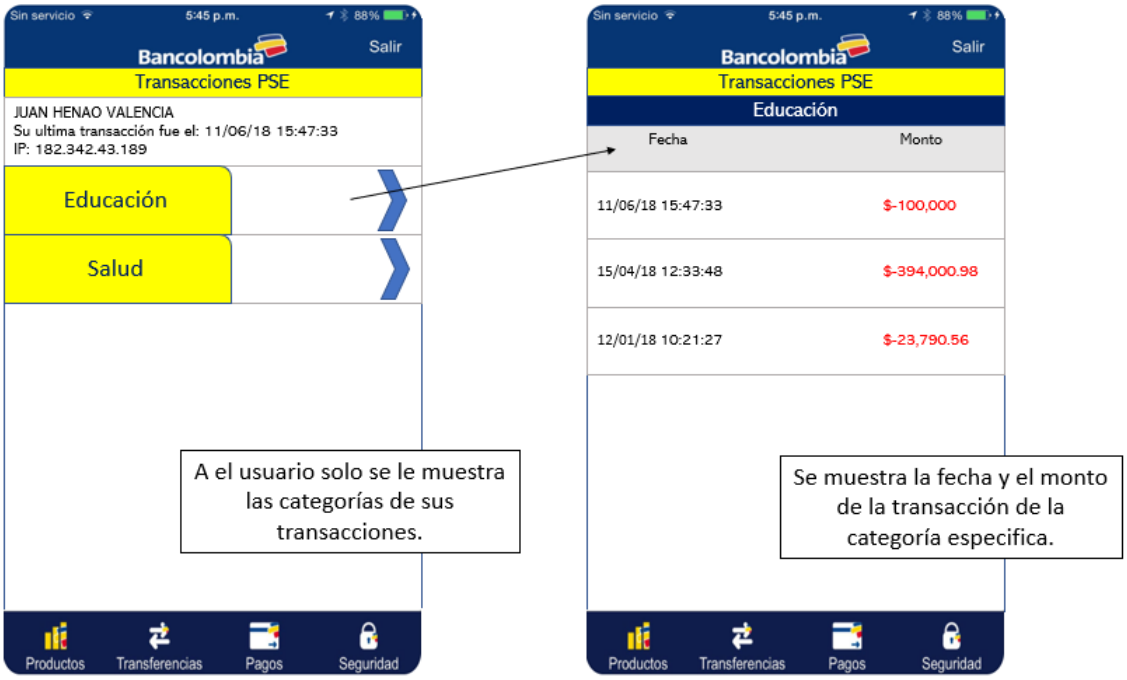


Figure 6: Posible implementación del modelo en una PFM.