TecnoLógicas
ISSN: 0123-7799
ISSN: 2256-5337
tecnologicas@itm.edu.co
Instituto Tecnológico Metropolitano
Colombia

Clasificación multiclase y visualización de quejas de organismos oficiales en twitter

Hernández-Pajares, Beatriz; Pérez-Marín, Diana; Frías-Martínez, Vanessa
Clasificación multiclase y visualización de quejas de organismos oficiales en twitter
TecnoLógicas, vol. 23, núm. 47, 2020
Instituto Tecnológico Metropolitano, Colombia
Disponible en: https://www.redalyc.org/articulo.oa?id=344262603021
DOI: https://doi.org/10.22430/22565337.1454

Artículos de investigación

Clasificación multiclase y visualización de quejas de organismos oficiales en twitter

Visualization and Multiclass Classification of Complaints to Official Organisms on Twitter

Beatriz Hernández-Pajares
Wavespace, España
beatriz.hernandezpajares@ey.es
http://orcid.org/0000-0001-9577-2350

Diana Pérez-Marín
Universidad Rey Juan Carlos, España
diana.perez@urjc.es
http://orcid.org/0000-0003-3390-0251

Vanessa Frías-Martínez
Universidad de Maryland, Estados Unidos
vfrias@umd.edu
http://orcid.org/0000-0001-5114-7633

DOI: https://doi.org/10.22430/22565337.1454
Redalyc: https://www.redalyc.org/articulo.oa?id=344262603021

Recepción: 02 Agosto 2019
Aprobación: 16 Octubre 2019

Resumen:

Las redes sociales acumulan gran cantidad de información. Las actuales técnicas de Procesamiento de Lenguaje Natural permiten su procesamiento automático y las técnicas de Minería de Datos permiten extraer datos útiles a partir de la información recopilada y procesada. Sin embargo, de la revisión del estado del arte, se observa que la mayoría de los métodos de clasificación de los datos identificados y extraídos de redes sociales son biclase. Esto no es suficiente para algunas áreas de clasificación, en las que hay más de dos clases a considerar. En este artículo, se aporta un estudio comparativo de los métodos svm y Random Forests, para la identificación automática de n-clases en *microblogging* de redes sociales. Los datos recopilados automáticamente para el estudio están conformados por 190 000 *tweets* de cuatro organismos oficiales: Metro, Protección Civil, Policía, y Gobierno de México. De los resultados obtenidos, se recomienda el uso de Random Forests, ya que se consigue una precisión media del 81.46 % y una cobertura media del 59.88 %, con nueve tipos de quejas identificadas automáticamente.

Palabras clave: Minería de texto, clasificación multiclase, redes sociales, Twitter.

Abstract:

Social networks generate massive amounts of information. Current Natural Language techniques allow the automatic processing of that information, and Data Mining enables the automatic extraction of useful info. However, a state-of-the-art review reveals that many classification methods only distinguish two classes. This paper presents a procedure to automatically classify tweets into several classes (more than two). The steps of the procedure are described in detail so that any researcher can follow them. The accuracy and coverage (instead of only coverage as usual in the literature) of two automatic classifiers (SVM and Random Forests) were analyzed in a comparative study. The procedure was applied to automatically identify more than two types of complaint from 190,000 tweets. According to the results, Random Forests should be used because they achieve an average accuracy of 81.46 % and an average coverage of 59.88 %.

Keywords: Text Mining, Multiclass Classification, Social Networks, Twitter.

1. INTRODUCCIÓN

Las redes sociales acumulan una gran cantidad de usuarios que conversan digitalmente, guardan textos y comentan textos de otros usuarios. En particular, en la actualidad, Facebook posee más de 2217 millones de

usuarios, que la convierten en la red social más usada, y Twitter cuenta con más de 326 millones de usuarios [1], que a diario generan unos 500 millones de *tweets* en todo el mundo [2].

Esta gran cantidad de información resulta muy difícil de procesar manualmente, por este motivo se utilizan técnicas de Procesamiento de Lenguaje Natural, que permiten automatizar su procesamiento. Este trabajo, se centra en el uso de métodos estadísticos [3], [4], en la eliminación de términos que se consideran superfluos, en la normalización de los textos y la aplicación de técnicas de lematización, para reducir las palabras a su raíz y parametrizar los documentos, mediante la asignación de un peso a cada uno de los términos relevantes, con la técnica tf-idf [5].

La minería de textos (*text mining*) consiste en un conjunto de técnicas que permiten extraer información relevante y desconocida, de manera automática, de grandes volúmenes de información textual, normalmente, en lenguaje natural y por lo general no estructurada [6]. En este trabajo, se revisan varios métodos de clasificación automática como svm, con varias funciones kernel y Random Forests. Se compara la precisión y cobertura obtenida para la identificación automática de clases que etiqueten el contenido de *tweets* recopilados automáticamente de organismos oficiales.

El objetivo es proporcionar un procedimiento para la clasificación automática multiclase de la información contenida en *tweets* de usuarios de organismos oficiales y su representación gráfica. Se enfatiza en la necesidad de que la clasificación sea multiclase y en la búsqueda de los métodos de clasificación automática, cuando dos clases no cubran todos los casos existentes en la información [7]; así mismo, se selecciona Twitter, puesto que contiene un gran volumen de textos disponibles, con api abierta para procesarlos. La escogencia de los usuarios de Twitter pertenecientes a organismos oficiales obedece al interés que puedan tener en identificar de qué están hablando los ciudadanos y, concretamente, de qué se están quejando.

La (Fig. 1) proporciona una visión global del procedimiento sugerido, que consta de las siguientes fases:

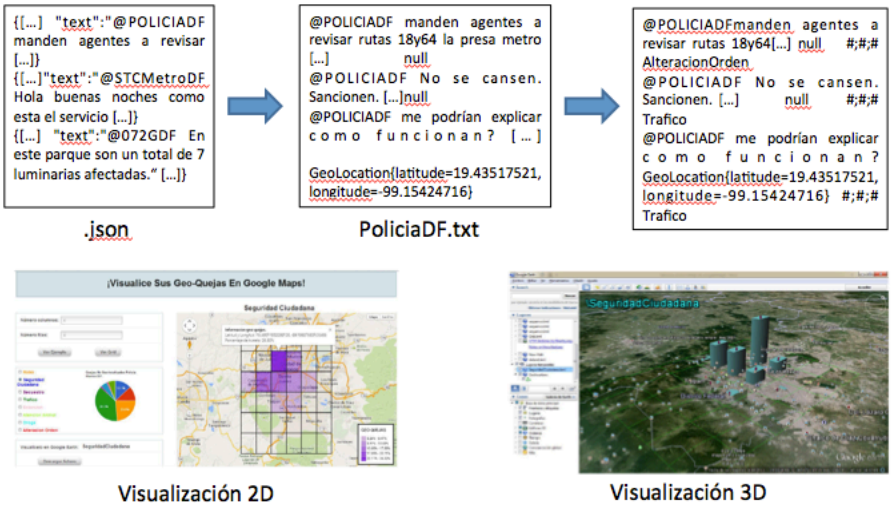

FIG. 1.
Visión global del procedimiento de clasificación y visualización.
Fuente: elaboración propia.

1) Recopilación de datos a través del api de Twitter (.json)
2) Etiquetado de una parte de los datos recopilados de forma supervisada (.txt)
3) Clasificación de forma automática de los datos no etiquetados (svm versus Random Forests)
4) Visualización geolocalizada 2D y 3D de los *tweets* etiquetados.

Al aplicar este procedimiento en 190 000 *tweets* recopilados durante un año de Metro, Protección Civil, Policía, y Gobierno de México (organismos seleccionados por presentar libre acceso), y etiquetar más de 2000 de estos *tweets* a mano, para entrenar en los métodos de clasificación automática, se recomienda el uso de

Random Forests como método de clasificación, ya que se obtiene una precisión media del 81.46 % y una cobertura media del 59.88 %.

Este artículo se compone de las siguientes secciones: en la Sección 2, se recoge el estado del arte; en la Sección 3, se presenta la propuesta de clasificación automática multiclase de *tweets*, y, por último, en la Sección 4, termina el artículo con las principales conclusiones y líneas de trabajo futuro.

2. REVISIÓN DEL ESTADO DEL ARTE

En general, en la clasificación de *microblogging* hay que tener en cuenta la técnica o combinación de técnicas para la recopilación de *tweets* y su clasificación, el dominio, idioma y tipo de clasificador, según el número de clases con las que es capaz de trabajar.

Respecto a la recopilación de *tweets* —con base en que el objetivo es clasificar los tópicos en clases generales para facilitar la recuperación de información—, esta se puede hacer mediante servicios web como What the Trend [8].

De esta manera, todos los *tweets* que contienen un *trending topic* constituyen un documento. En el caso de que un *tweet* contenga más de dos *trending topics*, este se guarda en todos los documentos relevantes.

Se experimentan dos enfoques para la clasificación de tópicos: 1) el enfoque Bag-of-Words [9] para la clasificación de textos y 2) la clasificación network-based [10].

En el método de clasificación basado en textos, se construyen vectores de palabras con definiciones de *trending topics* y *tweets*, y se asignan los pesos tf-idf [11], para clasificar los tópicos mediante el clasificador multinomial Naive Bayes (nb) [12]. En el método de clasificación basado en la red, se identifican los cinco tópicos similares más relevantes para un tópico dado, con base en el número de usuarios influyentes comunes.

Se construyen los modelos predictivos, para lo cual se utilizan varias técnicas de clasificación y se selecciona el que tiene como resultado la mejor precisión de clasificación. Gracias al uso de Naive Bayes Multinomial (nbm) [13], Naive Bayes (nb) [12] y Support Vector Machines (svm-l) [14] con kernels lineales, se obtiene que la precisión de la clasificación es una función del número de *tweets* y la frecuencia de los términos.

Los resultados arrojados fueron: para nbm, con 100 *tweets* y 1000 términos, se obtiene un *accuracy* del 65.36 %. Para svm, con esos mismos datos, se obtiene un 59.81 % y con nb un 45.31 % de *accuracy*.

Al emplear la clasificación network-based con diferentes técnicas de clasificación, se consigue un *accuracy* del 70.96 %, para el árbol de decisión C 5.0 [15] como mejor resultado.

Para mejorar el filtrado de la información, otros autores proponen usar un pequeño conjunto de características específicas del dominio, extraídas a partir del perfil del autor y del texto, para clasificar noticias, eventos y mensajes privados [16].

En general, se puede decir que, para la clasificación, se suelen usar combinaciones de minería de datos y Procesamiento de Lenguaje Natural [17], así como comparaciones de nb, svm y knn entre otras técnicas [18].

Respecto al dominio, suele ser habitual que esté relacionado con emociones [19], opiniones [20], situaciones de emergencia [21] o con análisis de redes sociales [22].

En cuanto al lenguaje, suele ser inglés, aunque también hay casos en los que se dan otros idiomas como turco o multiidioma, con técnicas que pueden aplicarse a cualquier idioma [23].

Sobre el número de clases que puedan clasificarse, lo habitual es que sean dos, esto es, que sean clasificadores biclase.

En los últimos años también se están proponiendo algunos clasificadores multiclase, capaces de clasificar más de dos clases, con lo cual se obtienen valores de precisión entre 68.16 y 73.24 [17], [24].

Por último, se registran mejoras en la clasificación de *tweets*, al tener en cuenta no solo su contenido sino también url, *retweets* y usuarios influyentes [18] y, en general, el multiequitado de la web social con procesamiento multinúcleo [25].

3. PROPUESTA DE CLASIFICACIÓN AUTOMÁTICA MULTICLASE DE TWEETS

Es importante destacar que, normalmente, en la literatura se encuentran clasificadores biclase limitados en el dominio, en el idioma y las técnicas que utilizan. La propuesta de este trabajo está contextualizada en el dominio social, particularmente, en las quejas.

Se centra en el lenguaje castellano, pero, como en [23], las técnicas se pueden aplicar a otros idiomas. Utiliza como técnicas svm y Random Forests, como en [17], que está basado en técnicas de minería de datos y Procesamiento de Lenguaje Natural, o en [18], que usan svm, entre otras técnicas.

De esta manera, la principal contribución propuesta es un clasificador multiclase para el dominio social, en castellano, aplicable a otros idiomas, que sigue los pasos principales de Minería de Datos, que se irán describiendo en detalle en los siguientes subapartados:

1) Selección del conjunto de datos, tanto en lo que se refiere a las variables que se quieren predecir como a las variables que sirven para hacer el cálculo.

2) Transformación del conjunto de datos de entrada, también conocido como pre-procesamiento de los datos, con el objetivo de prepararlo para aplicar la técnica de Minería de Datos que mejor se adapte a los datos y al problema.

3) Selección y aplicación de la técnica de Minería de Datos, y construcción del modelo predictivo de clasificación.

4) Extracción de conocimiento mediante una técnica de Minería de Datos, se obtiene un modelo de conocimiento, que representa patrones de comportamiento observados en los valores de las variables del problema

5) Interpretación y evaluación de datos.

Una vez obtenido el modelo, se procede a su validación, tras comprobar que las conclusiones que arroja son válidas y satisfactorias.

3.1 Recogida de datos

En primer lugar, se recogieron de forma paralela *tweets* de dos formas diferentes a, través de las funciones que proporciona el api Stream de Twitter. Con base en la idea que se planteó —identificar las quejas de los usuarios de México D. F.—, se recopilaron de forma genérica todos los *tweets* localizados dentro de las coordenadas que cercan la ciudad de México D. F., mediante la opción que nos proporciona la función post statuses/filter, llamada "locations".

De forma adicional y paralela, también se recopilaron los *tweets* de algunos de los organismos oficiales que se encuentran en México D. F. Para eso, se buscaron los organismos oficiales existentes en México; finalmente, se seleccionaron cuatro de ellos, en los que se intuyó podría haber más posibilidades de recopilar quejas de usuarios. Los usuarios seleccionados son: el Metro, la Protección Civil, el Gobierno y la Policía.

Para conseguir la recopilación de estos *tweets*, se empleó otra función llamada "follow" que recoge todos los *tweets* pertenecientes a un identificador (id) único para cada uno de los usuarios.

Con este objetivo, se desarrolló un *script* que se ejecutó de forma continua durante un año, al almacenar los *tweets* en ficheros con extensión ".json".

3.2 Extracción de datos

Una vez terminada la recogida de datos, se procedió al tratamiento y extracción de los datos necesarios para hacer la clasificación. Para atender a la estructura de un *tweet* y a la cantidad de campos que contiene, el procedimiento a seguir fue el estudio de cada uno de esos campos a fin de establecer cuáles eran los necesarios para este proceso de clasificación y cuáles no.

Finalmente, se decidió que los únicos campos necesarios iban a ser el texto del *tweet*, representado por el campo "text", que contiene una longitud de 140 caracteres alfanuméricos, y la ubicación geográfica del *tweet*, representada por el campo "coordinates", que contiene la latitud y longitud de su ubicación. Se tomó esta decisión, de acuerdo al objetivo marcado inicialmente para la recogida y clasificación de *tweets* geolocalizados, con lo que el campo "coordinates" proporciona la geo-localización del *tweet,* que servirá para ubicarlo en la visualización, y el campo "text" proporciona la información necesaria para la identificación de quejas.

Para poder extraer cada uno de estos campos del fichero. json, se empleó una librería desarrollada en código Java llamada "twitter4j" [26], cuya funcionalidad permite procesar el api de Twitter.

El texto y la geolocalización de todos los *tweets* recogidos para cada uno de los usuarios específicos se guardan en un fichero ".txt", para su posterior clasificación.

De esta forma, se crean cuatro ficheros ".txt", que contienen un total de 34 839 *tweets* para el Gobierno de México D. F., 123 873 para el Metro, 4122 para la Protección Civil y, finalmente, 13 944 *tweets* para la Policía.

3.3 Aplicación de técnicas de Procesamiento de Lenguaje Natural

Con los datos guardados en el fichero ".txt", el siguiente paso es aplicar técnicas estadísticas de Procesamiento de Lenguaje Natural a los textos [3]. Este tipo de procesamiento representa el modelo clásico de los sistemas de recuperación, y se caracteriza porque cada documento está descrito por un conjunto de palabras clave denominadas *término índice*. En este modelo, el procesamiento de los documentos consta de las siguientes etapas:

—Preprocesado de los documentos: se eliminan aquellos elementos que se consideran superfluos. Consta de tres fases básicas:

-Eliminación de elementos del documento que no son objeto de indexación; en este caso, pueden ser etiquetas, enlaces http, etc.

-Normalización de textos, que consiste en homogeneizar todo el texto e identificar N-Gramas que pueden ser unigramas (1-gramas), bigramas o digramas (2-gramas), trigramas (3-gramas), etc.

-Lematización de los términos, cuyo objetivo es reducir una palabra a su raíz mediante algoritmos de radicación o *stemming*, que permiten representar de un mismo modo las distintas variantes de un término, con el fin de reducir el tamaño del vocabulario y mejorar, en consecuencia, la capacidad de almacenamiento de los sistemas y el tiempo de procesamiento de los documentos.

—Parametrización: se hace una cuantificación de las características (es decir, de los términos) de los documentos, mediante la asignación de un peso a cada uno de los términos relevantes de un documento. El peso de un término se calcula normalmente mediante la función tf-idf (en inglés, Term Frecuency-Inverse Document Frecuency) [5], que consiste en una medida numérica que expresa cuán relevante es una palabra para un documento en una colección. El valor tf-idf aumenta proporcionalmente al número de veces que una palabra aparece en el documento, pero es compensada por la frecuencia de la palabra en la colección de documentos, lo que permite manejar el hecho de que algunas palabras son generalmente más comunes que otras.

3.4 Métodos de identificación de clases

Con los textos preprocesados, el siguiente paso para poder iniciar el proceso de clasificación es identificar cuáles van a ser las clases bajo las cuales se van a catalogar los *tweets*. Para eso, se emplearon dos técnicas: k-means y nubes de palabras, que se describen a continuación.

El método k-means tiene como objetivo la partición de un conjunto n en k grupos, en el que cada observación pertenece al grupo más cercano a la media. Una de las aplicaciones de este algoritmo es emplearlo como preprocesamiento para otros algoritmos, por ejemplo, para buscar una configuración inicial.

Es en este aspecto, cobra sentido la aplicación de este método en la identificación de clases para la colección de *tweets*. Se empleó k-means para obtener varios grupos de términos y observar si entre ellos se encontraba alguno que pudiera ser identificado como queja.

Para la obtención de diferentes *clusters* que pudieran ofrecer una idea inicial de algunas de las clases, se probó con múltiples números *clusters*, hasta que, con el número 12, se obtuvo una muestra orientativa de *clusters* que contenían un conjunto de seis términos en cada uno de ellos, y se hizo la función k-means, también disponible en el programa R.

En los resultados de los diferentes *clusters* obtenidos, se identificaron *a priori* quejas como robo, maltrato animal, circulación, extorsión, abandono. Con esta idea inicial, se consiguieron algunas de las etiquetas que posteriormente serían verificadas como clases, con las que se etiquetaría el *training set*.

El método nubes de palabras (*wordclouds*) permite obtener una representación visual, en forma de nubes de palabras, sobre la frecuencia con la que se repite cada una de las palabras.

La Fig. 2 muestra un ejemplo de nubes de palabras, identificadas para el usuario Policía. Estas etiquetas se pueden usar de forma complementaria a las identificadas en los k-means.

FIG. 2.
Ejemplo de identificación de clases, obtenida con k-means.
Fuente: elaboración propia.

El resto de los *tweets*, que no pertenecían a ninguna de las clases identificadas, fueron etiquetados con la clase "No Etiquetado", que correspondía con un alto porcentaje de los *tweets*. Debido a esto, se identificó el problema de las Clases No Balanceadas, que ocurre cuando en un problema de clasificación hay muchas más instancias de unas clases que de otras.

Las clases que tienen minoría de instancias son las que rara vez aparecen, pero son, asimismo, las que más importancia tienen [27].

3.5 Clasificación automática multiclase

El siguiente paso es obtener un clasificador de forma supervisada, para poder clasificar de forma automática los tweets recogidos. En este ámbito, cabe aclarar que los métodos de clasificación automática se engloban dentro del concepto de Aprendizaje Automático, cuyo objetivo es crear programas capaces de generalizar comportamientos, a partir de una información no estructurada. Este es, por lo tanto, un proceso de inducción al conocimiento.

En el Aprendizaje Automático existen técnicas de clasificación supervisada y no supervisada. La clasificación supervisada cuenta con modelos ya clasificados, que permiten clasificar los no clasificados.

Se pueden diferenciar dos fases en este tipo de clasificación:

1) En la primera fase, se dispone de un conjunto de entrenamiento o de aprendizaje y de otro llamado de test o de validación; estos sirven para construir un modelo o regla general para la clasificación. El proceso de entrenamiento es cuando un clasificador debe aprender cómo clasificar los objetos, al generalizar, a partir de los datos de entrenamiento, las situaciones no vistas.

2) En la segunda fase, se clasifican los objetos o muestras de las que se desconoce la clase a la que pertenecen.

La salida de un clasificador supervisado puede ser la etiqueta de la clase del nuevo objeto clasificado, un conjunto de etiquetas ordenadas por la probabilidad de ser la etiqueta correcta, así como un vector numérico, en el que cada valor representa el valor de pertenencia otorgado por el clasificador a cada clase.

En la actualidad, existen diversos clasificadores supervisados. Entre los más usados, se encuentran el Vecino más Cercano (KNN), las Redes Neuronales (ANN), el clasificador Bayesiano (NB), los Random Forests (RF), y la Máquina de Soporte de Vectores (SVM).

A diferencia de la clasificación supervisada, la clasificación no supervisada no cuenta con conocimiento a priori, por lo que se tiene un área de conocimiento disponible para la tarea de clasificación.

A la clasificación no supervisada se le suele llamar también clustering. En este tipo de clasificación, se cuenta con "objetos" o muestras que tienen un conjunto de características, de las que no se sabe a qué clase o categoría pertenecen; en razón a esto, su finalidad es el descubrimiento de grupos de "objetos" cuyas características afines permitan separar las diferentes clases.

Para la clasificación de las clases identificadas en la sección anterior, se optó por la técnica de clasificación supervisada.

En particular, se etiquetó un conjunto de 2000 tweets de un total de 13 944, recogidos de la Policía de forma manual y leído texto por texto. En estos casos, juega un papel importante la objetividad para identificar a qué etiqueta pertenece cada uno de los tweets; por este motivo, el encargado de etiquetarlos debe hacerlo con la mayor objetividad posible.

Posteriormente, para entrenar los clasificadores, los pasos a realizar son:

1) Dado un conjunto de 2000 tweets etiquetados por completo, se subdivide en dos conjuntos de tweets diferentes: el conjunto de entrenamiento (trainingset), que contiene el 60 % de los tweets seleccionados de forma aleatoria, y el conjunto de testeo (testset) que contiene el restante 40 % de los tweets. El conjunto de entrenamiento abarca las clases etiquetadas, en cambio, en la predicción, el conjunto de testeo no.

2) Esto se hace para ver qué precisión y cobertura alcanza el clasificador sobre una muestra inicial.

3) Se aplica el clasificador sobre el conjunto de entrenamiento y, a continuación, una predicción entre el resultado obtenido por el clasificador y el conjunto de testeo.

4) Mediante una matriz de confusión, en la que cada columna representa el número de predicciones de cada clase y cada fila el número de instancias de la clase real, se visualizan los resultados predichos y se calcula la precisión y la cobertura para cada una de las clases, a fin de obtener, finalmente, una media de la precisión y una media de la cobertura.

Con el conjunto de entrenamiento etiquetado, se procedió a hacer un estudio de SVM y Random Forests como clasificadores a entrenar para clasificar los tweets. Los resultados recogidos en las (Fig. 3) y (Fig. 4)

indican que, tanto en el caso de aplicar o no función de pesos, Random Forests obtiene mejores resultados, por lo que fue el clasificador escogido. Se pueden encontrar más tablas en [28].

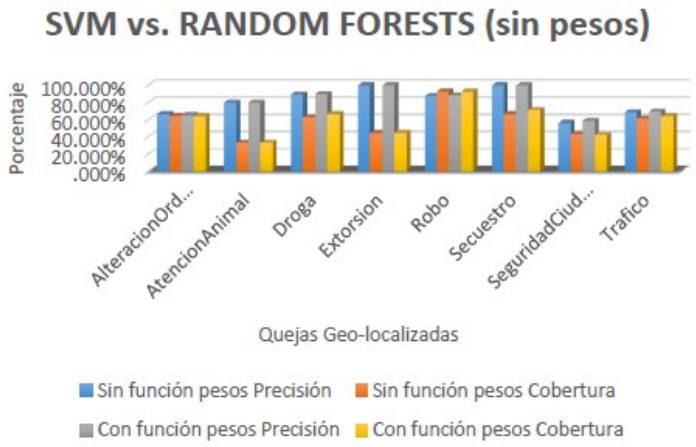

FIG. 3.
Comparación de los resultados de los clasificadores sin aplicar la función de pesos
Fuente: elaboración propia.

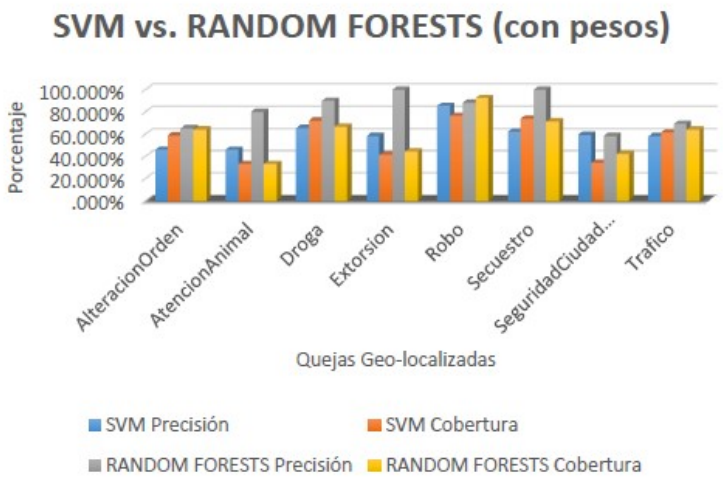

FIG. 4.
Comparación de los resultados de los clasificadores con pesos
Fuente: elaboración propia.

Para este proceso de clasificación, en este caso no se selecciona el 40 % de las muestras aleatoriamente, para separar el conjunto de testeo del conjunto de entrenamiento. En su lugar, el conjunto de entrenamiento serán los tweets etiquetados de forma manual y el conjunto de testeo los tweets que se van a etiquetar de forma automática.

Tras hacer la predicción de la clasificación automática, se construye una matriz que contiene una nueva columna añadida con la clase a la que pertenece cada uno de los *tweets* no etiquetados inicialmente, con el fin de añadir esa columna al fichero de *tweets* y hacer las visualizaciones como se explica en el siguiente apartado.

4. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha investigado el procedimiento para una clasificación automática multiclase de los *tweets* recogidos de algunos de los organismos oficiales de México D. F., cuyos datos almacenados rondan los 190 000 *tweets* acopiados a lo largo de un año.

Con este objeto, se hizo un estudio comparativo de varios clasificadores supervisados según sus resultados de precisión y cobertura, lo que arrojó que, Random Forests, con unos resultados de precisión en el entrenamiento entre el 58.46 % y el 100 %, y de cobertura de entre el 33.33 % y el 92.68 %, es el clasificador propuesto.

Estos resultados son interesantes para el estado del arte, puesto que, al revisar la literatura, los datos se suelen limitar a valores de *accuracy* y no de precisión-cobertura por separado.

En particular, el trabajo mayormente vinculado al desarrollo de esta parte del proyecto es el adelantado por Malkani & Gillie en 2012, que emplea svm multiclase y Random Forests, para la clasificación de dos conjuntos de datos con clases diferentes centrados en tópicos y actitudes.

En este caso, se han empleado estos clasificadores de forma diferente. svm se probó con one-vs-one y distintos tipos de kernel, aunados a la función pesos y Random Forests, con 150 árboles aleatorios; asimismo, la función pesos fue usada para identificar ocho clases diferentes de quejas en usuarios específicos.

En consecuencia, la principal contribución al estado del arte radica en la clasificación multiclase al usar clasificadores Support-Vector Machines (svm) multiclase y Random Forests (rf) para 35 000 *tweets*, para la identificación de quejas. De lo anterior, se obtuvo como resultado:

—Con svm:

-Precisión: entre 55.83 % y 92.26 %

-Cobertura: entre 33.33 % y 76.53 %

—Para rf:

-Precisión: entre 58.46 % y 100 %

-Cobertura: entre 33.33 % y 92.68 %.

Además, las técnicas utilizadas son aplicables a otros idiomas y dominios.

El código y el *dataset* se han publicado en [29] para otros investigadores que requieran avanzar en sus estudios.

Respecto al trabajo futuro, se está trabajando en mejorar la precisión y cobertura obtenidas con otros métodos de clasificación y añadir mayor variedad de organismos oficiales.

6. REFERENCIAS

[1] S. Galeano, "Cuáles son las redes sociales con más usuarios del mundo (2019)," *M4rketing Ecommerce*, 2019. Disponible en: https://marketing4ecommerce.net/cuales-redes-sociales-con-mas-usuarios-mundo-2019-top/. [Accedido: 27-Jan-2020].

[2] K. Smith, "44 estadísticas de Twitter," *Brandwatch*, 2016. Disponible en: https://www.brandwatch.com/es/blog/44-estadisticas-twitter/ [Accedido: 27-Jan-2020].

[3] C. D. Manning y H. Schiitze, *Foundations of Statistical Natural Language Processing: Massachusetts Institute of Technology*: MIT Press. Cambridge, 1999. Disponible en: https://www.cs.vassar.edu/~cs366/docs/Manning_Schuetze_StatisticalNLP.pdf

[4] M. Vallez y R. Pedraza-Jimenez, "El Procesamiento del Lenguaje Natural en la Recuperación de Información Textual y áreas afines," *Hipertext.net*, vol. 5, 2007. Disponible en: https://www.raco.cat/index.php/Hipertext/article/view/59496

[5] tf-idf, "What does tf-idf mean?". Disponible en: http://www.tfidf.com/cgi-sys/suspendedpage.cgi. [Accedido: 27-Jan-2020].

[6] C. C. Aggarwa y C. Zhai, *Mining Text Data*: Boston, MA: Springer US, 2012. https://doi.org/10.1007/978-1-4614-3223-4

[7] Z. Malkani y E. Gillie, "*Supervised Multi-Class Classification of Tweets*," pp. 1–6, Dec. 2012. Disponible en: https://pdfs.semanticscholar.org/bc78/1a147a3fe8477ade06ccf22a3aabe12236ea.pdf

[8] Twitter, "What The Trend," 2009. Disponible en: https://twitter.com/whatthetrend

[9] K. Lee, D. Palsetia, R. Narayanan, M. M. A. Patwary, A. Agrawal, y A. Choudhary, "Twitter Trending Topic Classification," en *2011 IEEE 11th International Conference on Data Mining Workshops*, Vancouver 2011. pp. 251–258. https://doi.org/10.1109/ICDMW.2011.171

[10] Y. Zhu, X. Shen, y W. Pan, "Network-based support vector machine for classification of microarray samples," *BMC Bioinformatics*, vol. 10, no S21, Jan. 2009. https://doi.org/10.1186/1471-2105-10-S1-S21

[11] J. Ramos, "Using tf-idf to determine word relevance in document queries," en *Proceedings of the first instructional conference on machine learning*, Piscataway, 2003, pp. 133–142.

[12] I. Rish, "An empirical study of the naive Bayes classifier," en *IJCAI 2001 workshop on empirical methods in artificial intelligence*, 2001, pp. 41–46. Disponible en: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.330.2788

[13] E. Anguiano-Hernández, *Naive Bayes Multinomial para clasificación de texto usando un esquema de pesado por clases*, pp.1-8, Apr. 2009. Disponible en: http://ccc.inaoep.mx/~esucar/Clases-mgp/Proyectos/MGP_RepPro y_Abr_29.pdf

[14] N. Cristianini y J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods* Cambridge: University Press, 2000. https://doi.org/10.1017/CBO9780511801389

[15] RuleQuest Research "*About us*," 2018. Disponible en: https://rulequest.com/about-us.html. [Accedido: 21-Sep-2019].

[16] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, y M. Demirbas, "Short text classification in twitter to improve information filtering," en *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval - SIGIR '10*, Geneva, 2010, pp. 841–842. https://doi.org/10.1145/1835449.1835643

[17] J. Nazura y B. L. Muralidhara, "Semantic classification of tweets: A contextual knowledge based approach for tweet classification," en *2017 8th International Conference on Information, Intelligence, Systems & Applications (IISA)*, Larnaca, 2017, pp.1-6. https://doi.org/10.1109/IISA.2017.8316358

[18] P. Selvaperumal y A. Suruliandi, "A short message classification algorithm for tweet classification," en *2014 International Conference on Recent Trends in Information Technology*, Chennai, 2014. pp. 1–3. https://doi.org/10.1109/ICRTIT.2014.6996189

[19] R. C. Balabantaray, M. Mohammad, y N. Sharma, "Multi-Class Twitter Emotion Classification: A New Approach," *Int. J. Appl. Inf. Syst.*, vol. 4, no. 1, pp. 48–53, Sep. 2012. https://doi.org/10.5120/ijais12-450651

[20] E. D'Andrea, P. Ducange, A. Bechini, A. Renda, y F. Marcelloni, "Monitoring the public opinion about the vaccination topic from tweets analysis," *Expert Syst. Appl.*, vol. 116, pp. 209–226, Feb. 2019. https://doi.org/10.1016/j.eswa.2018.09.009

[21] M. Habdank, N. Rodehutskors, y R. Koch, "Relevancy assessment of tweets using supervised learning techniques: Mining emergency related tweets for automated relevancy classification," en *2017 4th International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*, Münster, 2017, pp. 1–8. https://doi.org/10.1109/ICT-DM.2017.8275670

[22] J. F. Franco-Bermúdez y W. L. Ruiz-Castañeda, "Análisis de redes sociales para un sistema de innovación generado a partir de un modelo de simulación basado en agentes," *TecnoLógicas*, vol. 22, no. 44, pp. 21–44, Jan. 2019. https://doi.org/10.22430/22565337.1183

[23] R. S. Ghaly, E. Elabd, y M. A. Mostafa, "Tweets classification, hashtags suggestion and tweets linking in social semantic web," en *2016 SAI Computing Conference (SAI)*, London, 2016. pp. 1140–1146. https://doi.org/10.1109/SAI.2016.7556121

[24] E. Yar, I. Delibalta, L. Baruh, y S. S. Kozat, "Online text classification for real life tweet analysis," en *2016 24th Signal Processing and Communication Application Conference (SIU)*, Zonguldak, 2016. pp. 1609–1612. https://doi.org/10.1109/SIU.2016.7496063

[25] J. M. Rodriguez, D. Godoy, C. Mateos, y A. Zunino, "A multi-core computing approach for large-scale multi-label classification," *Intell. Data Anal.*, vol. 21, no. 2, pp. 329–352, Mar. 2017. https://doi.org/10.3233/IDA-150375

[26] Twitter4J.org, "Overview". Disponible en: http://twitter4j.org/javadoc/index.html

[27] R. Longadge, S. Dongre y L. Malik, "Class Imbalance Problem in Data Mining Review," *Int. J. Comput. Sci. Netw.*, vol. 2, no. 1, pp. 83–87, May, 2013. Disponible en: http://journaldatabase.info/articles/class_imbalance_problem_data_mining.html

[28] B. Hernández-Pajares, "Clasificación Automática Multiclase de Tweets y su Representación Gráfica,"(Tesis de Maestría), Facultad de ingeniería, Madrid, Universidad Rey Juan Carlos, 2013. Disponible en: https://ecienciа.urjc.es/handle/10115/11914

[29] B. Hernández-Pajares, D. Pérez-Marín y V. Frías-Martínez, "TFM_code", 2013. Disponible en: https://urjc-my.sharepoint.com/personal/diana_perez_urjc_es/_layouts/15/onedrive.aspx?id=%2Fpersonal%2Fdiana%5Fperez%5Furjc%5Fes%2FDocuments%2Fpublicaciones%2F2020%2Ftecnologicas%2Fcode%2Ezip&parent=%2Fpersonal%2Fdiana%5Fperez%5Furjc%5Fes%2FDocuments%2Fpublicaciones%2F2020%2Ftecnologicas&originalPath=aHR0cHM6Ly91cmpjLW15LnNoYXJlcG9pbnQuY29tLzp1Oi9nL3BlcnNvbmFsL2RpYW5hX3BlcmV6X3VyamNfZXMvRVhBb0JNSzJuSU5FbjZuaXoxenNMaTBCb3lWQzc5RmdzUFQ0dk1UbmJjdEFVQT9ydGltZT01X3BzMmtiTTEwZw

Información adicional

Cómo citar / How to cite: B. Hernández-Pajares, D. Pérez-Marín y V. Frías-Martínez, "Clasificación multiclase y visualización de quejas de organismos oficiales en twitter", *TecnoLógicas*, vol. 23, no. 47, pp. 109-120, 2020. https://doi.org//10.22430/22565337.1454