EVALUACIÓN DE ALGORITMOS DE CLASIFICACIÓN SUPERVISADA PARA EL MINADO DE OPINIÓN EN *TWITTER*

Suilan Estévez-Velarde y Yudivián Almeida Cruz
Universidad de La Habana, Cuba.

ABSTRACT

This paper proposes a comprehensive evaluation of supervised classification algorithms without semantic information for opinion mining on Twitter. The evaluation will assess the potential of this approach to tackle the given problem. Also, as result of evaluating, the impact of different variants of the text preprocessing is assessed, as well as different alternatives of dimension reduction techniques in the problem solution. Finally, the results obtained in the experiments offer evidence to direct the use of supervised classification to classify opinions and its valence in a domain of short messages such as Twitter.

KEY WORDS: machine learning, Twitter, opinion mining, classification

MSC: 68T05

RESUMEN

En este trabajo se propone la evaluación de distintos algoritmos de clasificación supervisada para abordar el problema de la minería de opinión en *Twitter*. Se analizará el potencial del enfoque supervisado, sin hacer uso de información semántica, en este dominio particular. Asimismo, como resultado del estudio, se abordará el impacto de las diferentes variantes del preprocesamiento textual, así como distintas alternativas para la reducción de dimensiones en la solución del problema. Finalmente, los resultados obtenidos en los experimentos ofrecen evidencia para la utilización del aprendizaje supervisado en la clasificación de las opiniones y su valencia en un dominio de mensajes cortos como *Twitter*.

1. INTRODUCCIÓN

Twitter es en la actualidad uno de los grandes protagonista de la red global. Esta plataforma de comunicación ha establecido, a base de popularidad, una nueva forma de comunicación: el *microblogging*. Luego de más de 8 años de uso y más de 500 millones de usuarios, *Twitter* se ha convertido en una plataforma esencial para el seguimiento, difusión y coordinación de eventos de diversa naturaleza e importancia [3] como puede ser una campaña presidencial, una situación de desastre, un conflicto bélico o la repercusión de una información. Un escenario así es considerado una fuente de información relevante para conocer las opiniones que se emiten sobre distintos asuntos o personas.
El flujo de información existente en esta red social es tal que no es práctico el procesamiento directo de este flujo sin el auxilio de sistemas computacionales. Para afrontar esta tarea son necesarios sistemas que realicen análisis de forma automática. Muchos de estos análisis tienen el objetivo de extraer información global de la red para valorar diversos temas. Para ello, en muchos casos, se utilizan las técnicas de minería de opinión, de lo cual la literatura recoge varios trabajos [3, 22]. Existen distintas aproximaciones al minado de opinión en *Twitter*, una de las más utilizadas es la

clasificación supervisada [30]. Sin embargo, a partir de los distintos trabajos consultados, aún no se identifica una solución universalmente reconocida como la mejor aproximación en la solución de este problema.
En esta línea, este trabajo propone una evaluación de algoritmos de clasificación supervisada para la solución del problema de minería de opinión en *Twitter*. Para ello se seleccionaron algoritmos representativos de la mayoría de los enfoques existentes en la clasificación supervisada, priorizando en cada caso las variantes clásicas. La evaluación de esta selección permitirá valorar las posibilidades de la clasificación supervisada para, sin utilizar información semántica alguna, enfrentar el problema de minado de opinión en este dominio particular. Además, como resultado de la evaluación, se obtendrán evidencias que permitirán la formulación de nuevas propuestas que, para la solución del problema de minería de opinión en *Twitter*, integren la clasificación supervisada de conjunto con otras técnicas.

2. MINERÍA DE OPINIÓN

La Minería de Opinión es una rama de la Minería de Textos que se enfoca en analizar y extraer los juicios de opinión expresados en un texto [27]. No está necesariamente asociada a un tema particular, lo que determina el análisis son los rasgos del texto. Además, dado un análisis fundamentalmente sintáctico es más relevante la forma que el dominio [23]. Regularmente los procesos minería de opinión se dividen en tres fases:

Representación de los datos: Los datos se representan en una estructura que sea común a todos los procesos de análisis. Para ello se utilizan varias técnicas, la más común consiste en un vector binario con todas las palabras del dominio (1 si la palabra aparece en el mensaje, 0 si no aparece) [17] o vectores con la cantidad de ocurrencias de la palabra en el texto [10].

Normalización de los datos: Las irregularidades que suelen presentar los datos dificultan el proceso de minería de opinión es por ello que estos precisan ser homogenizados en un preprocesamientos previo. El preprocesamiento depende, en gran medida, de los datos con los que se trabaje. Se han propuesto diferentes variantes de preprocesamiento para atenuar los problemas inherentes a los datos [7, 9]. Los mensajes de *Twitter* (*tweets*) se caracterizan por contener distorsiones del idioma, *urls* y emoticones, que dificultan el análisis que se desea realizar sobre el contenido del mensaje [14]. Estos son los elementos susceptibles de ser homogenizados durante el preprocesamiento.

Clasificación: La fase final consiste en otorgar a cada mensaje una categoría, para ello se utilizan técnicas de aprendizaje automático. Se han utilizado tanto clasificadores no supervisados, supervisados como semi-supervisados[23]. El tipo de aprendizaje más utilizado para documentos de longitud similar a los *tweets* (e.g. las oraciones) es la clasificación supervisada (tabla 1).

2.1. Minería de opinión en Twitter

Distintas investigaciones se han enfocando en realizar estudios de minería de opinión teniendo como fuente de datos a los *tweets*. Habitualmente, los usuarios de *Twitter* son inducidos a opinar sobre productos, servicios y política [21]. De esta forma los *tweets* se convierten en una fuente interesante para el análisis de sentimientos. Como los mensajes tienen una corta longitud, de aproximadamente una oración, se puede asumir que expresan una única idea [12]. Por tanto a cada mensaje se le asigna una sola opinión, como simplificación del problema.
Una aproximación se basa en reconocer palabras subjetivas y *hashtags* con significado subjetivo. Además se propone aplicar reglas para el tratamiento de juicios comparativos, la negación y otras expresiones que cambian la orientación de la frase [35]. Igualmente, en el análisis de sentimiento sobre *tweets* se han empleado también técnicas basadas en ontologías [16].

Problema	A nivel de:	Tipo de Aprendizaje	Referencia
Determinar la subjetividad	Documento	Supervisado	[34]
	Oración	Supervisado	[34, 26, 33]
	Término	Supervisado	[8]
Determinar la polaridad	Documento	Supervisado	[24]
		No Supervisado	[29]
	Oración	Supervisado	[1]
		No Supervisado	[29]
		Semi-Supervisado	[1]
	Término	Semi-Supervisado	[8]
Determinar el grado	Término	Supervisado	[14]

Cuadro 1: Tipos de aprendizaje automático para minería de opinión.

Otro de los estudios sobre la clasificación de mensajes de *Twitter*, para determinar su polaridad, conduce a utilizar métodos de clasificación supervisados [12]. Algunas propuestas sugieren el uso de n-gramas, combinado con algoritmos aprendizaje como Naive Bayes y la utilización de POS-TAGS como características del *tweet* [21]. Un estudio sobre métodos híbridos de clasificación plantea la existencia de dos paradigmas, uno basado en uso de recursos léxicos y otro basado en técnicas de aprendizaje automático [35].

3. METODOLOGÍA DE MINADO DE OPINIÓN EN *TWITTER*

Se puede notar que una de las técnicas más utilizada para el minado de opinión de mensajes con longitud similar a los *tweets* es la clasificación supervisada [23]. Sin embargo, ninguna de las aproximaciones al problema de minería de opinión estudiados se muestra superior al resto. Entonces, la evaluación de los algoritmos representativos de la clasificación supervisada en el minado de opinión en *Twitter* sería un resultado a tener en cuenta en futuras soluciones a este problema. Para ello es imprescindible diseñar una metodología de evaluación que tenga en cuenta tanto las dinámicas de *Twitter* como las del proceso de minería de opinión. Así, para la evaluación de las técnicas de clasificación supervisada, el proceso de minería de opinión en *Twitter* que se propone es dividido en tres etapas (figura 1):

1. Preprocesamiento o normalización de los datos.
2. Reducción de dimensiones:
   - Descomposición.
   - Agrupamiento de los términos.
3. Clasificación supervisada.

En cada etapa pueden ser utilizados diferentes algoritmos dependiendo de las características del problema a resolver en cada una de ellas.

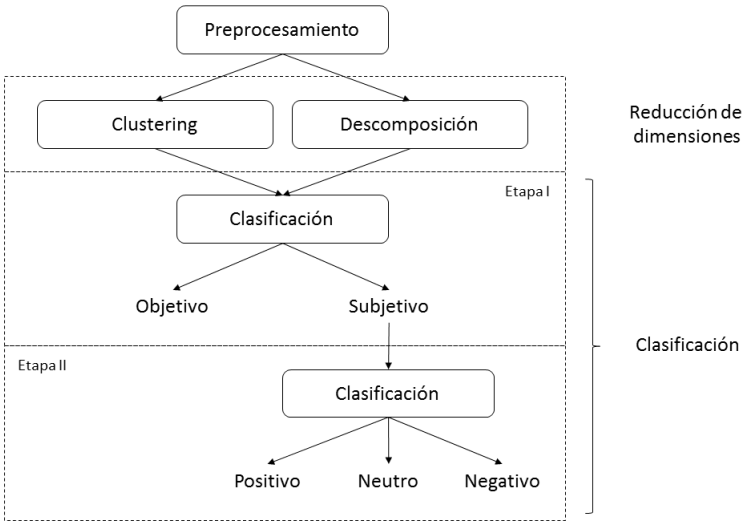

Figura 1: Esquema de la Metodología.

3.1. Normalización de los datos

En *Twitter* los mensajes no están sujetos a las estrictas reglas sintácticas del lenguaje [14], lo que dificulta la aplicación de algoritmos de minería de opinión sobre ellos. Por tanto, la normalización de los *tweets* en un texto apropiado puede incidir en su posterior procesamiento.
El modelo de normalización que se propone consta de 7 fases independientes y opcionales:

- Eliminar las etiquetas de *Twitter* (*hashtags*) y URLs.
- Separar los emoticones del texto.
- Traducir la jerga.
- Suprimir las letras repetidas.
- Eliminar las *stop words*.
- Aplicar corrección ortográfica.
- Aplicar *stemming*.

Una vez normalizado el conjunto de datos, el mismo es representado utilizando un modelo de bolsa de palabras. Así se obtiene una matriz donde las columnas representan los *tweets* y las filas representan el conjunto de los términos o palabras presentes en el universo de los *tweets* analizados.

3.2. Reducción de Dimensiones

En la representación matricial de los datos, donde los documentos están representados como vectores de términos, dichos documentos se expresan en la base canónica de los términos. Así, este espacio tiene una enorme dimensión y, dado que en este caso los documentos (*tweets*) solo poseen 140 caracteres, está muy esparcido.
Una alta dimensión de los datos afecta el funcionamiento de varios clasificadores [18]. Entonces es necesario valorar la utilización de algoritmos para el proceso de reducción de dimensiones. Se analizaron, fundamentalmente, dos aproximaciones: algoritmos de descomposición de matrices, desechando las componentes menos importantes, y algoritmos de *clustering* sobre el espacio de las palabras, que agrupan los términos semejantes.
Los algoritmos de descomposición a evaluar fueron:

Linear Discriminant Analysis (LDA): permite la reducción de dimensiones supervisada, mediante la proyección de la entrada en un subespacio compuesto por las direcciones más discriminantes. El criterio de selección de dimensiones se basa en la estimación de la probabilidad $P(X|y)$ de observar la entrada X en la clase y, donde X es un vector de características (términos). Estas probabilidades se modelan como distribuciones gaussianas con matrices de covarianza idénticas [19].

Principal Component Analysis (PCA): proyecta los documentos en el subespacio generado por los k valores propios más significativos de la descomposición SVD de la matriz de características [6].

Randomized PCA: basado en PCA, pero empleando una descomposición SVD estocástica (aproximada), más eficiente en matrices muy esparcidas [25].

Sparse PCA: encuentra las componentes esparcidas que permiten reconstruir la matriz de características original de manera óptima. A diferencia de LDA, no garantiza ortogonalidad en la base del subespacio generado [25].

Independent Component Analysis (ICA): encuentra las componentes que maximizan la cantidad de información independiente. A diferencia de los métodos de componentes principales, que maximizan la varianza explicada por el subespacio generado, ICA permite reconstruir distribuciones no gausianas [6].

Mientras tanto los algoritmos de *clustering* a evaluar fueron:

Growing Neural Gas (GNG): es un algoritmo de agrupamiento basado en distancias que intenta solventar la restricción de algoritmos como K-means de predefinir la topología de los grupos. Presenta dos grandes ventajas: detecta de forma automática la cantidad de grupos, y los grupos pueden tener cualquier topología [11].

DBSCAN: el algoritmo DBSCAN (*Density-Based Spatial Clustering for Applications with Noise*) pertenece a la familia de los llamados algoritmos de agrupamiento basados en densidad. Estos consideran la formación de grupos a partir de la concentración de puntos en una vecindad ϵ de cada ejemplo. DBSCAN permite detectar de forma automática la cantidad de grupos, y se adapta a la topología de los datos [2].

Ward: es un algoritmo de *clustering* jerárquico aglomerativo (*bottom-up*) que mezcla los *clusters* minimizando una métrica similar a K-means. Es mucho más eficiente que K-means para un número elevado de *clusters* [32].

K-means: construye una solución aproximada al problema de minimización de distancia inter-grupo, escogiendo k centroides de forma que cada elemento se asigne al grupo relacionado al centroide más cercano, y los centroides estén tan alejados entre sí como sea posible, a partir de una estrategia basada en *Hill Climbing* [15].

Affinity Propagation: determina los *clusters* mediante el envío de mensajes entre pares de ejemplos hasta converger. De esta forma un conjunto de datos es descrito empleando un número pequeño de ejemplos, que son identificados como los más representativos del resto de los ejemplos [31].

Mean Shift: determina los *clusters* mediante la estimación de parches en una matriz de puntos de densidad suave. Este algoritmo calcula la cantidad de *clusters* necesarios de forma automática [5].

Spectral Clustering: realiza una inmersión en una dimensión menor de la matriz de similaridad entre ejemplos, seguida de la ejecución de K-means en los vectores de baja dimensión [20].

3.3. Clasificación

La clasificación es el proceso final donde se determina en cada mensaje (*tweet*) la existencia o no de una opinión (subjetividad u objetividad) y su posible valencia (positiva, negativa o neutra). Entonces la clasificación se divide en dos etapas:

- Determinar si el mensaje es objetivo o subjetivo.
- Clasificar los mensajes subjetivos en positivos, negativos o neutros.

En cada etapa pueden ser utilizados los mismos clasificadores supervisados, pues ambas se enfrentan al mismo problema teórico. Con el objetivo de realizar una mejor exploración del espacio de búsqueda se analizaron algoritmos lineales y no lineales.
Los clasificadores no lineales propuestos para su evaluación fueron:

Dummy (uniform): clasifica aleatoriamente utilizando una distribución uniforme [25].

Dummy (stratified): clasifica cada ejemplo de forma aleatoria proporcional a la frecuencia relativa de cada clase [25].

Neural Network (1 layer): consiste en una red neuronal *feedfoward* con 1 capa oculta, de 10 neuronas. Esta arquitectura permite aproximar funciones de clasificación no lineales [13].

Rocchio: el clasificador de *Rocchio* se basa en aplicar el algoritmo de clasificación *k-nearest-neighboors* (KNN) utilizando como indicadores los centroides obtenidos del algoritmo *k-means* u otro método de agrupamiento, etiquetados según la clase mayoritaria de cada grupo, o por algún tipo de voto ponderado [17].

Nearest Neighbors (KNN): el funcionamiento del algoritmo KNN radica en asignar la clase a un nuevo ejemplo basado en las observaciones de las clases de sus vecinos más cercanos. Es necesario contar con una representación de los datos en un espacio métrico y con una definición de distancia que aproxime de cierta forma la topología del espacio original [6].

Naive Bayes: consiste en un clasificador bayesiano simple que asume independencia entre las características. La fase de entrenamiento consiste en computar 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[18].

RBF SVM: adiciona una función de *kernel* no lineal a las máquinas de soporte vectorial que proyecta los vectores a clasificar en un espacio de dimensión mayor. En este espacio los vectores se encuentran más esparcidos, por lo que se aumenta la probabilidad de encontrar un separador lineal. La proyección del separador lineal en el espacio original consiste en un separador no lineal cuya forma depende del *kernel* escogido. De esta forma es posible clasificar en espacios no separables linealmente [19].

Decision Tree: los árboles de decisión son clasificadores que aproximan una función a partir de la ejecución de un conjunto de pruebas sobre los valores asociados a atributos predefinidos. Se utiliza el algoritmo de aprendizaje ID3 para la construcción de árboles de decisión. Este algoritmo genera de forma iterativa un árbol, eligiendo en cada nodo el atributo que maximiza la cantidad de información obtenida sobre el conjunto de entrenamiento [28].

Random Forest: consiste en un conjunto *ensemble* de árboles de decisión. Cada árbol se construye de un ejemplo extraído con reemplazo del conjunto de entrenamiento. Al dividir un nodo, el corte realizado se escoge de forma óptima en un subconjunto aleatorio de las características. Como resultado generalmente aumenta ligeramente el sesgo del clasificador, disminuyendo en cambio la varianza. A menudo la disminución de la varianza compensa el aumento del sesgo, resultando en un clasificador más preciso [4].

En tanto los clasificadores lineales propuestos fueron:

Neural Network (linear): red neuronal de una sola capa, que solo permite aproximar funciones lineales [28].

Logistic Regression: también conocido en la literatura como clasificador de máxima entropía. Está basado en un modelo lineal que minimiza el costo de "hit or miss" de la función, en vez de la suma de las raíces de sus residuales, como una regresión ordinaria [19].

Lasso: consiste en un modelo lineal que estima coeficientes esparcidos. Muestra una tendencia a encontrar soluciones con una cantidad de parámetros reducida, disminuyendo la cantidad de variables de las que depende la hipótesis de clasificación [19].

Ridge: el clasificador se basa en un modelo lineal. Utiliza la regresión de Ridge sobre el problema de mínimos cuadrados, imponiendo una penalización cuadrática al tamaño de los coeficientes [19].

Perceptron: red neuronal para funciones lineales, sin capas ocultas, de la herramienta `sklearn` [25].

Passive-Aggressive: es familia de los algoritmos de aprendizaje a gran escala. Similar al Perceptron ya que no requiere un factor de aprendizaje, pero incluye la regularización de un parámetro [25].

Stochastic Gradient Descent (SGDC model): es un modelo lineal muy simple y eficiente. Particularmente utilizado con grandes conjuntos de características [25].

Linear SVM: las máquinas de soporte vectorial construyen una hipótesis mediante el cálculo de un hiperplano 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. En espacios muy esparcidos es bastante probable encontrar un hiperplano separador, o al menos minimice de forma considerable los elementos en la clase incorrecta [6].

LDA: modela la distribución condicional de los datos, $P(X|y = k)$, para cada clase k. La predicción puede obtenerse utilizando la regla de Bayes. La probabilidad $P(X|y)$ es modelada como una distribución Gausiana, asumiendo la misma matriz de covarianza para cada clase [19].

QDA: como en LDA pero no se asume ninguna restricción sobre las matrices de covarianza, lo que conlleva a una superficie de decisión cuadrática [19].

4. EVALUACIÓN Y RESULTADOS

La evaluación de los algoritmos de clasificación en el problema de minería de opinión en *Twitter*, según la metodología de minado propuesta, requiere de la implementación de los distintos algoritmos de clasificación así como un *corpus* de *tweets* con una clasificación asignada *a priori*.
Como implementación de los algoritmos de descomposición y *clustering* así como de clasificación fueron utilizados, en la mayoría de los casos, las implementaciones propuestas por la herramienta `sklearn` [25]. Solo no fue utilizada

`sklearn` en los casos de las implementaciones de Growing Neural Gas y las Neural Network (linear, 1 layer) para los cuales se realizaron implementaciones particulares.
Twitter no permite la difusión de *corpus* creados a partir de los datos de dicha red. Por ello fue necesario la elaboración de un *corpus* propio que permitiera entonces la evaluación de los algoritmos de clasificación. La colección de *tweets* fue construida a partir de 100000 mensajes de los usarios cubanos en *Twitter* en el período febrero-mayo de 2012. Posteriormente, los *tweets* fueron filtrados por idioma y, de estos, fueron seleccionados aleatoriamente el conjunto final de mensajes que conformó el *corpus*. Finalmente, estos *tweets* fueron clasificados a mano para establecer, *a priori*, sus categorías (objetivo-subjetivo,positivo-negativo-neutro). Así el corpus está compuesto por 1562 mensajes en idioma español, de ellos 899 son subjetivos y 663 son objetivos. Los subjetivos se dividen en 416 positivos, 248 negativos y 235 neutros.
De acuerdo con la metodología de minado de opinión definida y los distintos algoritmos de clasificación a evaluar, existen un total de 40460 combinaciones a valorar (dado por los 10 preprocesamientos, 14 algoritmos de reducción de dimensiones y los 17 algoritmos de clasificación a utilizar en las 2 etapas de clasificación). Este elevado número de combinaciones requiere de una gran cantidad de experimentos. Con el fin de reducir el volumen de procesamiento, fueron evaluadas inicialmente todas las combinaciones que permitieran la clasificación de los *tweets* en objetivo-subjetivo. Esta primera evaluación permitiría determinar los algoritmos más promisorios, en todas las etapas del minado de opinión, para realizar el proceso completo.

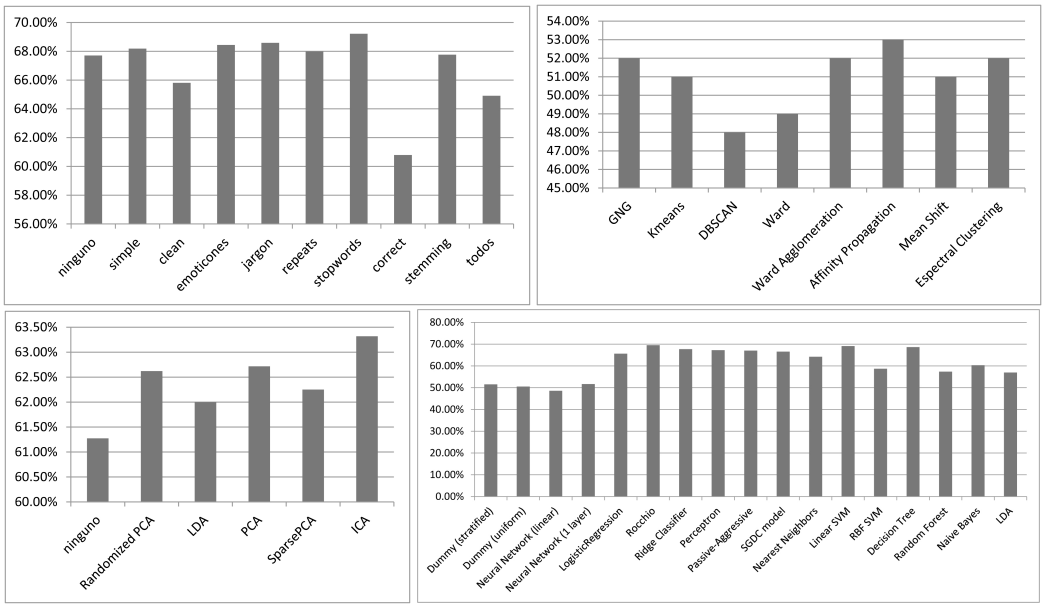

Figura 2: Precisión de elementos en la primera evaluación: superior-izquierda algoritmos de preprocesamiento, superior-derecha algoritmos de *clustering*, inferior-izquierda algoritmos de descomposición, inferior-derecha algoritmos de clasificación

En la primera evaluación (clasificación objetivo-subjetivo) se analizaron 2380 combinaciones. Para cada una de estas combinaciones se evaluó la precisión de la clasificación obtenida utilizando validación cruzada con un $60\,\%$ de los datos para entrenar y un $40\,\%$ para validar. Cada posible preprocesamiento o algoritmo de reducción o clasificación

interviene solo en algunas de las combinaciones. Así un algoritmo de preprocesamiento interviene en 238 combinaciones, uno de reducción de dimensiones en 170 y uno de clasificación en 140. Entonces, asumiendo independencia entre cada etapa hasta la clasificación objetivo-subjetivo, se obtuvo como medida de evaluación de cada elemento el promedio de la precisión de la clasificación de cada combinación en las que ellos intervienen (figura 2).
Con los resultados de esta primera evaluación se observó que los algoritmos de preprocesamiento ofrecen resultados similares, con excepción de la corrección ortográfica que muestra peores resultados. Esto se debe a que el corrector ortográfico elige a menudo palabras incorrectas y modifica el sentido del mensaje.
En los algoritmos de descomposición los mejores resultados se obtienen con PCA e ICA. Los algoritmos de *clustering* resultan mucho más susceptibles a la parametrización que los algoritmos de descomposición. En la clasificación los algoritmos con resultados más notables fueron: Rocchio, Linear SVM, Decision Tree y en un segundo lugar: Ridge, Perceptron Linear y Passive-Aggressive. Los clasificadores lineales, debido a la naturaleza esparcida de los datos, obtienen muy buenos resultados (figura 2), al ser más simples y menos sensibles al ajuste de parámetros.
Entonces en una segunda evaluación se valoraron las combinaciones que incluyen a los algoritmos que mejores resultados mostraron en la primera experimentación. De la misma forma que en el experimento anterior la calidad se determinó por la precisión de la clasificación utilizando validación cruzada.
La tabla 2 muestra los resultados en la clasificación objetivo-subetivo mientras que la tabla 3 muestra los resultados en la clasificación positivo-negativo-neutro. Los preprocesamientos empleados fueron *simple* y *clean*, pues son más eficientes y muestran resultados similares al resto.

	Ridge	Perceptron	Rocchio	SMV_L	Pas-Agg	DTree
Simple-PCA	**71**|1.41	65|3.16	**71**|1.41	**71**|1.73	63|4.47	66|2.45
Simple-ICA	**70**|2.00	64|5.48	**71**|1.41	57|1.41	67|1.41	65|2.83
Clean-PCA	**72**|1.73	65|4.47	**71**|1.41	**71**|1.41	63|6.32	67|2.65
Clean-ICA	**71**|1.73	64|7.07	**71**|1.41	57|1.41	68|3.16	65|2.45

Cuadro 2: Precisión|Desviación estándar de cada clasificador, según el preprocesamiento y descomposición aplicada, clasificando en Objetivo o Subjetivo.

	Ridge	Perceptron	Rocchio	SMV_L	Pas-Agg	DTree
Simple-PCA	**49**|2.24	45|3.16	**47**|2.65	**47**|2.65	43|4.47	42|3.16
Simple-ICA	**49**|1.73	42|6.32	**47**|2.45	46|2.24	**47**|3.16	45|3.16
Clean-PCA	**49**|2.45	43|2.45	**49**|3.16	**47**|2.24	44|3.16	43|2.83
Clean-ICA	**49**|2.00	43|5.48	46|3.00	**47**|1.73	45|3.16	44|3.16

Cuadro 3: Precisión|Desviación estándar de cada clasificador, según el preprocesamiento y la descomposición aplicada, clasificando en Positivo, Negativo o Neutro.

Como resultado de esta evaluación se pudo observar que para la clasificación objetivo-subjetivo los algoritmos Ridge y Rochio son los que mejores resultados ofrecen. Por otro lado, para la clasificación positivo-negativo-neutro los algoritmos que mejor precisión ofrecen son Ridge, Rochio y SVM_L. En el caso de la reducción de dimensiones PCA sobresale por sobre ICA que afecta el funcionamiento de algunos clasificadores.
Finalmente, con estos algoritmos que ofrecieron la mejor presición en los experimentos previos (tabla 2 y 3) se evaluó integramente el proceso de minado de opinión en *Twitter*. En este caso se utilizó el procesamiento clean que fue el de mejor precisión. Los resultados se muestran en la tabla 4.

OS \ PNN	Rocchio	Ridge	SVM_L
Rocchio	47|2.1	**51**|1.7	47|2.1
Ridge	48|1.9	**51**|1.2	49|2.0

Cuadro 4: Precisión|Desviación estándar de cada combinación de clasificadores, para la clasificación general.

En esta evaluación final la combinación que reporta mejores resultados en la clasificación es Ridge-Ridge con un 51 % de precisión clasificando en 4 clases. Este resultado duplica la efectividad de la clasificación aleatoria.

5. CONCLUSIONES

Los distintos experimentos realizados con los diferentes preprocesamientos propuestos muestran que el proceso de normalización del texto no influye significativamente en la clasificación. Aunque es necesario realizar un proceso básico que al menos elimine los signos de puntuación y separe correctamente las palabras.
La reducción de dimensiones con algoritmos de descomposición superó, de forma general, a la realizada con algoritmos de *clustering*. Esto es debido a que la identificación de *clusters* es muy dependiente de los parámetros, requiere de grandes volúmenes de información y nociones de distancia que utilicen información semántica.
Los resultados obtenidos con clasificadores lineales presentan una efectividad superior al resto de los algoritmos analizados. Como la clasificación ocurre en un espacio de grandes dimensiones, donde los datos están muy esparcidos, es relativamente sencillo encontrar un clasificador lineal con resultados aceptables. En tanto, los clasificadores no lineales, al ser más complejos, dependen mayor grado de la configuración de sus parámetros provocando muchas veces que, con el mismo conjunto de entrenamiento, sean superados por los clasificadores lineales.
Finalmente, los resultados alcanzados hablan de la necesidad de utilizar los algoritmos de clasificación supervisada complementados con información semántica o en un enfoque mixto con otras técnicas de minería de opinión para lograr una mayor efectividad en el minado de opinión en *Twitter*.

RECEIVED SEPTEMBER, 2014
REVISED MAY, 2015

REFERENCIAS

[1] ARREDONDO, N. P. A. [2009]: Método Semisupervisado para la Clasificación Automática de Textos de Opinión Master's thesis, Instituto Nacional de Astrofísica, Óptica y Electrónica, Tonantzintla, Puebla.

[2] BIRANT, D. and ALP, K. [2007]: ST-DBSCAN: An algorithm for clustering spatial–temporal data **Data & Knowledge Engineering**, 60(1):208–221.

[3] BOLLEN, J., ALBERTO, P., and HUINA, M. [2009]: Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena **arXiv preprint arXiv:0911.1583**.

[4] BREIMAN, L. [2001]: Random forests **Machine learning**, 45(1):5–32.

[5] COMANICIU, D. and PETER, M. [2002]: Mean shift: A robust approach toward feature space analysis **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, 24(5):603–619.

[6] DUDA, R., P.E., H., and D.G., S. [2001]: **Pattern classification** Pattern Classification and Scene Analysis: Pattern Classification. Wiley, New York.

[7] ERASO, H. A. O. and LOZADA, C. A. C. [2011]: Stemming en Español para Documentos Recuperados de la Web **Revista Unimar**, (58):107–114.

[8] ESULI, A. and FABRIZIO, S. [2006]: Determining Term Subjectivity and Term Orientation for Opinion Mining **Proceedings of EACL-06, 11th Conference of the European Chapter of the Association for Computational Linguistics**, pages 193–200.

[9] FIGUEROLA, C. G., RAQUEL, G., and LÓPEZ, D. S. R. E. [2000]: Stemming and n-grams in spanish: an evaluation of their impact on information retrieval **Journal of Information Science**, 26(6):461–467.

[10] FRAKES, W. B. and RICARDO, B.-Y., editors [1992]: **Information Retrieval: Data Structures & Algorithms** Prentice Hall, New Jersey.

[11] FRITZKE, B. [1995]: A growing neural gas network learns topologies **Advances in neural information processing systems**, 7:625–632.

[12] GO, A., RICHA, B., and LEI, H. [2009]: Twitter sentiment classification using distant supervision **CS224N Project Report, Stanford**, 1:12.

[13] HORNIK, K., MAXWELL, S., and HALBERT, W. [1989]: Multilayer feedforward networks are universal approximators **Neural networks**, 2(5):359–366.

[14] HU, X., LEI, T., JILIANG, T., and HUAN, L. [2013]: Exploiting Social Relations for Sentiment Analysis in Microblogging In **Proceedings of the Sixth ACM International Conference on Web Search and Data Mining**, WSDM '13, pages 537–546, New York, NY, USA. ACM.

[15] KANUNGO, T., M, M. D., S, N. N., D, P. C., RUTH, S., and Y, W. A. [2002]: An efficient k-means clustering algorithm: Analysis and implementation **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, 24(7):881–892.

[16] KONTOPOULOS, E., CHRISTOS, B., THEOLOGOS, D., and NICK, B. [2013]: Ontology-based sentiment analysis of twitter posts **Expert Systems with Applications**, 40:4065–4074.

[17] MANNING, C. D., PRABHAKAR, R., and HINRICH, S. [2008]: **Introduction to information retrieval**, volume 1 Cambridge University Press, Cambridge, UK.

[18] MITCHELL, T. [1997]: **Machine Learning** McGraw-Hill international editions - computer science series. McGraw-Hill Education, New York.

[19] MURPHY, K. [2012]: **Machine Learning: A Probabilistic Perspective** MIT Press, Massachusetts.

[20] NG, A. Y., I, J. M., and YAIR, W. [2002]: On spectral clustering: Analysis and an algorithm **Advances in neural information processing systems**, 2:849–856.

[21] PAK, A. and PATRICK, P. [2010]: Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In **LREC**, volume 10, pages 1320–1326.

[22] PANG, B. and LILLIAN, L. [2004]: A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts In **Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics**, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

[23] PANG, B. and LILLIAN, L. [2008]: Opinion mining and sentiment analysis **Foundations and trends in information retrieval**, 2(1-2):1–135.

[24] PANG, B., LILLIAN, L., and SHIVAKUMAR, V. [2002]: Thumbs Up?: Sentiment Classification Using Machine Learning Techniques In **Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10**, EMNLP '02, pages 79–86, Stroudsburg, PA, USA. Association for Computational Linguistics.

[25] PEDREGOSA, F., G., V., A., G., V., M., B., T., O., G., M., B., P., P., R., W., V., D., J., V., A., P., D., C., M., B., M., P., and E., D. [2011]: Scikit-learn: Machine Learning in Python **Journal of Machine Learning Research**, 12:2825–2830.

[26] RILOFF, E. and JANYCE, W. [2003]: Learning Extraction Patterns for Subjective Expressions In **Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing**, EMNLP '03, pages 105–112, Stroudsburg, PA, USA. Association for Computational Linguistics.

[27] RÖHLER, A. B. [2007]: Generación automática de recursos lingüísticos para la minería de opiniones. Master's thesis, Universidad de la Habana.

[28] RUSSELL, S., P.A., N., and R.B., G. [1996]: **Inteligencia Artificial: Un Enfoque Moderno** Colección de Inteligencia Artificial. Prentice Hall Hispanoamericana, S.A., Ciudad de México.

[29] TURNEY, P. D. [2002]: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews **Proceedings of the 40th Annual Meet- ing of the Association for Computational Linguistics (ACL)**, pages 417–424.

[30] VINODHINI, G. and RM., C. [2012]: Sentiment Analysis and Opinion Mining: A Survey **International Journal of Advanced Research in Computer Science and Software Engineering**, 2:281–292.

[31] WANG, K., JUNYING, Z., DAN, L., XINNA, Z., and TAO, G. [2007]: Adaptive Affinity Propagation Clustering **Acta Automatica Sinica**, 33(12):1242–1246.

[32] WARD JR, J. H. [1963]: Hierarchical grouping to optimize an objective function **Journal of the American Statistical Association**, 58(301):236–244.

[33] WIEBE, J. and ELLEN, R. [2005]: Creating Subjective and Objective Sentence Classifiers from Unannotated Texts **Proceedings of the Sixth International Confer- ence on Intelligent Texto Processing and Computational Linguistics 2005**, pages 486–497.

[34] YU, H. and VASILEIOS, H. [2003]: Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences In **Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing**, EMNLP '03, pages 129–136, Stroudsburg, PA, USA. Association for Computational Linguistics.

[35] ZHANG, L., RIDDHIMAN, G., MOHAMED, D., MEICHUN, H., and BING, L. [2011]: Combining lexicon-based and learning-based methods for twitter sentiment analysis **HP Laboratories, Technical Report HPL-2011**, 89.