ISSN 1870-4069

Clasificación automática de opiniones de películas usando aprendizaje automático

Jesús Andres Sierra-Rangel, Rafael Guzmán-Cabrera

Universidad de Guanajuato,
Departamento de estudios multidisciplinarios,
México

{ja.sierrarangel, guzmanc}@ugto.mx

Abstract. El emitir opiniones sobre productos o servicios en foros y redes sociales se ha convertido en una tarea cotidiana. Cuando el número de opiniones es muy grande se requiere contar con herramientas automáticas que realicen esta tarea eficientemente. En este trabajo se presentan resultados obtenidos al llevar a cabo la clasificación automática de opiniones en distintos escenarios de clasificación y con diferentes conjuntos del corpus "Large movie review dataset". La idea principal de la configuración experimental propuesta fue mostrar la cantidad de opiniones que se requiere como mínimo para que el sistema de aprendizaje automático pueda aprender las características que distinguen una clase de otra. Los resultados obtenidos nos muestran que dependiendo del método de aprendizaje existe un sobre aprendizaje para métodos como SVM, KNN y Random Forest, mientras que para métodos como Naïve Bayes los mejores resultados son los que deben utilizar un 70-80 de los atributos y solo en Logistic regression debe usarse en crudo para tener los mejores resultados.

Keywords: Gran conjunto de datos de reseñas de películas, SVM, Naïve Bayes, bosque aleatorio, KNN, regresión logística.

Automatic Movie Review Classification Using Machine Learning

Abstract. Expressing opinions about products or services in forums and social networks has become an everyday task. When the number of opinions is very large, it is necessary to have automatic tools that perform this task efficiently. This work presents results obtained by carrying out the automatic classification of opinions in different classification scenarios and with different sets of the "Large movie review dataset" corpus. The main idea of the proposed experimental setup was to show the minimum number of opinions that is required so that the machine learning system can learn the characteristics that distinguish one class from another. The results obtained show us that depending on the learning method there is overlearning for methods such as SVM, KNN and Random Forest, while for methods such as Naive Bayes the best results are those that must use 70-80 of the attributes and only in Logistic regression should be used raw to have the best results.

Keywords: Large movie review dataset, SVM, Naïve Bayes, random forest, KNN, logistic regression.

1. Introducción

Las tecnologías han evolucionado y se ha encontrado que la gran mayoría de información se encuentra en forma no estructura [1] se estima que entre el 80% y el 90% de los datos de las organizaciones son no estructurados, para obtener esa información es necesario procesarla y tener una estructura que se pueda analizar, para esto tenemos la minería de datos, dentro de la minería de datos, encontramos una rama llamada "text mining" o la minería de texto que [2, 3] es el descubrimiento por computadora de información nueva, previamente desconocida, mediante la extracción automática de información, en esta rama se trabaja con el análisis de sentimientos, se define [4] como minería de opinión o "opinion mining", se trata de una tarea de clasificación masiva de documentos de manera automática, que se centra, entre otras cosas, en catalogar los documentos en función de la connotación positiva o negativa del lenguaje utilizado en el mismo, [5] el enfoque común para comprender la opinión es verlo desde ambas perspectivas, es decir, positiva y negativa y el proceso se llama análisis de sentimiento [6].

Las opiniones clasificadas se han identificado como tarea primordial del sentimiento. Tradicionalmente, la clasificación de textos se realizaba principalmente mediante tareas de ingeniería manual, como el desarrollo de características artesanales consultando diccionarios, técnicas basadas en el conocimiento o componentes jerárquicos personalizados.

1.1. Corpus "Large Movie Review Dataset"

De acuerdo con [7], dentro del corpus podemos encontrar 50k de opiniones, de las cuales se tiene 25k para entrenamiento y otras 25k de test en la categoría de positiva y negativas, este corpus esta balanceado por lo que tendremos en total 25k de positivas y 25k negativas.

Dentro de los investigadores que han utilizado este corpus, buscan la clasificación a través de diferentes métodos para la selección de características basada en diferentes índices, [8] como el Gini con un clasificador de máquina de vectores de soporte (SVM) para la clasificación de sentimientos, o por medio [9] de características hibridas que se obtiene al concatenar características de aprendizaje automático (TF, TF-IDF) con características de léxico (recuento de palabras positivo-negativo, connotación) con el que obtienen mejores resultados tanto en términos de precisión como de complejidad cuando se prueba, también utilizan [10] modelos híbrido CNN_LSTM que han superado a las redes MLP y singular CNN y LSTM. CNN_LSTM se han informado una precisión del 89,2 %, mientras que CNN ha proporcionado una precisión del 87,7 %, mientras que MLP y LSTM han informado una precisión del 86,74 % y 86,64.

En otros casos usan algoritmos como [11] Word2Vec de Google para la clasificación de texto de manera que se conserve las asociaciones semánticas entre los términos de las palabras para aprender las funciones de aprendizaje del algoritmo, o métodos [12] para abordar el problema de la clasificación de sentimientos a través de arquitecturas de redes neuronales recurrentes, además de una red neuronal recursiva para el análisis a nivel de oración y una red neuronal recurrente.

Muchas veces estos experimentos no los podemos realizar debido al hardware que poseemos y es necesario una reducción del corpus, pero qué tanto es lo necesario de

reducir para obtener los resultados que nos puedan brindar información de características similares o iguales a los que usan el corpus entero.

2. Metodología

En este trabajo se realiza la clasificación del corpus “large movie review dataset”, del cual se utilizaron 5000 instancias balanceadas, los cuales se encuentran etiquetados encuanto a la polaridad de la opinión como: positiva o negativa, siendo así un 2500 positivas y 2500 negativas, este trabajo de etiquetación fue realizado por los investigadores Andrew L. Maas y su equipo, en 2011.

En nuestro caso utilizamos de clasificación “random sampling”, se entrenó en base a 5clasificadores Knn, Logistic Regression, Naïve Bayes, SVM, Random Forest, dentro denuestra metodología se encuentran dos caminos, el primero que incluye preprocesado y un segundo camino que se considera nuestro “baseline” , que son datos en crudo sin realizar preprocesamiento, mientras que en el primero se realiza un preprocesamiento que consiste en eliminar las palabras de paro, también llamadas “stop words”, números y acentos, las cuales se consideran palabras vacíos.

También se realiza la transformación de las palabras en “lowcase” traducido en español palabras minúsculas,las cuales nos produce reducir las diferencias de palabras por alguna mayúscula, creanuna cantidad menor de dimensionalidad. Seguido de esto, dentro del preprocesamiento marcamos un número máximo de tokensque definirá el tamaño de nuestra matriz para obtener los mejores resultados, en este proceso se llegó a utilizar el 2.7%-80% de las palabras de mayor relevancia para comparar y saber qué cantidad de palabras son necesarias para obtener los mejores resultados.

En ambos casos se utiliza una bolsa de palabras en donde se selecciona como frecuencia del documento el IDF (Inverse document frecuency) que indica la relevancia de las palabras, esto entra al sistema y a partir de los clasificadores nuestro sistema los utiliza de aprendizaje para poder obtener las 3 métricas que buscamos obtener las cuales son F1, Recall y Precision.

3. Estado del arte

En esta sección describimos las técnicas más utilizadas de clasificación de texto y métricas que pueden aplicarse para mejorar los resultados de clasificación.

3.1. Métodos de clasificación utilizados como métodos de aprendizaje

Basado en [13]: Naïve Bayes es un método de clasificación supervisado y generativo que se basa en el teorema de Bayes y en la premisa de independencia de los atributos para obtener la probabilidad de que un documento pertenezca a una determinada clase. Support Vector Machine es un método supervisado de clasificación binaria en el cual el entrenamiento consiste en encontrar un hiperplano que separe los vectores de atributos que representan los documentos del conjunto de datos en dos grupos, siendo esta separación la más grande posible.

Tabla 1. Resultados con KNN.

	F1	**Precisión**	**Recall**
Baseline	0.596	0.596	0.596
2.70%	0.626	0.631	0.628
10%	**0.618**	0.636	**0.626**
20%	0.598	**0.642**	0.616

Tabla 2. Resultados con SVM.

	F1	**Precisión**	**Recall**
Baseline	0.397	0.618	0.521
10%	0.405	**0.624**	0.524
20%	**0.411**	0.617	**0.526**
30%	0.405	0.622	0.524

Aquellos vectores que definen los márgenes de la máxima separación entre las clases se conocen como support vectors.

Logistic Regression Se define en [14] como un algoritmo de aprendizaje automático de clasificación utilizado para predecir la probabilidad y datos mediante rectas, que requieren que la variable dependiente sea binaria.

Knn (K-Nearest Neighbor): basado en [15] KNN se une al algoritmo clasificador que categoriza nuevos elementos en un conjunto de datos de prueba. Detecta los vecinos más cercanos de cada elemento en un conjunto de datos de entrenamiento a través de una medida de similitud, expresada por una función de distancia (es decir, euclidiana, Manhattan, Minkowski).

Basado en [16] Random forest, el algoritmo del bosque del árbol de decisión se entrena en múltiples árboles de decisión impulsados por subconjuntos de datos ligeramente diferentes. El bosque aleatorio es parte de los métodos de conjuntos familiares que toman el árbol de decisión como un predictor individual, son basado en los métodos deembolsado, aleatorización de salidas y subespacio aleatorio que excusan el impulso.

3.2. Métricas de evaluación

Precisión: Basado en [4] La precisión se define como la relación entre las observaciones predichas correctamente y el número total de observaciones:

$$Precisión = \frac{tp}{tp+tp}, \tag{1}$$

donde tp equivale a un valor verdadero-positivo y fp a un valor falso-positivo. Recall: [4] La recuperación se define como la proporción de observaciones que se predicen positivamente correctas con respecto al número total de observaciones en unaclase real:

$$Recall = \frac{tp}{tp+tf}, \tag{2}$$

donde tp representa un valor verdadero-positivo y fn representa un valor falso-negativo.

F1: [14, 4] es una medida de precisión en una prueba que se calcula a partir de la precisión y el recall de la prueba que se está llevando a cabo, en pocas palabras F1 es la media armónica de la precisión y el recall:

Tabla 3. Resultados con Random Forest.

	F1	**Precisión**	**Recall**
Baseline	0.737	0.737	0.737
40%	0.784	0.785	0.784
50%	**0.793**	**0.795**	**0.793**
60%	0.784	0.786	0.785

Tabla 4. Resultados con Logistic Regression.

	F1	**Precisión**	**Recall**
Baseline	0.737	0.737	0.737
20%	0.784	0.785	0.784
30%	**0.793**	**0.795**	**0.793**
40%	0.784	0.786	0.785

$$F1 = \frac{tp}{tp+\frac{1}{2}(tp+fn)}, \tag{3}$$

donde tp es un valor verdadero-positivo, fp un valor falso-positivo y fn un valor falso-negativo.

3.3. Muestreo

Basado en [17] Muestreo aleatorio: en este caso se utilizan dos argumentos para definirel muestreo, el primero es el porcentaje que se va a tomar del total de los datos para usar como datos de prueba, el segundo es la cantidad de iteraciones que se van a realizarpara entrenar al algoritmo.

4. Resultados

Se puede observar que KNN aumenta las métricas comparado si se utilizara en crudo, también podemos ver que a partir del 20% de los atributos hay disminución de los resultados.

Se puede mencionar que en la precisión en SVM es superior que F1 y Recall,siendo estas muy inferiores para los análisis y que también se puedeobservar una disminución en las métricas a partir del 30%.

En Random Forest se puede observar que comparado con SVM y KNN es superior por aproximadamente .15, y superior a usarlo que, en base cruda, una desventaja que podemos notar es utilizar el 50% de los atributos lo cualnos lleva más tiempo. En Logistic Regression es el que tiene mejores resultados de los 5 métodosde aprendizaje, pero se debe utilizar en baseline.

Naïve Bayes es muy cercano a Logistic Regression con la ventaja de que dura un poco menos en procesar la información, pero no con una gran diferencia de tiempo si lo comparamos con SVM, KNN o Logistic Regression.

Se puede mostrar que la diferencia más grande que se puede observar de losmejores resultados de este articulo y los referenciados es de 0.023, en comparación de que en estos trabajos utilizan todo el corpus comparado contra la utilización solo del 5k de

Tabla 5. Resultados con Naïve Bayes.

	F1	**Precisión**	**Recall**
Baseline	0.857	0.862	0.858
60%	0.85	0.855	0.85
70%	**0.862**	**0.862**	**0.862**
80%	**0.862**	**0.862**	**0.862**

Tabla 6. Resultados comparados con otros trabajos.

Trabajos relacionados [10]	Precisión	**Resultados**	Precisión
MLP	0.8674	SVM-10%	0.624
LSTM	0.8664	KNN-20%	0.642
CNN_LSTM	**0.892**	RF- 50%	0.795
CNN	0.877	LR-Baseline	**0.869**
		NB-70%	0.862

instancias y un escenario de atributos especifico reduciendo el tiempo de nuestros ensayos.

5. Discusión

Este trabajo demuestra que podemos encontrar resultados similares si usáramos menos atributos, lo que nos da menos tiempo de procesamiento y que otros equipos con menor capacidad de hardware puedan encontrar resultados buenos, pero hace falta revisar con más configuraciones en el experimento para poder encontrar resultados que puedan superar el margen del 90% en la precisión.

6. Conclusión

La conclusión a la que se llegó con esta investigación es en base a los resultados obtenidos, existe un "sobre aprendizaje" a la hora de agregar más cantidad de atributos en los métodos de aprendizaje como SVM, KNN y Random Forest, ya que como se puede observar en los resultados cuando se utilizan más cantidad de atributos, éstos obtienen menor rendimientos al compararlo con menos palabras, por lo que es significativo no superar en estos métodos de aprendizaje más del 20%, salvo para "Random Forest" es recomendable usar la mitad de los atributos para su mejor rendimiento, en caso de Naïve Bayes se recomienda usar el 70%-80% de los atributos, mientras que para "Logistic regression" es recomendable utilizarlo con datos en crudo.

Para el futuro, consideramos también la posibilidad de utilizar para los algoritmos tradicionales otros tipos de atributos como n-gramas de varios tipos y tamaños (de palabras, de caracteres, de pos tags, mixtos, sintácticos), mismo que utilizar los transformes en el caso de aprendizaje profundo [18].

References

1. Barrera, M. C.: Text mining: a current view. Biblioteca Universitaria; vol. 17, no. 2, pp. 129–138 (2014)
2. Gupta, V., Lehal, G. S.: A survey of text mining techniques and applications. Journal of Emerging Technologies in Web Intelligence, vol. 1, no. 1, pp. 60–76 (2009)
3. Shah, K., Patel, H., Sanghvi, D., Shah, M.: A comparative analysis of logistic regression, random forest and KNN models for the text classification. Augmented Human Research, vol. 5, no. 1, pp. 1–16 2020). doi: 10.1007/S41133-020-00032-0
4. Sánchez del Hoyo, R.: Análisis de sentimientos con Twitter: turismo y política electoral. Depósito de Investigación Universidad de Sevilla, pp. 1–84 (2019)
5. Bhatia, S., Chaudhary, P., Dey, N.: Opinion mining in information retrieval. Springer Singapore (2020) doi: 10.1007/978-981-15-5043-0
6. Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 142–150 (2011)
7. Manek, A. S., Shenoy, P. D., Mohan, M. C.: Aspect term extraction for sentiment analysis in large movie reviews using Gini index feature selection method and SVM classifier. World Wide Web, vol. 20, no. 2, pp. 135–154 (2017) doi: 10.10 07/s11280-015-0381-x
8. Harish, B. S., Kumar, K., Darshan, H. K.: Sentiment analysis on IMDb movie reviews using hybrid feature extraction method. International Journal of Interactive Multimedia and Artificial Intelligence, vol. 5, no. 5, p. 109 (2019) doi: 10.9781/IJIMAI.2018.12.005
9. Nehal, M. A., Marwa, A. E., Aliaa, Y.: sentiment analysis for movies reviewsdataset using deep learning models. vol. 9, no. 2/3, pp. 19–27 (2019) doi: 10.5121/ ijdkp.2019.9302
10. Timmaraju, A., Khanna, V.: Sentiment analysis on movie reviews using recursive and recurrent neural network architectures. Semantic Scholar, pp. 1–5 (2015)
11. Chakraborty, K., Bhattacharyya, S., Bag, R., Hassanien, A. E.: Comparative sentiment analysis on a set of movie reviews using deep learning approach. In: The International Conference on Advanced Machine Learning Technologies and Applications, AMLTA'18, Springer International Publishing, vol. 723, pp. 311–318 (2018). doi: 10.1007/978-3-319-74690- 6_31
12. Dubiau, L., Ale, J. M.: Análisis de sentimientos sobre un corpus en español: Experimentación con un caso de estudio. In: XIV Argentine Symposium on Artificial Intelligence (ASAI)-JAIIO (2013
13. Morales-Castro, J. C., Ledesma-Carrillo, L. M., Guzman-Cabrera, R.: Identificación de polaridad en twitter usando validación cruzada. Identidad Energetica, vol. 4 (2021)
14. https://www.researchgate.net/publication/357860394_Identificacion_de_polaridad_en_Twitter_usando_validacion_cruzada
15. Chatzigeorgakidis, G., Karagiorgou, S., Athanasiou, S., Skiadopoulos, S.: FML-kNN: Scalable machine learning on big data using k-nearest neighbor joins. Journal of Big Data, vol. 5, no. 1, pp. 1–27 (2018) doi: 10.1186/S40537-0180115-x
16. Al-Amrani, Y., Lazaar, M., El-Kadiri, K. E.: Random forest and support vector machine-based hybrid approach to sentiment analysis. Procedia Computer Science, vol. 127, pp. 511–520 (2018) doi: 10.1016/J.PROCS.2018.01.150
17. Pauli, P. A.: Análisis de sentimiento: comparación de algoritmos predictivos y métodos utilizando un lexicón español (2019)
18. Sidorov, G.: Syntactic n-grams in computational linguistics. Springer (2019)