ISSN 1870-4069

Clasificación automática de sentimientos en textos de canciones en idioma español

Omar García-Vázquez, Tania Alcántara, Grigori Sidorov, Hiram Calvo

Instituto Politécnico Nacional, Centro de Investigación en Computación, México

omar.gava@hotmail.com, {talcantaram2020, sidorov, hcalvo}@cic.ipn.mx

Resumen. Los sentimientos son el estado afectivo de ánimo, los cuales son producidos en el cerebro y son provocados por una emoción. Este sentimiento se ha trasladado de múltiples maneras, como textos, pinturas o música. La música transmite diferentes emociones, lo cual hace aún más importante saber que tipo de sentimientos se encuentra dentro de una canción, que pueden ser, entre muchos otros, positivos, negativos o neutrales. A través de este trabajo se analizó y clasificó los textos de canciones del idioma español, esto a través de un extractor de características basado en co-ocurrencias y la aplicación de modelos más modernos como las Redes Neuronales Convoluciones y una *Long Short-Term Memory*, donde se obtuvieron resultados competitivos en el estado del arte.

Palabras clave: Procesamiento de lenguaje natural, clasificación de textos, songs, CNN, LSTM.

Automatic Sentiment Classification in Spanish Language Song Texts

Abstract. Feelings are the affective state of mind, which are produced in the brain and are caused by an emotion. This feeling has been transfered in multiple ways, such as texts, paintings or music. Music transmits different emotions, which makes it even more important to know what kind of feelings are found within a song, which can be, among many others, positive, negative or neutral. Through this work, the texts of songs in the Spanish language were analyzed and classified, this through a feature extractor based on co-occurrences and the application of modern models such as Convolutional Neural Networks and a *Long Short-Term Memory*, where competitive results were obtained in the state of the art.

Keywords: Machine learning, classification, natural language processing, songs, CNN, LSTM.

1. Introducción

La opinión, los sentimientos y todos los conceptos que los rodean; como el afecto, los estados de ánimo y la actitud siempre se han basado en las creencias de cada

persona. Estos aspectos de la individualidad humana hacen que siempre las acciones que realizamos estén influenciadas por otras.

La inserción y el rápido crecimiento del análisis de sentimiento coincide con lo mostrado en redes sociales, foros de discusión y blogs, ya que, por primera vez en la historia humana, se tiene un gran volumen de datos de opinión en medios digitales [1]. Los datos recolectados a través de internet a menudo presentan opiniones, y es por eso que el análisis de sentimientos se ha convertido en una de las principales herramientas empleadas en el análisis de redes sociales.

El análisis de sentimientos es un campo de investigación dentro del Procesamiento de Lenguaje Natural (PLN o NLP por sus siglas del inglés *Natural Language Processing*), el cual utiliza técnicas de Aprendizaje Máquina (AP o ML por sus siglas del inglés *Machine Learning*).

Por otro lado, la búsqueda de emociones en las oraciones no es tan sencilla, ya que suelen ser oraciones subjetivas, las cuales enuncian hechos, porque las opiniones y los sentimientos son inherentemente subjetivos. Hoy en día, estos sentimientos han sido plasmados de diferente manera, desde libros, poemas y hasta en música.

La música es capaz de activar áreas emocionales e inclusive evocar recuerdos, a través de las partituras rítmicas. Pero el ritmo no es el único conducto de emociones, sino los textos y aquellas palabras que utilizan los autores para expresar tristeza, felicidad o emoción. La clasificación de este tipo de emociones puede realizarse a través de PLN.

Durante este trabajo se exploró el AS en textos de canciones, trabajando con un conjunto de datos en español. Esto por medio de la extracción de características con *embeddings*, combinado con clasificadores poco usuales en textos como las Convolutional Neural Network y las Long Short-Term Memory.

2. Marco teórico

2.1. Análisis de sentimientos

EL AS es una técnica de PLN, que se enfoca en identificar y extraer la emoción representada en un texto, como positiva, negativa o neutral [2]. La manera de abordar el SA tiene varios enfoques, desde los basados en reglas hasta el aprendizaje profundo.

De acuerdo con la página *QuestionPro*[1]: "El análisis de sentimiento utiliza tecnologías avanzadas de inteligencia artificial, como PLN, análisis de texto y ciencia de datos, para identificar, extraer y estudiar información subjetiva. En términos más simples, clasifica un texto como positivo, negativo o neutral". Para determinar esa polaridad, se puede hacer de técnicas de aprendizaje automático, aprendizaje profundo o del análisis semántico [3].

[1] Análisis de sentimiento. ¿Qué es y cómo realizarlo? https://www.questionpro.com/blog/es/herramienta-de-analisis-de-sentimientos/

Tabla 1. Distribución del corpus *Textos de canciones en español* [13].

Sentimiento	No. de oraciones	Porcentaje
S	97	6.67 %
P	780	52.80 %
N	600	40.53

2.2. Aprendizaje profundo

Las Redes Neuronales Profundas o mejor conocido como Deep Learning (DL) son una rama de la Inteligencia Artificial (IA o AI por sus siglas en inglés *Artificial Intelligence*).

La principal diferencia entre una red de DL y las Redes Neuronales (RN o NN por sus siglás en inglés Neural Network) clásicas, radica en la complejidad de la arquitectura [4]. Se puede describir de manera sencilla la estructura de una red neuronal profunda [4]:

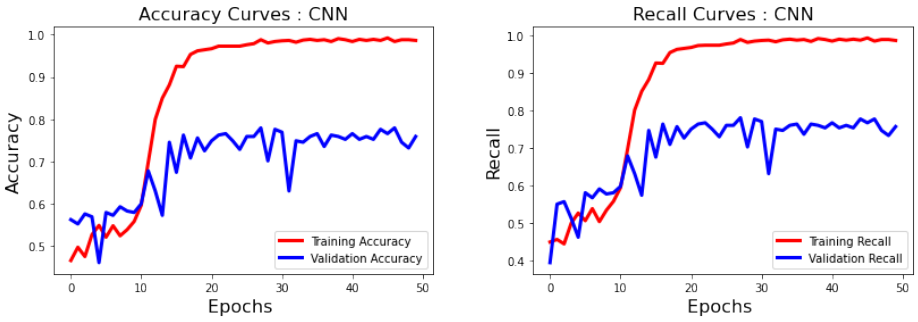
1. **Capa de entrada:** Son las neuronas que representan los datos de entrada.
2. **Capas ocultas:** La red neuronal profunda contendrá al menos una capa, los parámetros mínimos son el número de neuronas, la función de activación y la dimensión de los datos de entrada.
3. **Capa de salida:** Es la capa que da la respuesta codificada, se tendrán tantas salidas como entradas y se interpretara cada una como la probabilidad de que el dato de parámetros mínimos son el número de neuronas.

Existen varios tipos de DL, pero, este trabajo se centra principalmente en las siguientes [5]:

1. **Redes neuronales convolucionales (CNN, por sus siglas en inglés *Convolutional Neural Network*):** Constan de una o varias capas llamadas "convoluciones", en donde se aplican filtros a la entrada para extraer las principales características. Estos filtros son matrices pequeñas aplicando una operación de multiplicación y sumando los resultados para producir un mapa de características.
2. **Memoria prolongada de corto plazo (LSTM, por sus siglas en inglés *Long Short-Term Memory):*** Este tipo de modelo utiliza una estructura de celdas de memoria con puertas de entrada, salida y olvidar, esto con el fin de capturar la información a largo plazo de una secuencia de palabras.

2.3. Extractores de características basados en *embeddings*

Los extractores de características utilizados con *embedding* son modelos pre-entrenados para representar textos a vectores numéricos de alta dimensión [6]. Un ejemplo de un *embedding* es ***GloVe*** *(Global Vectors for Word Representation*), este es capaz de capturar información semántica y sintáctica de la palabra. GloVe utiliza una matriz de co-ocurrencia, la cual realiza la representación de la frecuencia de una palabra [7]. Después de la obtención de la matriz, se realiza una factorización para obtener los vectores de palabras finales, que capturen la co-ocurrencia de manera distribuida [7].



(a) Accuracy (b) Recall

Fig. 1. Curvas de métricas *accuracy* y *recall* para CNN.

3. Estado del arte

Existen múltiples trabajos en torno a la clasificación de sentimientos. En [8] se propone una clasificación de polaridad, con un conjunto de canciones tailandesas, basándose puramente en el texto de las mismas.

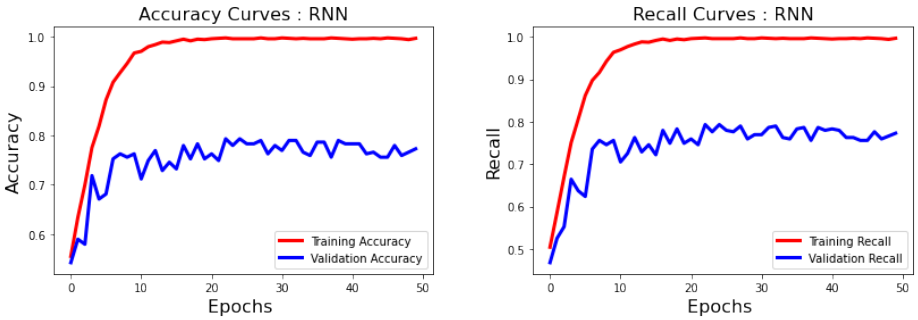
Proponen la utilización de un lexicón y utilización de técnicas de aprendizaje automático tradicional. De las diferentes partes de una canción (título, versos, estribillo, pre-coro, coro y puente) en donde solo se decidió utilizar el coro y los versos como corpus, ya que, se piensa que en estas dos partes es donde hay más probabilidad de que se encuentre el tema de la canción.

En el caso de [9] se utilizaron tres géneros como etiqueta (inspirador, divertido y romántico) para así poder utilizar minería de asociación en los datos de entrenamiento y encontrar a qué etiqueta pertenecen las palabras claves del texto de la canción. Posterior a esta tarea, se utilizó el modelo Naive Bayes para el cálculo de probabilidad. Encuentran una maximización de la probabilidad de observar las palabras que realmente se encontraron en los textos de ejemplo, mejorando así la habitual independencia de Naive Bayes.

Para [10] se utilizó una ontología llamada *SentiWordNet*, la cual incluye puntajes relacionados con los aspectos positivos o negativos de las palabras. La ontología fue utilizada para la extracción de características de sentimientos, todo esto en los textos de canciones, para encontrar el estado de ánimo al que pertenecen dichas canciones.

Los experimentos fueron desarrollados en un corpus de 185 canciones y se utilizaron tres diferentes algoritmos de clasificación; Naive Bayes, *K-Nearest Neighbor* y Máquinas de Vectores de Soporte (SVM por sus siglas en inglés *Support Vector Machine*).

En [11] se compara el rendimiento de algunos modelos de *Word embedding*, previamente entrenados en análisis de letras de canciones y tareas de polaridad de reviews de películas. Los resultados muestran que los *tweets* son lo mejor para el análisis de letras de canciones, mientras que Google *News* y *Common Crawl* son los mejores para el análisis de películas, ya que el vocabulario que se utiliza en estos portales es muy parecido en ambos casos. Los modelos entrenados con GLoVe superan ligeramente a los entrenados con Skip-gramas.



(a) Accuracy (b) Recall

Fig. 2. Curvas de métricas *accuracy* y *recall* para LSTM.

Por otro lado, se encontró que existen combinaciones de modelos que no son comúnmente usados para clasificación, como CNN [12]. Este modelo combina CNN y redes LSTM bidireccionales para capturar características relevantes del texto en diferentes niveles. Estos experimentos realizados en diferentes conjuntos demostraron un *accuracy* del 90.66 %.

4. Conjunto de datos

El conjunto de datos, denominado *Textos de canciones en español* [13], se trata de un corpus privado de PLN desarrollado en el Laboratorio de Procesamiento de Lenguaje Natural del Centro de Investigación en Computación del Instituto Politécnico Nacional[2].

El conjunto está formado por 91 canciones, todas escritas en el idioma español y con ritmos variados (bachata, pop, balada, entre otros). Cada canción fue seccionada en pequeños párrafos de manera manual, siguiendo un sentido de la oración, es decir, no se tienen ideas incompletas u oraciones que terminen en palabras de parada, lo que da un total de 1,477 datos.

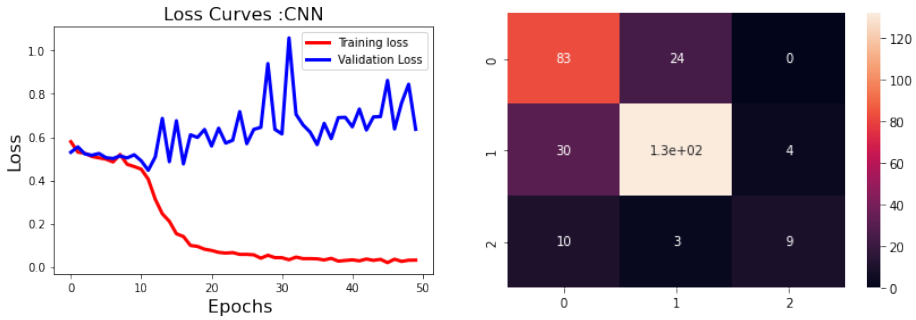
Para el etiquetado de canciones se utilizó un método desarrollado por los autores del conjunto. Para el etiquetado, se consideraron en 3 principales emociones: S, neutral; P, positivos; N, negativos. El cuadro 1 representa la distribución de los datos del conjunto, se puede notar que se trata de un conjunto desbalanceado. Para este trabajo, el conjunto de datos fue divido en 80 % para el entrenamiento y en 20 % para validación.

5. Propuesta de solución

5.1. Preprocesamiento

Para obtener los mejores resultados, es necesario preparar los datos de un texto con los mecanismos clásicos de preprocesamiento de datos, por ejemplo:

[2] Conjunto de datos *Textos de Canciones en español*, para consultarlo o acceder escriba a sidovor@cic.ipn.mx



(a) Función de perdida de CNN (b) Matriz de Confusión

Fig. 3. Función de perdida y matriz de confusión de CNN.

– Paso de mayúsculas a minúsculas.
– Remoción:
  1. Corrección ortográfica.
  2. Palabras gramaticales (*stop words*).
  3. Diagonales y diagonales invertidas.
  4. Números.
  5. Saltos de línea.
  6. Paréntesis.
  7. Dobles espacios en blanco.
– Tokenización.
– Lematización.
– Obtención de raíces (*Stemming*).

5.2. Extracción de características

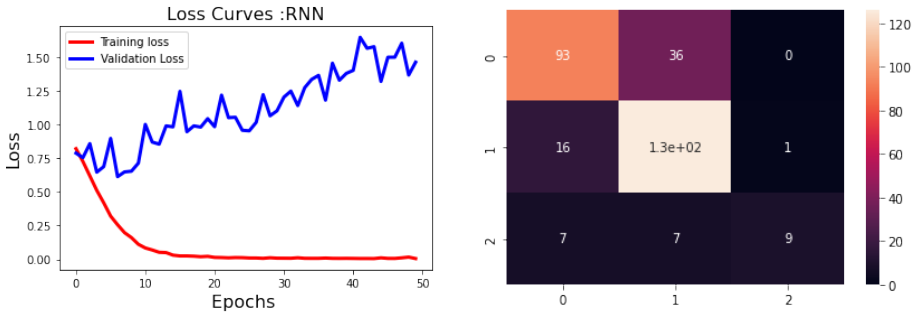
Se realizó la extracción de características por medio de GLoVe. La cual crea una matriz con la co-orcurrencia de la similitud de palabras dentro de una ventana (puede ser el número de palabras cercanas). Dicha matriz está conformada con la probabilidad de que dos palabras aparezcan. Para este trabajo se eligió una ventana de 10.

La matriz se transforma en otra por medio de la ponderación. La matriz, ahora, se pondera, es decir, a partir de una factorización matricial se hace la reducción de la dimensión de la matriz. Esta matriz se descompone en dos matrices para realizar representaciones vectoriales: Una para representar las co-ocurrencias y otra para contextos. Al finalizar estos procesos, las matrices se combinan, para obtener los *embeddings* finales.

5.3. Clasificación

Aprendizaje profundo

– **LSTM:** En la capa de entrada se define la máxima longitud de secuencia que aceptara la red, la segunda capa le pertenece al *embedding* GloVe, las siguientes



(a) Función de perdida de LSTM (b) Matriz de Confusión

Fig. 4. Función de perdida y matriz de confusión de LSTM.

capas pertenecen al algoritmo LSTM bidireccional, capas densas y al final la capa de salida en forma categórica. Posteriormente, se define la siguiente configuración: Función de perdida, *categorical crossentroy*; Optimizador, *ramsprop*;*Batch size*, 16; No. épocas, 50.

– **CNN:** Para este modelo se utilizaron tres capas convolucionales, tres capas de *MaxPooling* de una dimensión, una capa plana, una capa densa y por último, la capa de salida. La configuración fue la siguiente: Función de perdida, Binary crossentropy; Optimizador, rmsprop; Learning rate, 0.001.

6. Resultados

Se realizaron experimentos con clasificadores LSTM y CNN, acompañados de la extracción de características GLoVe. En las siguientes gráficas se muestran los mejores resultados de cada uno de los clasificadores.

En las figuras 1 para CNN y 2 para LSTM, se observa la evolución de las métricas por medio de las épocas.

En la figura 3a se observa que durante las primeras 15 épocas se obtuvo un comportamiento desfavorable en cuanto a la perdida, pero a partir de la época 16 esta mejoro llegando a un valor muy cercano a cero en la época 50. Para la figura 4a se observa que en el caso del entrenamiento para LSTM, existe un buen comportamiento, ya que, existe una curva favorable sobre la función de perdida, la cual llegó a valores muy próximos al cero.

En las figuras 3b para CNN y 4b, muestran las matrices de confusión, representadas como mapas de calor, donde los colores más claros representan resultados más favorables.

En el cuadro 2 se muestran los resultados para la métrica de *accuracy*. Se eligió reportar únicamente la métrica de *accuracy*, ya que es la más utilizada en el estado del arte.

En el cuadro 3 se muestran los resultados para la métrica *accuracy*, para los clasificadores en el estado del arte. En letras negritas, se puede observar el posicionamiento de los clasificadores descritos en este artículo.

Tabla 2. Resultados del metodo propuesto.

Clasificador	***Accuracy***
CNN	78.4 %
LSTM	80.1 %

Tabla 3. Resultados comparativo con los métodos del estado de arte.

Clasificador	***Accuracy***
CNN-BiLSTM [12]	90.66 %
Genre Classification [9]	85 %
LSTM (nuestro)	**80.1 %**
CNN (nuestro)	**78.4 %**
SentiWordNet 2 [10]	71 %
SentiWordNet 1 [10]	69 %
Thai Songs [8]	62 %
QWE [11]	61 %

Es importante mencionar que la comparación no puede ser 100 % directa, ya que el estado del arte, ni la propuesta en este artículo, utilizan el mismo conjunto de datos, pero es importante resaltar los resultados promedio en tareas similares, y así determinar una mejora en un trabajo a futuro.

7. Conclusiones y trabajo futuro

En este artículo se presentó la clasificación de sentimientos a través del extractor de características GLoVe, el cual extrae las características mediante las co-ocurrencias de palabras. Esta extracción de características, sirvió de entrada para clasificarlos por medio de LSTM y CNN.

Con los resultados obtenidos, se puede determinar un excelente punto de partida, ya que, aunque no se contemplaron modelos que consideran el contexto o modelos de atención, la graficas muestran que modificando los elementos que se toman para el entrenamiento se podrían mejorar los resultados obtenidos con creces.

Aunado a lo anterior, también se observa que se requiere de un modelo capaz de poder manejar cadenas de texto más largas, ya que el modelo LSTM tiene un rendimiento excelente solo con cadenas cortas. Por último, se utilizaron modelos de entrenamiento pequeños donde su coste computación es muy bajo, dando pie a que con modelos más robustos se incrementaría el valor de las métricas.

Se debe apreciar que involucrar la clasificación con una CNN no es usual, así que los resultados para la tarea de análisis de sentimientos, se demostró que con el preprocesamiento correcto y añadiendo capas adicionales de convolución, se podrían obtener resultados muy favorables.

Como trabajo a futuro se proponen diferentes enfoques: 1. Aplicar un extractor de características basado en el contexto y combinar las CNN con LSTM; 2. Aplicar mecanismos de atención que contemplen el contexto de las frases, poniendo especial enfoque en BERT o T5.

Referencias

1. Poria, S., Cambria, E., Hazarika, D., Majumder, N., Zadeh, A., Morency, L. P.: Context-dependent sentiment analysis in user-generated videos. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, vol. 1, pp. 873–883 (2017) doi: 10.18653/v1/P17-1081
2. Wang, C., Zhang, Q., Liu, W., Liu, Y., Miao, L.: Facial feature discovery for ethnicity recognition. WIREs Data Mining and Knowledge Discovery, vol. 9 (2019) doi: 10.1002/widm.1370
3. Mártinez-Cámara, E., Martín-Valdivia, M., Ureña, L. A.: Análisis de sentimientos. In: IV Jornadas TIMM Tratamiento de la Información Multilingüe y Multimodal (2011)
4. Rivera, M.: Perceptrón multicapa en Tensorflow-Keras. Aprendizaje Automático CIMAT (2022) personal.cimat.mx:8181/~mrivera/cursos/aprendizaje_profundo/mlp/mlp.html
5. Tariq, U., Tariq, S., Ahmad, R.: Deep Learning: A review of the state-of-the-art with an insight into Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM), pp. 71614–71630 (2018) doi: 10.1109/ACCESS.2018.2870225
6. Géron, A.: Hands-On machine learning with scikit-learn, keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media, Inc (2nd ed.) (2019)
7. Pennington, J., Socher, R., Manning, C.: GloVe: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014) doi: 10.3115/v1/D14-1162
8. Srinilta, C., Sunhem, W., Tungjitnob, S., Thasanthiah, S.: Lyric-based sentiment polarity classification of Thai songs. In: Proceedings of the International MultiConference of Engineers and Computer Scientists, vol. 1 (2017)
9. Giras, A., Advirkar, A., Patil, C., Khadpe, D., Pokhare, A.: Lyrics based song genre classification. Journal of Computing Technologies, vol. 3, no. 2 (2014) http://jctjournals.com/feb2014/v4.pdf
10. Kumar, V., Minz, S.: Mood classification of lyrics using SentiWordNet. In: 2013 International Conference on Computer Communication and Informatics, pp. 1–5 (2013) doi: 10.1109/ICCCI.2013.6466307
11. Çano, E., Morisio, M.: Quality of word embeddings on sentiment analysis tasks. Natural Language Processing and Information Systems, vol. 10260 (2017) doi: 10.1007/978-3-319-59569-6_42
12. Rhanoui, M., Mikram, M. Yousfi, S., Barzali, S.: CNN-BiLSTM model for document-level sentiment analysis. Machine learning and knowledge extraction, vol. 1, no. 3, pp. 832–847 (2019) doi: 10.3390/make1030048
13. Sidorov, G., Soto-Osorio, D., Chanona-Hernandez, L., Núñez-Prado, C. J.: Corpus "Textos de canciones en español". Laboratorio de Procesamiento de Lenguaje Natural, Centro de Investigación en Computación (2019)