

Pseudoetiquetado para el análisis de polaridad en tuits: Un primer acercamiento

Diana Jimenez, Marco A. Cardoso-Moreno,
Cesar Macias, Hiram Calvo

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Laboratorio de Ciencias Cognitivas Computacionales,
México

{djimenez12022, mcardosom2021,
cmaciass2021, hcalvo}@cic.ipn.mx

Resumen. El análisis de polaridad en textos es un tema de creciente interés, sobre todo en redes sociales, ya que ayuda a conocer si la opinión de las personas es negativa o positiva con respecto a un tema en particular en un ambiente libre, lo que nos permite conocer el impacto que productos y noticias, por mencionar algunos ejemplos, tienen en la sociedad. En este trabajo se realiza análisis de polaridad en tuits en español mexicano, mediante el uso de pseudoetiquetas generadas automáticamente, con el fin de intentar combatir la escasez de datos etiquetados, reto inherente a la tarea de análisis de polaridad dada la vasta cantidad de información disponible y lo complicado de asignar etiquetas adecuadas a la misma; además de técnicas de clasificación para tareas de procesamiento de lenguaje natural.

Palabras clave: Pseudoetiquetado, polaridad, tuits, PLN, aprendizaje automático.

Twits Pseudolabeling for Polarity Analysis: A First Approach

Abstract. Polarity analysis in texts is experiencing an interest growth, specially on social networks, due to its assistance in understanding if the opinion people have about certain topic, such as products or news, to name a few, is positive or negative, as well as the impact they have in societies. In this work we perform a polarity analysis on twits written in mexican spanish, by creating pseudolabels via machine learning techniques to try to overcome the issues related with the vast data available online and the enormous amount of effort that implies to correctly label it; in addition, we use classification techniques for natural language processing tasks.

Keywords: Polarity, pseudolabeling, twits, machine learning, NLP.

1. Introducción

Las redes sociales hoy en día forman una parte de la vida cotidiana para la población en general, ya sea para cuestiones de relaciones interpersonales, networking e incluso, para la consulta y diseminación de información [14, 17, 13] .

A partir de este incremento en el uso de redes sociales, intensificado en años recientes gracias a la pandemia de COVID-19 [6], es que estas plataformas se han vuelto parte del discurso público, ya que los algoritmos utilizados en las mismas permiten a sus usuarios interactuar con diversos grupos sociales, lo que los mantiene al tanto de los eventos y problemáticas actuales [4].

En particular, Twitter no presenta muchas restricciones sobre el contenido de las publicaciones que sus usuarios pueden efectuar por lo que, en general, suelen ser sobre cualquier tema, esta aparente libertad que la plataforma provee es la principal razón de que esta red social tiene preferencia entre los internautas para, en ella, mostrar sus opiniones [13].

Dentro de las áreas de procesamiento de lenguaje natural (PLN) y lingüística computacional existe la tarea de análisis de opiniones, que consiste en, mediante el análisis del texto donde un comentario opinión fue expresado, determinar el la opinión que una persona sobre el tema en cuestión [15]; el análisis de la polaridad en una opinión se considera, a su vez, una subtarea de este campo [10].

Determinar la polaridad de un texto se refiere, entonces, a clasificar, dado un texto, si la opinión que se ha vertido en este es positiva o negativa, es decir, qué tan polarizada resulta.

2. Revisión de la literatura

La clasificación de textos, por su parte, puede llevarse a cabo mediante estrategias de aprendizaje automática, específicamente, aprendizaje supervisado. Estas técnicas han sido, y siguen siendo, ampliamente utilizadas en la clasificación de textos para diferentes tareas, siendo una de las más destacadas el análisis de sentimientos.

Por ejemplo, en [7] se hace uso de redes neuronales recurrentes (RNR), específicamente redes Bi-LSTM (del inglés Bi Long Short-Term Memory) para esta tarea; de manera similar, [1] utilizan Twitter como un medio donde la gente puede expresar síntomas de depresión que requieren ser reportados por un individuo con esta afectación psicológica para detectar dicho padecimiento de manera temprana, para lo cual utilizaron RNR tradicionales, así como redes LSTM; en [18] utilizan el modelo de transforme RoBERTa-GRU (del inglés Robustly Optimized BERT Pretraining Approach y Gated Recurrent Units) para la clasificación de sentimientos en diversos datasets considerados como baselines; por su parte, [5] utilizan el clasificador Naive Bayes (NB) para la misma tarea, apoyándose del recurso léxico sentiwordnet para agregar a cada palabra un puntaje de sentimiento positivo, negativo u objetivo.

En cuanto a la tarea específica de análisis de polaridad en texto, uno de los primeros trabajos que se llevaron a cabo fue aquel de [16], en el cual se utilizaron clasificadores tradicionales, tales como: NB, Entropía Máxima y Máquinas de Soporte Vectorial (SVM, del inglés Support Vector Machines) para la clasificación de polaridad

Tabla 1. Las cuatro configuraciones utilizadas; para cada una de ellas se muestra si se removieron o no palabras auxiliares, y lematización.

Configuración	Remoción de palabras Auxiliares	Lematización
1	No	No
2	No	Sí
3	Sí	No
4	Sí	Sí

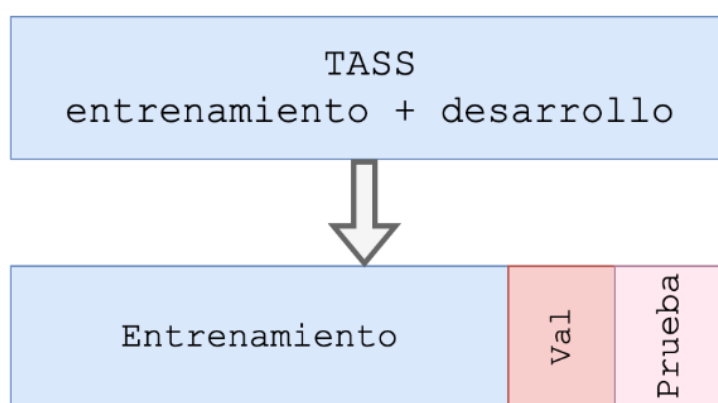


Fig. 1. Particiones sobre el dataset TASS.

de reseñas (en inglés) de películas; en [8] se realiza un estudio sobre el impacto de la negación en la clasificación de la polaridad en tuits en español, concluyendo que el tomar en cuenta dicho aspecto contribuye significativamente a una mejora en la clasificación de la polaridad; además, se encuentra el trabajo de [12], donde se utilizan múltiples clasificadores, tales como: Entropía Máxima, NB Multinomial (NBM), SVMs y BETO, un modelo BERT (del inglés Bidirectional Encoder Representations from Transformers) entrenado con un corpus en español, para obtener la polaridad de tuits en español, lo que incluía encabezados de noticias, los hilos de la conversación correspondiente a dichos encabezados, tuits citados y los hilos de conversación que se generaron a partir de éstos.

También podemos encontrar el trabajo de Arias et al. [2], quienes crearon un banco de datos mediante la extracción de tuits a través de la API de Twitter, con clases: positivo, negativo y neutro, durante las pruebas de clasificación los autores optaron por diversos modelos, como son: Random Forest (RF), K-nearest Neighbors (KNN), NB, Gradient Boosting (GB), Support Vector Classifier (SVC) y Extreme Gradient Boosting (XGBoost).

Por otro lado, se ha observado que los modelos de aprendizaje de máquina y aprendizaje profundo suelen ver afectado su desempeño cuando no se cuenta con suficientes datos, por lo que se suelen utilizar técnicas de aumento de datos y, en los casos en los que no se puede preservar las etiquetas, pseudoetiquetado, tal es el caso de [3], donde se utilizó dicha técnica para mejorar el desempeño de diversas arquitecturas de redes neuronales para la tarea de detección de agresión en redes sociales.

Experimento 1

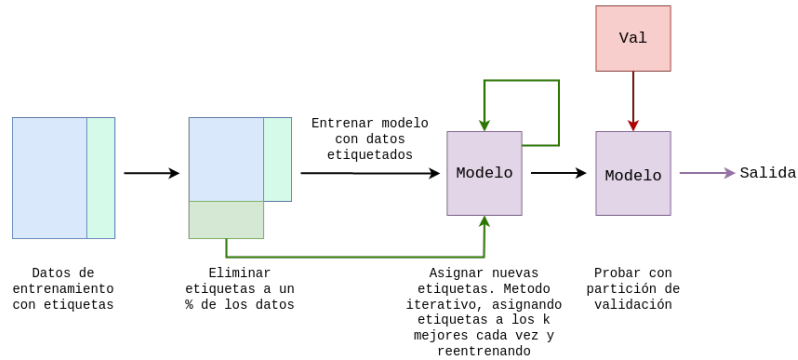


Fig. 2. Experimento 1.

Experimento 2

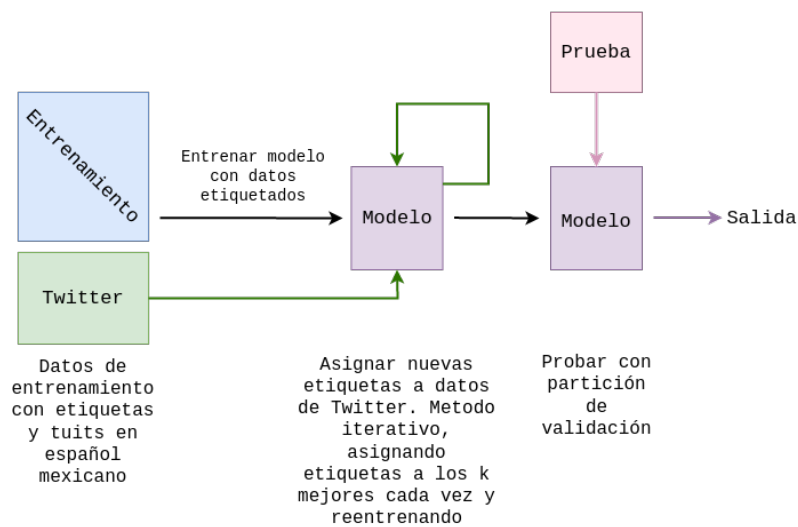


Fig. 3. Experimento 2.

También destaca el trabajo de [11], donde se hace uso del modelo DistilBERT en la tarea de clasificación de preguntas para su incorporación de sistemas tipo chatbots dedicados a responder preguntas, para contravenir la falta de los datos etiquetados se utilizan técnicas de pseudoetiquetado, obteniendo como resultado que el modelo cuyo banco de datos de entrenamiento datos pseudoetiquetados presentó un mejor desempeño que aquellos con los cuáles no se utilizó dicha técnica para su entrenamiento.

Tabla 2. Resultados del experimento 1 con el modelo de RL.

Conf	% ignorado	Recall	Precisión	Acc	F1
1	0	0.4981	0.5018	0.5822	0.4992
	20	0.5297	0.5493	0.6151	0.5341
	40	0.5028	0.5142	0.5813	0.5048
	60	0.4829	0.4982	0.584	0.4825
	80	0.4222	0.437	0.584	0.4094
2	0	0.4992	0.49	0.5644	0.4557
	20	0.4316	0.4777	0.544	0.4287
	40	0.4339	0.4589	0.5333	0.4293
	60	0.4288	0.4742	0.5458	0.4191
	80	0.4038	0.4383	0.5124	0.3824
3	0	0.5262	0.5469	0.6044	0.5326
	20	0.5105	0.5212	0.5884	0.5137
	40	0.5036	0.5231	0.5862	0.5079
	60	0.4696	0.4908	0.5644	0.4694
	80	0.421	0.4607	0.5378	0.4076
4	0	0.4731	0.4771	0.5422	0.4747
	20	0.4953	0.5005	0.5667	0.4958
	40	0.4809	0.4878	0.5596	0.4815
	60	0.4539	0.4662	0.5489	0.4544
	80	0.4284	0.434	0.5307	0.4212

Por último, en [9] se hace uso del pseudoetiquetado para la mejora en los sistemas de detección de noticias, ya que el etiquetado manual de texto suele ser una tarea laboriosa, sobre todo dada la ingente cantidad de recursos disponibles en internet, lo que resulta en una falta de datos etiquetados disponibles; en este trabajo se observó un incremento en el desempeño de clasificación de entre el 2% y 3% cuando se agregaron nuevos datos cuya etiqueta fue asignada mediante algún algoritmo.

3. Desarrollo del proyecto

3.1. Datasets

Se utilizaron dos datasets: del TASS 2019 se seleccionaron las partes de entrenamiento y desarrollo para ser unidas en una sola partición, adicionalmente, se hizo un proceso de web scraping mediante la API versión 2 de Twitter, con lo que se extrajeron 6500 tuits en español mexicano.

3.2. Preproceso de datos

Como parte de la etapa del preprocesamiento de datos se llevaron a cabo los siguientes procedimientos:

- Entidades HTML: se remueven las entidades HTML que contenga el texto.

Tabla 3. Resultados del experimento 1 con el modelo de SVM.

Conf	% ignorado	Recall	Precisión	Acc	F1
1	0	0.5039	0.5899	0.6444	0.4949
	20	0.526	0.5379	0.6049	0.5291
	40	0.4979	0.5104	0.5884	0.4989
	60	0.4758	0.4952	0.5778	0.4749
	80	0.4275	0.4484	0.5449	0.4187
2	0	0.4538	0.6328	0.5956	0.4448
	20	0.4416	0.4913	0.5547	0.4371
	40	0.4317	0.4821	0.536	0.4237
	60	0.4312	0.4545	0.5422	0.4222
	80	0.4078	0.4428	0.5302	0.3874
3	0	0.4764	0.7902	0.6311	0.4601
	20	0.5211	0.5388	0.5969	0.5259
	40	0.5049	0.5187	0.5827	0.508
	60	0.4872	0.521	0.5756	0.4918
	80	0.4291	0.4786	0.5471	0.4224
4	0	0.4901	0.6983	0.6356	0.4822
	20	0.4794	0.4848	0.5591	0.4804
	40	0.4761	0.489	0.5578	0.4764
	60	0.4535	0.4679	0.5418	0.4535
	80	0.4326	0.4592	0.5329	0.4279

- Saltos de línea: se quitan los saltos de línea.
- Hashtags: En caso de haber hashtags, se separa el texto contenido en los mismos (p.e. #CiudadDeMexico → Ciudad De Mexico).
- Entidades de Twitter: se les dice así a las entidades que se utilizan propiamente en Twitter para denotar usuarios, etiquetas, hashtags y retuits, cada uno de estos tiene un identificador especial (@User, rt, #hashtag), se identifican estas entidades y se remueven del texto.
- URLs: se identifican y se remueven del texto.
- Transformar el texto a únicamente letras minúsculas.
- Palabras auxiliares: en caso de que así se requiera, se remueven las palabras auxiliares que contenga el texto.
- Lematización: si se requiere, las palabras son lematizadas utilizando la librería spaCy.
- Apóstrofes: tras la lematización, se remueven los apóstrofes del texto, conservando el caracter sin el apóstrofe (p.e concatenación, niño → concatenacion, nino).
- Puntuación: se remueven los caracteres utilizados para puntuar el texto (puntos, comas, punto y comas, etc.).
- Caracteres repetidos: en caso de que un caracter se repita más de tres veces, este se corta a dos repeticiones (p.e. Nooooo → Noo).
- Palabras alfanuméricas: si el texto contiene palabras compuestas por letras y números, como en el leet speaking, estas se remueven.
- Caracteres especiales: se remueven todos los caracteres especiales que no aportan nada al texto, signos de admiración, interrogación, etc.

Tabla 4. Resultados del experimento 1 con el modelo de NBM.

Conf	% ignorado	Recall	Precisión	Acc	F1
	0	0.4883	0.4883	0.4883	0.4883
1	20	0.4589	0.4550	0.6244	0.4235
	40	0.4356	0.4882	0.6031	0.3964
	60	0.4001	0.4511	0.5707	0.3455
	80	0.3498	0.4344	0.5218	0.2604
	0	0.4883	0.5459	0.5956	0.4406
2	20	0.4502	0.5855	0.5973	0.4343
	40	0.4271	0.5129	0.5836	0.3969
	60	0.4137	0.4854	0.5764	0.3723
	80	0.3856	0.4493	0.5551	0.3245
	0	0.5539	0.624	0.6667	0.5608
3	20	0.5201	0.6097	0.648	0.5127
	40	0.5099	0.5996	0.644	0.4969
	60	0.4671	0.5786	0.6076	0.448
	80	0.4327	0.4582	0.5636	0.3937
	0	0.5341	0.6427	0.6578	0.5388
4	20	0.5159	0.6908	0.6498	0.5131
	40	0.4913	0.6497	0.6329	0.4772
	60	0.4751	0.6585	0.6147	0.457
	80	0.3765	0.5051	0.5391	0.3169

- Espacios en blanco: en caso de que exista más de un espacio en blanco entre palabras, estos se remueven para homogeneizar el texto.

Durante este proceso se crearon cuatro configuraciones para los conjuntos de datos utilizados, las cuáles difieren únicamente en si se incluyó o no, tanto la remoción de palabras auxiliares como la lematización de los textos; es decir, las cuatro configuraciones utilizadas contienen todo el preproceso enlistado previamente, y solo difieren entre sí por la presencia o ausencia de palabras auxiliares y la lematización. La Tabla 1 muestra cada una de las configuraciones.

Por último, para la extracción de características se utilizó un método de bolsa de palabras binario, donde los vectores solo tienen valores 0 y 1, dependiendo de si la palabra está presente o no en un tuit dado.

3.3. Experimentos

En este trabajo se utilizan tres modelos: Regresión Logística (RL), NBM y una SVM con kernel lineal; además, creamos una partición de tres conjuntos sobre el dataset generado a partir de TASS: una para entrenamiento, que se conforma por el 70 % de los datos; además de dos particiones de validación y prueba, respectivamente, cada una formada por 15 %. La Figura 1 muestra gráficamente el proceso de partición.

Se realizaron dos tipos de experimentos, cada uno de ellos utilizando los 3 modelos seleccionados, para las 4 configuraciones de datos descritas en la Tabla 1.

Tabla 5. Resultados del experimento 2 con el modelo de RL.

Conf	% pseudo	Total	Orig	Pseudo	Recall	Precisión	Acc	F1
1	0	1050	1050	0	0.5403	0.5526	0.6267	0.5405
	20	1312	1050	263	0.5404	0.5682	0.6262	0.5440
	40	1749	1050	700	0.5516	0.5764	0.6324	0.5557
	60	2623	1050	1574	0.5451	0.5694	0.6248	0.5484
	80	5245	1050	4196	0.5495	0.5774	0.6328	0.5532
2	0	1050	1050	0	0.4217	0.4474	0.5333	0.4079
	20	1312	1050	263	0.3642	0.4583	0.508	0.3035
	40	1749	1050	700	0.3623	0.4656	0.5075	0.2955
	60	2623	1050	1574	0.3656	0.4369	0.5106	0.3013
	80	5245	1050	4196	0.3662	0.4571	0.5111	0.3033
3	0	1050	1050	0	0.5034	0.507	0.5867	0.5021
	20	1312	1050	263	0.5226	0.5484	0.6004	0.5273
	40	1749	1050	700	0.5296	0.5471	0.604	0.533
	60	2623	1050	1574	0.528	0.5495	0.6062	0.5315
	80	5245	1050	4196	0.5248	0.545	0.6027	0.5279
4	0	1050	1050	0	0.5155	0.5313	0.5956	0.5177
	20	1312	1050	263	0.4913	0.5016	0.5796	0.4894
	40	1749	1050	700	0.4864	0.4979	0.5733	0.4843
	60	2623	1050	1574	0.4818	0.4935	0.5702	0.4803
	80	5245	1050	4196	0.4919	0.5007	0.58	0.4885

En el primer experimento se utiliza la partición de entrenamiento para entrenar al modelo como pseudoetiquetador.

En primera instancia, mediante un proceso inspirado en k-fold cross validation, se eliminan las etiquetas de un porcentaje de los datos de entrenamiento (variando en cada iteración los datos a los cuales se les quitó su etiqueta); posteriormente se entrena al modelo con los datos que aún mantienen su etiqueta verdadera; toda vez que se ha entrenado al modelo, se procede a asignar etiquetas a los datos que carecen de ellas, aquí se conservan los k mejores resultados para reentrenar el modelo con los datos con etiquetas verdaderas y los nuevos k datos, este proceso continúa hasta terminar de asignar nuevas etiquetas; por último, se utiliza el conjunto de validación (Figura 1) para medir el desempeño del modelo, tanto como pseudoetiquetador y como clasificador al final del experimento. La Figura 2 muestra el procedimiento llevado a cabo.

En el segundo experimento se vuelve a trabajar con el conjunto de entrenamiento para utilizar al modelo como pseudoetiquetador, una vez que se ha entrenado, se procede a añadir etiquetas a los datos extraídos de Twitter (que no tienen una etiqueta asignada) mediante el mismo proceso del experimento 1, es decir, se asignan etiquetas a todos los datos, se conservan las k mejores para reentrenar al modelo y así sucesivamente, hasta terminar de etiquetar los datos de Twitter.

Una vez que se ha concluido esa primera etapa, se procede a medir el desempeño del modelo con el conjunto de prueba (Figura 1). De igual modo, la Figura 3 muestra el procedimiento realizado en el experimento 2.

Tabla 6. Resultados del experimento 2 con el modelo de SVM.

Conf	% pseudo	Total	Orig	Pseudo	Recall	Precisión	Acc	F1
1	0	1050	1050	0	0.4938	0.7688	0.6311	0.4599
	20	1312	1050	263	0.4979	0.7707	0.6329	0.4706
	40	1749	1050	700	0.4988	0.7713	0.6342	0.4713
	60	2623	1050	1574	0.4933	0.7684	0.6307	0.4595
	80	5245	1050	4196	0.4938	0.7688	0.6311	0.4599
2	0	1050	1050	0	0.4135	0.4816	0.5644	0.3692
	20	1312	1050	263	0.3904	0.4629	0.5458	0.3282
	40	1749	1050	700	0.4054	0.4742	0.56	0.3512
	60	2623	1050	1574	0.4096	0.471	0.5627	0.3591
	80	5245	1050	4196	0.415	0.4729	0.5667	0.3689
3	0	1050	1050	0	0.4652	0.7853	0.6089	0.4335
	20	1312	1050	263	0.4613	0.619	0.6067	0.4258
	40	1749	1050	700	0.4617	0.6507	0.6067	0.4265
	60	2623	1050	1574	0.4648	0.785	0.6084	0.433
	80	5245	1050	4196	0.4652	0.7853	0.6089	0.4335
4	0	1050	1050	0	0.4592	0.4278	0.6044	0.4174
	20	1312	1050	263	0.4587	0.4274	0.604	0.4169
	40	1749	1050	700	0.4587	0.4274	0.604	0.4169
	60	2623	1050	1574	0.4592	0.4278	0.6044	0.4174
	80	5245	1050	4196	0.4592	0.4278	0.6044	0.4174

4. Resultados

En esta sección se presentan los resultados de ambos experimentos. En las Tablas 2, 3 y 4 se presentan los resultados obtenidos en el experimento 1 con los modelos RL, SVM y NBM, respectivamente.

Por su parte, las tablas Tablas 5, 6 y 7 muestran los resultados obtenidos en el experimento 2 con los modelos RL, SVM y NBM, respectivamente.

5. Conclusiones

En este trabajo se presentó un primer acercamiento en el uso de técnicas de pseudoetiquetado para el robustecimiento de modelos desarrollados para la tarea de análisis de polaridad en tuits.

En primera instancia, el método aquí propuesto mostró ayudar al desempeño de los modelos de RL y SVM que, en términos generales, crean una frontera de decisión a través de un hiperplano que separa el espacio de características; por su parte, nuestro método no refleja una mejora en los resultados del clasificador NBM, que realiza sus decisiones mediante computaciones probabilísticas.

Es importante hacer notar que en el experimento 1, para aquellos casos en los que hubo una mejora en la clasificación gracias al pseudoetiquetado, dicha mejora se presenta cuando el porcentaje de datos ignorados no excede la mitad de la cardinalidad del dataset original; es decir, una vez que se ignora la etiqueta de la mitad, o más, patrones, deja de haber una mejora en los resultados.

Tabla 7. Resultados del experimento 2 con el modelo de NBM.

Conf	%	pseudo	Total	Orig	Pseudo	Recall	Precisión	Acc	F1
	0	1050	1050	0	0.461	0.5766	0.5956	0.4343	
1	20	1312	1050	263	0.4569	0.665	0.5964	0.4277	
	40	1749	1050	700	0.4356	0.533	0.5844	0.3921	
	60	2623	1050	1574	0.4211	0.4577	0.5733	0.3727	
	80	5245	1050	4196	0.3962	0.4833	0.5516	0.3365	
	0	1050	1050	0	0.3858	0.4469	0.5289	0.3372	
2	20	1312	1050	263	0.3366	0.3983	0.4964	0.2272	
	40	1749	1050	700	0.3333	0.1644	0.4933	0.2202	
	60	2623	1050	1574	0.3333	0.1644	0.4933	0.2202	
	80	5245	1050	4196	0.3333	0.1644	0.4933	0.2202	
	0	1050	1050	0	0.4981	0.5527	0.6044	0.4874	
3	20	1312	1050	263	0.4908	0.5452	0.5982	0.4824	
	40	1749	1050	700	0.4738	0.5413	0.5902	0.4570	
	60	2623	1050	1574	0.4638	0.5367	0.5916	0.4384	
	80	5245	1050	4196	0.4506	0.5183	0.5871	0.4143	
	0	1050	1050	0	0.4608	0.4934	0.5778	0.4347	
4	20	1312	1050	263	0.4679	0.5188	0.5849	0.4447	
	40	1749	1050	700	0.4615	0.5085	0.588	0.4288	
	60	2623	1050	1574	0.4571	0.5472	0.5924	0.4196	
	80	5245	1050	4196	0.4505	0.4724	0.592	0.4102	

Por último, se observa que el ignorar etiquetas de algunos patrones, para posteriormente asignarles a los mismos nuevas etiquetas mediante pseudoetiquetado, contribuyó a una mejora en el desempeño de los modelos, lo que presenta una nueva veta de investigación que permita utilizar el pseudoetiquetado como un método de regularización.

Referencias

1. Apoorva, A., Goyal, V., Kumar, A., Singh, R., Sharma, S.: Depression detection on twitter using RNN and LSTM models. In: *Advanced Network Technologies and Intelligent Computing: Second International Conference, ANTIC 2022*, vol. 1798, pp. 305–319 (2023) doi: 10.1007/978-3-031-28183-9_22
2. Arias, F., Guerra-Adames, A., Zambrano, M., Quintero-Guerra, E., Tejedor-Flores, N.: Analyzing spanish-language public sentiment in the context of a pandemic and social unrest: The Panama case. *International Journal of Environmental Research and Public Health*, vol. 19, no. 16 (2022) doi: 10.3390/ijerph191610328
3. Aroyehun, S. T., Gelbukh, A.: Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling. In: *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC'08)*, pp. 90–97 (2018)
4. Bastick, Z.: Would you notice if fake news changed your behavior? An experiment on the unconscious effects of disinformation. *Computers in Human Behavior*, vol. 116, pp. 1–12 (2021) doi: 10.1016/j.chb.2020.106633
5. Goel, A., Gautam, J., Kumar, S.: Real time sentiment analysis of tweets using Naive Bayes. In: *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pp. 257–261 (2016) doi: 10.1109/NGCT.2016.7877424

6. Greenhow, C., Staudt-Willet, K. B., Galvin, S.: Inquiring tweets want to know: #Edchat supports for # RemoteTeaching during COVID-19. *British Journal of Educational Technology*, vol. 52, no. 4, pp. 1434–1454 (2021) doi: 10.1111/bjet.13097
7. Jaca-Madariaga, M., Zarrabeitia-Bilbao, E., Rio-Belver, R. M., Moens, M. F.: Sentiment analysis model using Word2vec, Bi-LSTM and attention mechanism. In: *IoT and Data Science in Engineering Management: Proceedings of the 16th International Conference on Industrial Engineering and Industrial Management and XXVI Congreso de Ingeniería de Organización*, vol. 160, pp. 239–244 (2023) doi: 10.1007/978-3-031-27915-7_43
8. Jimenez-Zafra, S. M., Martin-Valdivia, M. T., Martinez-Camara, E., Urena-Lopez, L. A.: Studying the scope of negation for spanish sentiment analysis on twitter. *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 129–141 (2019) doi: 10.1109/TAFFC.2017.2693968
9. Jiménez, D., Gambino, O. J., Calvo, H.: Pseudo-labeling improves news identification and categorization with few annotated data. *Computación y Sistemas*, vol. 26, no. 1, pp. 183–193 (2022) doi: 10.13053/cys-26-1-4163
10. Juárez-Gambino, J. O.: Sentiment polarity prediction of twitter users' opinions to national newspapers news. Ph.D. thesis, Centro de Investigación en Computación (2019)
11. Kuligowska, K., Kowalczyk, B.: Pseudo-labeling with transformers for improving question answering systems. *Procedia Computer Science*, vol. 192, pp. 1162–1169 (2021) doi: 10.1016/j.procs.2021.08.119
12. Macias, C., Calvo, H., Gambino, O.: News intention study and automatic estimation of its impact. In: *Advances in Computational Intelligence - 21st Mexican International Conference on Artificial Intelligence, MICAI'22, Proceedings*, vol. 13613, pp. 83–100 (2022) doi: 10.1007/978-3-031-19496-2_7
13. Macias-Sánchez, C.: Estudio de la intención de noticias y estimación automática de su impacto. Master's thesis, Centro de Investigación en Computación, Instituto Politécnico Nacional (2022)
14. Madni, H. A., Umer, M., Abuzinadah, N., Hu, Y. C., Saidani, O., Alsubai, S., Hamdi, M., Ashraf, I.: Improving sentiment prediction of textual tweets using feature fusion and deep machine ensemble model. *Electronics*, vol. 12, no. 6 (2023) doi: 10.3390/electronics12061302
15. Mejova, Y.: Sentiment analysis: An overview. University of Iowa, Computer Science Department, pp. 1–34 (2009)
16. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pp. 79–86 (2002) doi: 10.3115/1118693.1118704
17. Pezoa-Fuentes, C., García-Rivera, D., Matamoros-Rojas, S.: Sentiment and emotion on twitter: The case of the global consumer electronics industry. *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 18, no. 2, pp. 765–776 (2023) doi: 10.3390/jtaer18020039
18. Tan, K. L., Lee, C. P., Lim, K. M.: RoBERTa-GRU: A hybrid deep learning model for enhanced sentiment analysis. *Applied Sciences*, vol. 13, no. 6 (2023) doi: 10.3390/app13063915