Universidad Nacional del Altiplano
Ingenier´ıa Estad´ıstica e Inform´atica
Alumno: Edson Bladimir Pinto Luque
Código: 171198

Tarea - N° 002

Minería de Datos [Técnicas]

MODELADO DEL LENGUAJE Y CLASIFICACIÓN DE TEXTOS

1. DEFINICIÓN

La clasificación de texto y el modelado del lenguaje son dos actividades de PNL (procesamiento del lenguaje natural) relacionadas pero separadas. [1]

La construcción de un modelo estadístico o computacional que pueda anticipar la siguiente palabra o frase en un texto dado se conoce como modelado de lenguaje. El propósito básico del modelado del lenguaje es capturar la estructura y las relaciones gramaticales en el texto y desarrollar un modelo que pueda generar una escritura coherente y significativa. [2] Los grandes conjuntos de datos de texto se utilizan para entrenar modelos de lenguaje, que luego se pueden aplicar a una variedad de tareas, incluida la producción automática de texto, el autocompletado de palabras en dispositivos móviles y la traducción automática.

Por otro lado, la clasificación de texto se enfoca en asignar una o más etiquetas predefinidas a un texto determinado. El objetivo es clasificar el texto en categorías o clases específicas según su contenido o tema. Para hacer esto, primero se debe entrenar un modelo para que pueda reconocer patrones y características en el texto y aplicar ese conocimiento para categorizar el texto nuevo. Algunas instancias populares de categorización de texto son la detección de correo no deseado, el análisis de sentimientos de las redes sociales [3] [4], la división de noticias en distintos grupos y la detección de temas de documentos.

En resumen, el modelado del lenguaje se enfoca en predecir palabras o frases en función del contexto del texto, mientras que la clasificación de texto se enfoca en asignar etiquetas o categorías predefinidas a un texto en función de su contenido. Ambas tareas son cruciales en el procesamiento del lenguaje natural y se emplean en una amplia variedad de aplicaciones y sistemas.

2. LISTA DE TÉCNICAS

- Métodos para modelar lenguajes

  a) Modelos de N-gramas: se basan en la frecuencia con la que un corpus de textos contiene secuencias de palabras de longitud N.

  (b) Modelos de lenguaje basados en estadísticas: utilizan enfoques como cadenas de Markov, modelos ocultos de Markov (HMM) y modelos de máxima entropía para capturar la probabilidad de ocurrencia de palabras o frases en un contexto particular.

(c) Modelos de lenguaje neuronal: use redes neuronales, como redes neuronales recurrentes (RNN) y redes neuronales convolucionales (CNN), para modelar el contexto y las relaciones entre las palabras en un texto.

(d) Modelos de lenguaje basados en transformadores: se basan en la arquitectura transformadora, que permite capturar relaciones de largo alcance entre palabras en un texto y ha demostrado ser especialmente eficaz en el modelado de lenguajes. Algunos ejemplos incluyen GPT (Generative Pre-trained Transformador) y BERT (Representaciones de codificador bidireccional).

• Métodos para la Clasificación de Textos:

(a) Representaciones de vectores de palabras: para captar mejor el significado y la similitud semántica entre las palabras, utilice los métodos Word2Vec o GloVe para representar las palabras como vectores numéricos densos.

(b) Modelos de clasificación convencionales: estos modelos clasifican los textos en varios grupos utilizando métodos de aprendizaje supervisado como Naive Bayes, Support Vector Machines (SVM) o Random Forests.

(c) Redes neuronales para clasificación de texto: utilizan arquitecturas de redes neuronales, como RNN, CNN o modelos de atención, para extraer características y patrones relevantes del texto y clasificarlo en diferentes categorías.

(d) Transferencia de aprendizaje: toman modelos previamente entrenados en conjuntos de datos masivos, como GPT o BERT, y los modifican utilizando métodos de ajuste fino para tareas particulares de categorización de texto.

3. CÓDIGO

El código utilizado es el siguiente

```
1 importar pandas como
pd 2 importar numpy
como np 3 importar tensorflow
como tf 4 desde tensorflow importar
keras 5 desde tensorflow.keras.preprocessing.text importar Tokenizer 6 desde
tensorflow.keras.preprocessing.sequence importar pad_sequences
7
8 # Cargar los datos desde el archivo CSV 9 data =
pd.read_csv('datos.csv', encoding='latin-1')
10
11 # Dividir los datos en conjuntos de entrenamiento y prueba 12 train_data = data.sample(frac=0.8, random_state=42) 13 test_data =
data.drop(train_data.index)

14
```

```
15 # Obtener las columnas relevantes para el modelado de lenguaje y clasificación
       de texto
test_data['Sentiment'].valores




20

21 # Crear los diccionarios de etiquetas para convertir las etiquetas de texto
       un número
22 label_dict = {'Positivo': 0, 'Negativo': 1, 'Neutral': 2, 'Extremadamente positivo': 3,
       'Extremadamente negativo': 4} 23
inverse_label_dict = {v: k para k, v en label_dict. elementos()}
24

25 # Convertir las etiquetas de texto a nmeros 26
train_labels = np.array([label_dict[label] for label in train_labels]) 27 test_labels =
np.array([label_dict[label] for label in test_labels])
28

30 # Modelado de lenguaje _secuencias =
pad_secuencias ( prueba_secuencias , _ _ _




       maxlen=train_padded_sequences.shape[1])
36

37 # Definir modelo de lenguaje 38
model =
39          keras.Sequential([ keras.layers.Embedding(len(tokenizer.word_index) +
40          1, 16), keras.layers.GlobalAveragePooling1D(),
41          keras.layers.Dense(16, activación= 'relu'),
42          keras.layers.Dense(len(label_dict), activación='softmax')
43 ])
44 model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
       métricas=['precisión']) 45
modelo.fit(tren_padded_sequences, tren_etiquetas, épocas=10)
46

47 # Evaluacin en el conjunto de prueba 48
test_loss, test_accuracy = model.evaluate(test_padded_sequences, test_labels) 49
       print("Prdida
en el conjunto de prueba:", test_loss) 50 print("Precisin en el conjunto
de prueba:", test_accuracy )
51

52 # Clasificación de texto en el conjunto de prueba 53 test_predictions =
model.predict(test_padded_sequences) 54 test_predictions_classes = np.argmax(test_predictions, axis=1)


55
```

```
# Convertir las etiquetas numricas predichas a texto
inverse_label_func = np.vectorize(lambda x: inverse_label_dict[x])
predicted_labels = inverse_label_func(test_predictions_classes)

# Comparar las etiquetas reales con las predichas
for i in range(len(test_sentences)):
    print("Texto:", test_sentences[i])
    print("Etiqueta real:", test_labels[i])
    print("Etiqueta predicha:", predicted_labels[i])
    print("----------")
```

Listado 1: Código para modelado de lenguaje y clasificación de texto

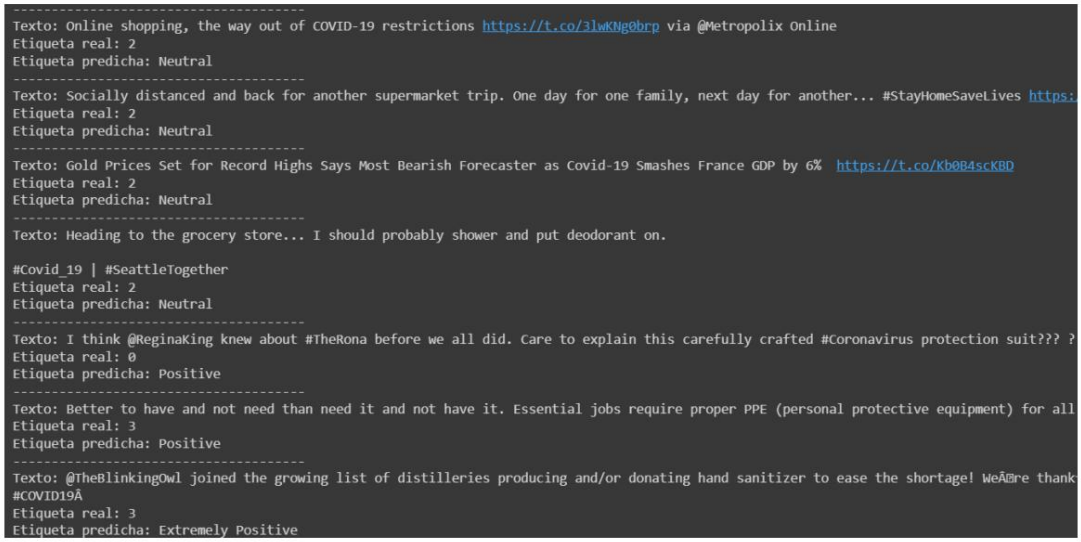

Figura 1: Resultado de la predicción

4. EJEMPLOS DE ARTÍCULOS

- ”UN LENGUAJE DE MODELADO DE AMENAZA PARA GENERAR GRÁFICOS DE ATAQUE SCI DE SISTEMAS DE AUTOMATIZACIÓN DE SUBESTACIONES”

  El artículo analiza las medidas de seguridad existentes para proteger los sistemas de automatización de las subestaciones, como los cortafuegos, la autenticación de usuarios y el cifrado de datos. Sin embargo, se enfatiza la necesidad de mejorar la seguridad en este ámbito, ya que los ciberataques evolucionan constantemente y se vuelven cada vez más sofisticados. [5], En resumen, el artículo aboga por mejorar la seguridad de los sistemas de automatización de subestaciones y propone el uso de un lenguaje de modelado de amenazas específico para este entorno, con el objetivo de identificar y reducir posibles vulnerabilidades y riesgos de ciberseguridad.

- "¿PUEDEN LOS MODELOS DE LENGUAJE DE LA IA REEMPLAZAR A LOS PARTICIPANTES HUMANOS?"

El artículo explora la posibilidad de usar modelos de lenguaje como participantes en la investigación psicológica, particularmente en términos de capturar juicios morales similares a los humanos. Se discuten las ventajas y limitaciones de los modelos de lenguaje en relación con los participantes humanos y se destacan las áreas en las que pueden complementar la investigación. Si bien los modelos son prometedores en ciertos aspectos, se concluye que aún se necesita una comprensión directa de la mente humana y que los modelos de lenguaje no pueden reemplazar completamente a los participantes humanos en la ciencia psicológica. [2]

- ”TEXTO CLASIFICACIÓN DE AVERÍAS EN EQUIPOS DE A BORDO EN ALTA TRENES DE VELOCIDAD BASADOS EN ETIQUETADO-DOC2VEC Y BIGRU”

El tema trata sobre la categorización de texto de fallos relacionados con vehículos ferroviarios. El objetivo es utilizar técnicas de procesamiento de lenguaje natural para aumentar la eficiencia y precisión de la categorización manual de texto defectuoso. Proporcionan una técnica para lograr este objetivo que implica la creación de incrustaciones de frases usando Labeled Doc2vec, la extracción de características usando una unidad recursiva cerrada bidireccional (BiGRU) y la aplicación de un mecanismo de atención mejorado (IAtt). La técnica sugerida puede categorizar la falla. texto teniendo en cuenta las condiciones de funcionamiento del tren antes y después de la avería y estudiando la semántica del lenguaje de avería. Cuando se compara con otros enfoques, el método sugerido logra el nivel más alto de precisión en la clasificación de textos de fallas, reduce los gastos de mano de obra y aumenta la eficiencia de la clasificación de textos de fallas. [6] describe una técnica para clasificar los mensajes de defectos de los vehículos ferroviarios utilizando Labeled-Doc2vec, BiGRU y un mecanismo de atención mejorado. De acuerdo con los hallazgos experimentales, el texto defectuoso puede clasificarse con un alto grado de eficiencia y precisión.

- ”TOPICSTRIKER: UN ENFOQUE IMPULSADO POR NÚCLEO TÓPICO PARA TEXTO CLASIFICACIÓN"

El artículo describe la aplicación del modelado de temas y los núcleos de cadenas, dos conocidas técnicas de aprendizaje automático y procesamiento del lenguaje natural. El objetivo es proporcionar mejores representaciones de texto para tareas posteriores, incluida la categorización de texto. Mientras que los núcleos de cadenas son operaciones matemáticas en secuencias de texto que permiten construir matrices de características para la clasificación, el modelado de temas se utiliza para descubrir patrones semánticos latentes en vastas colecciones de textos. El método sugerido en el texto combina núcleos de cuerdas supervisados con modelos de temas no supervisados para maximizar sus beneficios. Cuando se utilizan en aplicaciones prácticas como la clasificación de textos, los conjuntos de datos de referencia se utilizan para validar los resultados experimentales. El examen de ambas metodologías, la investigación de las estrategias de aprendizaje automático de próxima generación y la introducción de un método que combina el modelado de temas con núcleos de cadenas para producir representaciones de texto superiores son las contribuciones clave del artículo. [7]

- ”COMPARACIÓN DE MODELOS DE LENGUAJE PREENTRENADOS EN TÉRMINOS DE EMISIONES DE CARBONO, TIEMPO Y PRECISIÓN EN TEXTO DE ETIQUETAS MÚLTIPLES

CLASIFICACIÓN UTILIZANDO AUTOML”

Usando un conjunto de datos de 250K, este estudio evaluó los resultados de varios modelos de lenguaje que habían sido entrenados previamente para la categorización de texto en turco.Empleando un conjunto de datos de 250K para la categorización de texto. El turco es un aglutinante de idiomas con características lingüísticas únicas, incluidas frases repetidas, modismos y metáforas, que dificultan el procesamiento robótico de textos. tener características lingüísticas únicas, como repeticiones, expresiones idiomáticas y metáforas, que dificultan el procesamiento robótico de textos. La evaluación de los modelos de lenguaje ConvBERTurk mC4 y BERTurk (sin cáscara, 128K). Los resultados demostraron que el modelo BERTurk funcionó mejor en términos de precisión en el conjunto de datos, con una duración de entrenamiento de 66 minutos y una emisión de CO2 razonablemente baja. Además, el modelo ConvBERTurk mC4 produjo resultados efectivos. produjo efectos efectivos.

El estudio proporciona más información sobre las capacidades de procesamiento del lenguaje natural turco de los modelos de lenguaje preentrenados y su uso en la clasificación de textos. El uso de modelos preentrenados para la categorización de textos. [8] Este estudio, que se centra en la evaluación de modelos lingüísticos preentrenados para la categorización de textos en turco, establece la viabilidad de los modelos BERTurk y ConvBERTurk. La investigación ofrece datos sobre el rendimiento del procesamiento de textos, la duración del entrenamiento, las emisiones de CO2 y otros parámetros pertinentes en turco.

- ”UN ANÁLISIS COMPARATIVO DE MODELOS BASADOS EN TRANSFORMADORES PARA CLASIFICACIÓN DEL LENGUAJE FIGURATIVO”

El ensayo discute la necesidad de crear técnicas efectivas para extraer varios sentimientos de cantidades significativas de texto. Los investigadores evalúan la eficacia de varios modelos basados en la arquitectura de transformadores para el procesamiento del lenguaje figurativo. El desempeño de los transformadores en lenguaje metafórico se evalúa utilizando un conjunto de datos. El estudio enfatiza lo crucial que es para las computadoras comprender el lenguaje figurado, lo difícil que es clasificarlo y cómo varía del lenguaje literal. Se dan avances en el tema y se comparan los resultados de varios modelos. Los investigadores también investigan cómo estos modelos se generalizan a otras formas comparables de lenguaje figurativo después de haber sido entrenados en un tipo particular. [9] Los investigadores utilizan los modelos basados en topologías de transformadores para evaluar la capacidad de procesamiento del lenguaje figurativo. Para demostrar cómo funcionan los transformadores en el lenguaje figurativo y su capacidad para generalizar a través de diversos tipos de lenguaje figurativo, se comparan los hallazgos de varios modelos y se dan los desarrollos en el área.

Referencias

[1] S. Bhattacharjee, D. Delen, M. Ghasemaghaei, A. Kumar y EW Ngai, “Aplicaciones comerciales y gubernamentales del procesamiento de lenguaje natural (nlp) de minería de texto para beneficio social: Introducción a la edición especial sobre “texto minería de pnl”,” Decision Support Systems, vol. 162, pág. 113867, 11 2022.

[2] D. Dillion, N. Tandon, Y. Gu y K. Gray, “¿Pueden los modelos de lenguaje ai reemplazar los modelos humanos? participantes?”, Trends in Cognitive Sciences, 5 2023.

[3] T. Shaik, X. Tao, C. Dann, H. Xie, Y. Li y L. Galligan, "Análisis de sentimientos y extracción de opiniones sobre datos educativos: una encuesta", Natural Language Processing Journal, vol. 2, pág. 100003, 3 2023.

[4] K. Denecke y D. Reichenpfader, "Análisis de sentimiento de narrativas clínicas: una revisión de alcance", Journal of Biomedical Informatics, vol. 140, pág. 104336, 4 2023.

[5] ER Ling y M. Ekstedt, “Un lenguaje de modelado de amenazas para generar gráficos de ataques de sistemas de automatización de subestaciones”, International Journal of Critical Infrastructure Protection, vol. 41, pág. 100601, 7 2023.

[6] W. Wei y X. Zhao, “Clasificación de texto de fallas del equipo a bordo en el ferrocarril de alta velocidad basado en etiquetado-doc2vec y bigru”, Journal of Rail Transport Planning Management, vol. 26, pág. 100372, 6 2023.

[7] NV Chandran, VS Anoop y S. Asharaf, "Topicstriker: un enfoque basado en núcleos temáticos para la clasificación de textos", Results in Engineering, vol. 17, pág. 100949, 3 de 2023.

[8] P. Savci y B. Das, "Comparación de modelos de lenguaje pre-entrenados en términos de emisiones de carbono, tiempo y precisión en la clasificación de texto de etiquetas múltiples usando automl", Heliyon, vol. 9, pág. e15670, 5 2023.

[9] T. Junaid, D. Sumathi, AN Sasikumar, S. Suthir, J. Manikandan, R. Khilar, PG Kuppusamy y MJ Raju, "Un análisis comparativo de modelos basados en transformadores para la clasificación del lenguaje figurativo", Computers and Electrical Engineering, vol. 101, pág. 108051, 7 2022.