Universidad Nacional Autónoma de México

Facultad de Ciencias

Reconocimiento de Patrones y Aprendizaje Automatizado

TruthFinder

Alex Nakamura Díaz Francés - 421023697

David Hernández Uriostegui - 420003708

Diego Javier Padilla Lara - 420003007

Erick Daniel Arroyo Martínez - 318163790

Sebastián Alamina Ramírez - 318685496

Trabajo presentado como parte del curso de **Reconocimiento de Patrones y Aprendizaje Automatizado**, impartido por el profesor **Sergio Hernández López** durante el semestre 2023-2 en la Facultad de Ciencias, UNAM.

Fecha de entrega: **lunes 5 de junio del 2023**.

Introducción

Las noticias falsas (*fake news*) contienen afirmaciones falsas y/o exageradas que son fácilmente viralizadas por algoritmos, por lo que es importante detectarlas con un grado suficientemente alto de precisión.

Los distintos métodos de clasificación presentes en los algoritmos de *Machine Learning* trabajan con vectores de números como entradas, a veces teniendo la necesidad de ser reducidos en dimensionalidad con el fin de obtener un entrenamiento más eficaz. La elección de la combinación de estos tres factores tiene un gran peso al momento de determinar qué tan exacto es un modelo.

Objetivos

- Con base en nuestro conjunto de datos, comparar diferentes estrategias de preprocesamiento de texto, y su impacto en el desarrollo de estrategias de aprendizaje automatizado.
- Mostrar cuál fue la mejor combinación de preprocesamiento + clasificación.
- Contrastar el desempeño de realizar o no una reducción de dimensionalidad por lograr una mejora en el rendimiento computacional.

Metodología

La metodología utilizada en este proyecto consistió en comparar la eficiencia de diferentes enfoques de vectorización y técnicas de reducción dimensional en la clasificación de textos utilizando modelos de Regresión Logística (LR) y K Vecinos más Cercanos (KNN). Principalmente nos enfocamos en las siguientes etapas:

- Preprocesamiento de datos: Se realizó una etapa de preprocesamiento de los datos, que incluyó la limpieza y transformación de los datos. Esto aseguró que los datos estuvieran en un formato adecuado para el desarrollo de las técnicas empleadas.
- Vectorización: Se aplicaron diferentes técnicas de vectorización, como CountVectorizer, TF-IDF y Word2Vec, para convertir los datos en representaciones numéricas. Estas técnicas capturaron las características y estructura de los datos textuales, permitiendonos un tratamiento integral de los mismos.
- Entrenamiento de modelos (sin reducción dimensional): Se entrenaron modelos de RL y KNN utilizando las representaciones vectoriales obtenidas por medio de CountVectorizer y TFIDF. Para después analizar la eficiencia de los modelos.
- Reducción dimensional: Se aplicó la técnica de reducción dimensional Descomposición en Valores Singulares (SVD), a los vectores obtenidos en la etapa de vectorización. Esto permitió disminuir la dimensionalidad de los datos y extraer características relevantes con el propósito de mejorar potencialmente la eficiencia y capacidad de generalización de los modelos. Se utilizó esta estrategia sobre PCA, que fue estudiado en clase, porque funciona mejor con matrices dispersas.
- Entrenamiento de modelos con reducción dimensional: Se entrenaron nuevamente los modelos de RL y KNN utilizando los vectores reducidos obtenidos en la etapa de reducción dimensional. Esto permitió comparar el rendimiento de los modelos después de aplicar la aplicación de SVD.
- Desarollo de RL con composición de vectorizaciones: Se implemento una composición de vectorizaciones (CountVectorizer y Word2Vec) para poder determinar la eficiencia de estas técnicas de vectorización combinadas sin relación de orden, pues ambas se calcularon basados en el mismo conjunto de datos.

- Evaluación y comparación de resultados: Se evaluó el rendimiento de los modelos utilizando métricas apropiadas para la clasificación, como la precisión, el recall y la puntuación F1. Se compararon los resultados obtenidos utilizando diferentes enfoques de vectorización y reducción dimensional para determinar cuál fue el más eficiente en términos de precisión y capacidad de generalización.

Resultados

Regresión Logística

Con regresión logística, se probaron todas las combinaciones de métodos de vectorización y reducción de dimensiones.

LR - TFIDF

Se entrenó este modelo utilizando la vectorización obtenida por TFIDF tal cual que contó con 86562 columnas (el número de palabras diferentes conservadas después de la limpieza). El modelo entrenado tuvo un desempeño de 0.98 en las métricas de precisión, recall y puntuación F1.

LR - CountVectorizer

Se entrenó este modelo utilizando la vectorización obtenida por Count tal cual que, al igual que la vectorización por TFIDF, contó con 86562 columnas. El modelo entrenado tuvo un desempeño de 0.99 en las métricas de precisión, recall y puntuación F1, mejor que el modelo entrenado con la vectorización TFIDF.

LR - TFIDF - SVD

Con tal de revisar si se volvería menos asertivo el modelo con una reducción de dimensiones, se aplicó Regresión Logística a los vectores generados por TFIDF después de haber sido reducidos dimensionalmente de 86562 entradas a 800 (se eligió 800 porque números mayores tardaban mucho en ejecutarse). Las métricas de evaluación no cambiaron radicalmente con respecto del modelo sin reducción de dimensiones.

LR - CountVectorizer - SVD

Como en el punto anterior, se aplicó Regresión Logística a los vectores generados por CountVectorizer después de haber sido reducidos dimensionalmente de 86562 entradas a 800). Las métricas de evaluación no cambiaron radicalmente con respecto del modelo sin reducción de dimensiones.

LR - CountVectorizer - Word2Vec

El entrenamiento de este modelo utilizando la combinación de vectorizaciones como CountVectorizer y Word2Vec puede tener un impacto significativo en el rendimiento y las capacidades del modelo. Ese fue el supuesto para desarrollarlo, tras obtener las vectorizaciones independientes, se procedió a componerlas para obtener una nueva representación de los datos, obteniendo valores en las métricas de evaluación del modelo entrenado, como: precisión = 1, recall=1 y la puntuación F1=1.

LR - CountVectorizer - Word2Vec - SVD

En este caso, dado que el número de características obtenidas por Word2Vec fueron cien, entonces precedimos a aplicar la reducción dimensional a la matriz dispersa obtenida por Count, de tal forma que redujimos a cien caracteríticas. Obteniendo como resultado una representación vectorial de doscientas. En la evaluación del modelo entrenado con dicha vectorización reducida compuesta obtuvimos una ligera disminución en las métricas de evaluación anteriores: precisión = 0.98, recall=0.98 y la puntuación F1=0.98.

K Nearest Neighbors

Algo a tener en consideración es el hecho de que al ser KNN un algoritmo ineficiente en vectores grandes no se realizaron experimentos sin aplicar reducción de dimensiones, esto debido al alto coste computacional que supondría no hacerlo.

KNN - TFIDF - SVD

Este modelo, entrenado con la vectorización obtenida por `TFDI` trás aplicar una reducción dimensional a 800 entradas (valor escogido arbitrariamente), mostró un desempeño subpar si lo comparamos con cualquiera de los modelos que hacen uso de LR.

KNN - CountVectorizer - SVD

Por otro lado, el modelo entrenado con `CountVectorizer` y reducción dimensional, mostró un incremento en sus valores para las métricas de evaluación (precisión, recall y F1) respecto a lo arrojado por el modelo anterior.

Experimentando con el hiperparámetro k

A continuación se presentó un enfoque mejorado para realizar una comparación más precisa, el cual consistió en probar con diferentes valores de k seleccionados de un conjunto : $\{75, 100, 150, 200, 225, 250, 300, 350, 400, 500, 600, 700\}$. Este procedimiento se llevó a cabo tanto para las técnicas `CountVectorizer - SVD` como para `TFIDF - SVD`.

En ambos casos, se observó que a medida que aumentaba el valor de k, las métricas utilizadas para evaluar la calidad de nuestro modelo disminuían drásticamente. No obstante, se destacó una diferencia significativa entre los dos enfoques. En el caso de `CountVectorizer - SVD`, se observó que al aumentar k, el *recall* para las clasificaciones verdaderas (**True**) se mantuvo constante en 0.99, mientras que la precisión para las clasificaciones falsas (**Fake**) se mantuvo en un rango de $0.89 - 0.92$.

Conclusiones

El objetivo principal de este proyecto fue explorar el impacto de la selección adecuada de un conjunto de datos y la aplicación de diversas técnicas de preprocesamiento, vectorización y reducción dimensional en la eficiencia de los modelos de regresión logística y k vecinos más cercanos.

Las técnicas de preprocesamiento de datos, como la limpieza, tokenización y lematización, desempeñan un papel esencial al eliminar inconsistencias en los datos y reducir la dimensionalidad, mejorando así su calidad y coherencia.

Por otro lado, las técnicas de vectorización son utilizadas para obtener una representación numérica de los datos. Estas técnicas permiten capturar tanto la frecuencia de aparición de las palabras como su importancia relativa en el contexto del conjunto de datos.

La reducción dimensional es una técnica que ofrece beneficios y riesgos. Por un lado, reduce la dimensionalidad de los datos y proporciona nuevas representaciones de las características, lo que ayuda a evitar el sobreajuste de los modelos. Sin embargo, también existe el riesgo de perder información importante durante este proceso. En los resultados obtenidos, se observó que la combinación de técnicas de vectorización, como CountVectorizer y Word2Vec, permite obtener características más ricas en los datos. Estas técnicas no solo capturan la frecuencia e importancia de las palabras, sino que también consideran las relaciones semánticas y sintácticas entre ellas.

En conclusión, los resultados del proyecto destacan la importancia de una selección adecuada del conjunto de datos, la aplicación consistente de técnicas de preprocesamiento y la correcta elección de técnicas de vectorización que se ajusten a la naturaleza del problema. La reducción dimensional, aunque útil para disminuir el costo computacional y mejorar las representaciones de los datos, debe aplicarse con precaución para evitar la pérdida de información relevante. Además, se evidenció que la combinación de las técnicas de vectorización CountVectorizer y Word2Vec mejora la precisión y generalización del modelo de regresión logística. En contraste con la aplicación de SVD a la representación CountVectorizer para el mismo modelo, tal que, esta resultó en una ligera disminución de las métricas de evaluación, debido a la pérdida de varianza, lo que influyo negativamente el rendimiento del modelo.