Predicción de Noticias Falsas en EE.UU.

Fernando Fetis
Dpto. de Ingeniería Matemática
Universidad de Chile, FCFM

Valentina Gómez
Dpto. de Ingeniería Matemática
Universidad de Chile, FCFM

Francisco Muñoz
Dpto. de Ingeniería Matemática
Universidad de Chile, FCFM

***Abstract*—En este trabajo se estudiarán formas de clasificar una noticia como verdadera o falsa, en base al titular y contenido de la noticia.**

La forma en que se realizará la clasificación será utilizando herramientas de Aprendizaje de Máquinas, como lo son el Procesamiento de Lenguaje Natural, y algoritmos de clasificación, como lo es Naïve Bayes, Regresión Logística, el Perceptrón, entre otros.

Se concluirá que, a partir del solo contenido de una noticia, es posible clasificarla como verdadera o falsa, con más del 99% de precisión. Además, se concluirá que basta únicamente el titular de la noticia para determinar si una noticia es verdadera o falsa con apróximadamente un 90% de precisión.

***Index Terms*—Classification, Natural Processing Language, Fake News**

I. Introducción

En la actualidad, los usuarios del internet se encuentran constantemente bombardeados de información y noticias, proveniente de fuentes que parecen diversificarse día a día.

Es por ello, que no siempre se puede identificar con certeza cuando una noticia es verdadera o falsa. Lo cual cobra importancia, ya que las noticias son la forma principal en que las personas pueden entender el contexto en que están viviendo y formar sus opiniones. Además de que las noticias falsas son una forma maliciosa de propagar desinformación para manejar la opinión pública en beneficio de sus creadores.

Es por esto que el objetivo de este trabajo será el de desarrollar una herramienta que permita **clasificar** una noticia como verdadera o falsa, en base al título y cuerpo de la misma.

Para esto, se utilizarán diversas herramientas clásicas de Aprendizaje de Máquinas, como lo es Procesamiento de Lenguaje Natural y Clasificación Binaria. En específico se usará vectorización, análisis léxico y lematización sintáctica para procesar el texto, y luego modelos como *Naïve Bayes*, *Regresión Logística*, *Perceptrón* y *Support Vector Machine* para la clasificación.

La presentación del proyecto se puede observar en el link que se adjunta a continuación[1].

II. Marco Teórico

El procesamiento del lenguaje natural investiga el uso de computadores para procesar o entender el lenguaje humano (natural) con el objetivo de realizar tareas útiles. Dentro de las aplicaciones más comunes están: reconocimiento de voz, interpretación del lenguaje hablado (SLU por sus siglas en inglés), sistemas de diálogos, análisis léxico, análisis gramatical, análisis sintáctico, análisis morfológico, traducciones, recuperación de texto, análisis de sentimientos, etc.

Una de las herramientas más importantes en el procesamiento de textos, es la **vectorización**. Ya que permite a la máquina entender el contenido simbólico a través de representaciones numéricas significativas. En el caso de la técnica usada, construye una matriz para registrar la frecuencia de cada componente léxico observado.

Sin embargo, el trabajo requerido para convertir el texto natural en componentes léxicos que puedan aportar significado la matriz no es trivial. Se ocupan técnicas de análisis léxico, análisis sintáctico y lematización.

- El **análisis léxico** del texto, corresponde a dividir oraciones o párrafos en componentes léxicos o símbolos, para así poder visualizar individualmente las palabras o símbolos que componen una secuencia de texto.
- Luego, la **lematización** es el proceso mediante el cual las palabras de un texto que pertenecen a un mismo paradigma flexivo o derivativo son llevadas a una forma normal que representa a toda la clase. Esta forma normal, llamada lema, es típicamente la palabra utilizada como entrada en los diccionarios de lengua.
- En la lematización usada, se necesita distinguir si la palabra es un verbo, sustantivo, adjetivo o adverbio, para llevar a su raíz más exacta. Dicho proceso es el **análisis sintáctico**.

Así, cada componente léxico corresponde a un lema. Lo cual permite agrupar de mejor manera las palabras por significado y otorga mayor importancia a cada representación en la vectorización.

Finalmente, cabe mencionar el concepto de **stopwords**. Este concepto se refiere a las palabras que no tienen un significado por si solas, sino que modifican o acompañan a otras. Este grupo de palabras usualmente está conformado por los artículos, pronombres, preposiciones, adverbios e incluso algunos verbos.

III. Descripción de los Datos

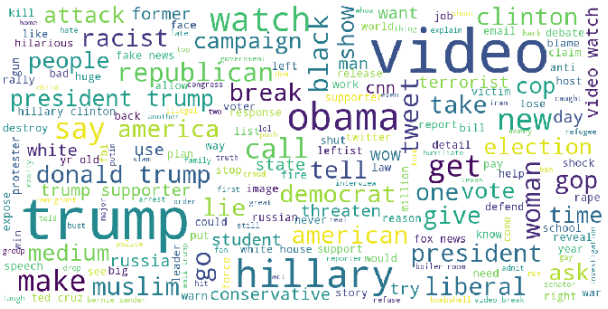
A. Descripción del dataset

La base de datos se obtuvieron a partir de Kaggle, del *dataset* llamado "Fake and real news dataset"[2].

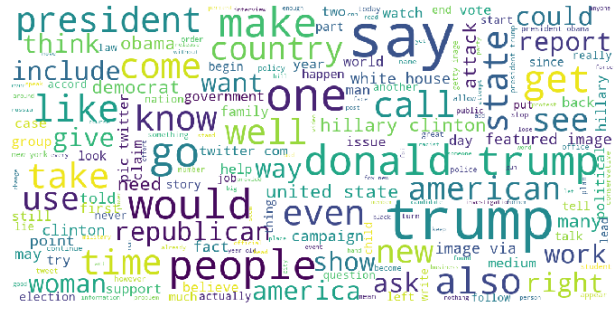
Esta base de datos consiste en dos *datasets*: el de noticias falsas (17.903 datos) y el de noticias verdaderas (20.826

[1] https://youtu.be/5OHAfpI1F2I

[2] https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset

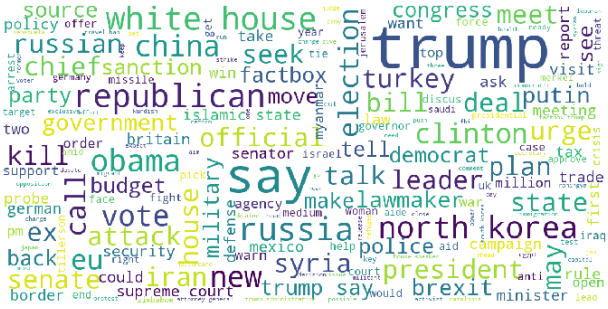


(a) Titulares de noticias falsas

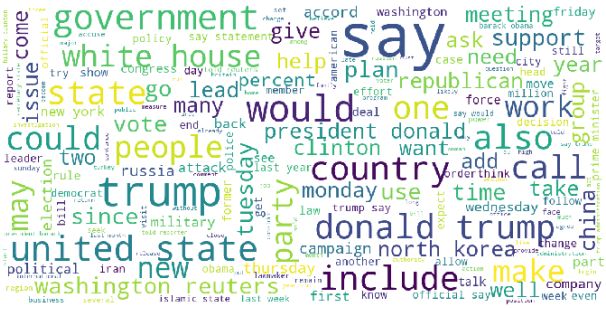


(b) Cuerpo de noticias falsas

Fig. 1: Nube de palabras para las noticias falsas.



(a) Titulares de noticias verdaderas



(b) Cuerpo de noticias verdaderas

Fig. 2: Nube de palabras para las noticias verdaderas.

datos). Ambos datasets comparten las mismas columnas: *title*, *text*, *subject* y *date*. Estas corresponden al titular de la noticia, al contenido de la noticia, al tipo de noticia que es y la fecha de la noticia, respectivamente.

Las columnas *title* y *text* son de tipo cadena; *subject* es un tipo de dato categórico y *date* es un tipo de dato temporal.

En el *dataset* de las noticias verdaderas, existen dos etiquetas para la columna *subject*, las cuales son **politicsNews** y **worldnews**. La etiqueta **politicsNews** posee 11.272 datos y es para referirse a las noticias que tengan una connotación política, mientras que la etiqueta **worldnews** posee 10.145 datos y es para referirse a las noticias que ocurren fuera de EE.UU. Mientras que el *dataset* de las noticias falsas, posee seis etiquetas: **News**, **politics**, **Government News**, **left-news**, **US news** y **Middle-east**. A continuación se dará una breve descripción de cada categoría:

- La etiqueta de **News** posee 9.050 datos y corresponde a las noticias estándar,
- La de **politics** posee 6.836 datos y corresponde a noticias de connotación política,
- La de **Government News** posee 1.568 datos y corresponde a noticias sobre el gobierno,
- La de **left-news** posee 4.456 datos y corresponde a noticias del espectro de la izquierda política,
- La de **US News** posee 784 datos y corresponde a noticias sobre EE.UU., y finalmente
- La de **Middle-east** posee 778 datos y corresponde a noticias del Oriente Medio.

El *dataset* originalmente no posee valores nulos. Sin embargo, el *dataset* de las noticias falsas posee 10 datos etiquetados como fecha, a pesar de que no lo sean.

B. Tratamiento de los Datos

1) Creación de Columnas y Limpieza: Se crea una nueva columna en ambos *datasets*, llamada *true*, el cual determina si la noticia es verdadera o falsa. Se le asigna a todas las noticias verdaderos, el valor de true igual a 1, y 0 a las noticias falsas. Paso siguiente, se concatenan ambos *datasets*.

C. Análisis Exploratorio de los Datos

Además, se marcan los datos que no sean fechas de la columna *date* por el tipo de dato llamado *Not a Time*, y se eliminan los datos nulos que posea el *dataset*.

1) Procesamiento de los datos: Luego, se procede a procesar la información de tipo texto (esto es, para las columnas *title* y *text*) a través de NLP. Se escoge una muestra aleatoria de 2500 filas[3] de los datasets originales (para las noticias verdaderas y falsas) y se empieza transformando el texto de las cadenas en minúsculas.

Luego se realiza un análisis léxico (a.k.a. *tokenization*), y por cada componente léxica (a.k.a. *token*), si es que no corresponde a una *stopword*, se lematiza la componente léxica. Realizada la lematización de cada componente léxica, se procede a unir nuevamente todas las componentes en un único texto de tipo cadena.

Paso seguido, este nuevo texto generado se reemplaza en una copia del *dataset*, en el lugar dónde estaba la información no procesada originalmente.

[3]Se realiza esto pues, si se escogen más filas, la memoria para guardar la información en el computador queda sobrecargada a la hora de vectorizar los textos.

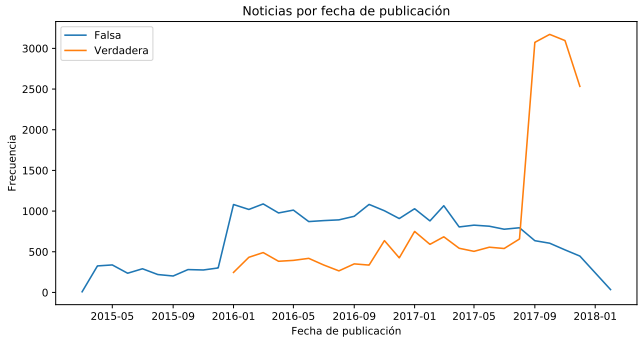


Fig. 3: Frecuencia de noticias por fecha de publicación

Finalmente, se empieza a codificar la información a través de un vector de características.

En las Figuras 1a y 1b se observan las nubes de palabras realizadas a partir de los titulares y el cuerpo de las noticias falsas. A partir de ellas, se concluye que las temáticas principales de las noticias falsas son: videos, Trump, Obama, Hillary, ataque, racista, presidente, mujer, republicanos, demócratas, musulmán, blanco, negro, terrorista, mentira, etc. Que corresponden a figuras políticas y temas sensibles de discriminación.

Mientras que en las Figuras 2a y 2b, las cuales corresponden a las nubes de palabras de las noticias verdaderas, las palabras mayoritarias son: Trump, Casa blanca, Rusia, China, Obama, gobierno, EE.UU., Corea del Norte, Turquía, elección, muerte, voto, Clinton, lider, etc. Es decir, son noticias orientadas a las relaciones internacionales, y no abordan temas tan polémicos como las noticias falsas.

En la Fig. 3 se puede observar la distribución de las frecuencias de las noticias por su fecha de publicación. A partir de esta figura, se puede observar que el periodo en que se recolectaron las noticias verdaderas data desde los inicios del 2016, hasta finales del 2017. Además, que las noticias falsas fueron recolectadas desde inicios del 2015, hasta principios del 2018.

Y como última observación acerca de la Fig. 3, se observa que, mientras que las noticias falsas tuvieron una pequeña alza en inicios del año 2016 (la cual mantuvo), las noticias verdaderas tuvieron una enorme alza en el último cuatrimestre del año 2017. Esto último se puede deber a que en esos años sucedió el cambio de gabinete del actual presidente de EE.UU., Donald Trump.

IV. Metodología

A. Modelos

Se escogerán cuatro modelos: *Naïve Bayes* (NB), por su simpleza, *Regresión Logística* (RL), el *Perceptrón*, ambos modelos por su capacidad de predecir sin utilizar técnicas geométricas, y *Support Vector Classifier* (SVC), por su versatilidad en separar los datos de forma geométrica.

1) Preparación del modelo de NB: Se escogerá un prior uniforme para Naïve Bayes.

TABLA I: Tabla de Resumen

	title	text	title & text
Regresion Logistica	89.13	**99.39**	98.92
Naive Bayes	84.67	83.59	84.40
Perceptron	88.99	98.38	97.70
SVC	88.93	98.18	98.24

2) Preparación de los modelos de RL, Perceptrón y SVC: Se utilizarán los hiperparámetros que vienen por defecto en *sklearn*.

B. Preparación del Conjunto de Entrenamiento y Test

Procesada la información del texto, se escoge una columna para realizar la predicción. En nuestro caso, se escogen las columnas *title*, *text* (ambas por separado) y una concatenación de *title* y *text*.

Fijada una columna, se procede a dividir de forma aleatoria un 70% de los datos para el entrenamiento, y un 30% de los datos para el test.

C. Entrenamiento de los modelos

Por cada uno de los escenarios que se habló anteriormente (*title*, *text* y *title & text*) se utilizan todos los modelos (NB, RL, Perceptrón y SVC), en dónde se obtendrán cada una de las precisiones de los modelos.

La implementación de lo anteriormente comentado se puede encontrar en el siguiente repositorio[4].

V. Resultados y Análisis

En la Tabla I se puede observar el porcentaje de precisión que resultó después de utilizar los cuatro modelos en cada uno de los escenarios planteados.

Observamos a partir de la Tabla I que, ocupando el modelo de *Regresión Logística* sobre la columna de *text* (procesada) es la que resulta con la mayor precisión entre todas combinaciones. Además de que en la mayoría de los modelos (exceptuando NB) la columna *text* resultó ser la que mayor precisión obtuvo.

Sin embargo, es destacable que ocupando únicamente el titular de la noticia, todos los modelos tengan aproximadamente un 90% de precisión a la hora de predecir la veracidad de la misma. Lo cuál significa que, basta con considerar el titular de una noticia para determinar con bastante precisión si ésta es una noticia verdadera o falsa.

Además, es destacable que utilizando el titular de la noticia, junto con el texto de la misma, termina empeorando la calidad de la predicción en los modelos de RL y Perceptrón. Esto resulta contra intuitivo, pues se espera que añadiendo "más información" que puede resultar útil, la predicción mejore, sin embargo, terminó perjudicando la calidad de predicción.

[4] https://github.com/asolnn2a8/Proyecto-ML

VI. Conclusión

Observamos que la división entre el conjunto de entrenamiento y de test no es la mejor forma de particionar los datos. Más que nada por el siguiente hecho: los datos son dependientes del tiempo.

Esta dependencia temporal es lo que obliga a considerar el escenario de "se están ocupando los datos del pasado para predecir el futuro", con lo que una forma óptima de particionar los datos seria la siguiente: ordenar los datos por fecha, luego dividir los primeros $x\%$ de los datos para el conjunto de entrenamiento y los restantes $(100 - x)\%$ como conjunto de test.

Es posible que los resultados de la predicción hayan mejorado bastante porque el modelo sufrió de un *overfitting* a causa de no considerar las limitaciones temporales.

También, a partir de la Fig. 3, es posible que sea conveniente el de considerar la intersección de las fechas de las noticias verdaderas y falsas, pues las noticias varían bastante entre un año y otro.

Referencias

[1] Ahmed H, Traore I, Saad S. "Detecting opinion spams and fake news using text classification", Journal of Security and Privacy, Volume 1, Issue 1, Wiley, January/February 2018.

[2] Ahmed H, Traore I, Saad S. (2017) "Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques". In: Traore I., Woungang I., Awad A. (eds) Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments. ISDDC 2017. Lecture Notes in Computer Science, vol 10618. Springer, Cham (pp. 127-138).

[3] Bassi., A. (2000). Lematización basada en análisis no supervisado de corpus (1). Departamento de Ciencias de la Computación Universidad de Chile. Recuperado de https://users.dcc.uchile.cl/~abassi/ecos/lema.html

[4] Deng, L., & Liu, Y. (2018). Deep Learning in Natural Language Processing. New York, Estados Unidos: Springer Publishing.

[5] Singh, A. K., & Shashi, M. (2019). Vectorization of Text Documents for Identifying Unifiable News Articles. International Journal of Advanced Computer Science and Applications, 10(7), 305-310. https://doi.org/10.14569/ijacsa.2019.0100742