Predicción de Tweets

"Una rosa con cualquier otro nombre olería igual de dulce" Julieta Capuleto

Diana Yarith Higuera Cogua

Juan Sebastian Vallejo Triana

Lizbeth Karina Hernandez Noriega

Riky Andrés Carrillo Cadena

Problem Set 4

Universidad de los Andes
Facultad de Economía
Bogotá, Colombia
20 de marzo 2023

Introducción

En la sociedad actual cada vez más las personas están siendo receptoras y productoras de información, por ejemplo, las redes sociales son hoy en día un medio de comunicación y difusión importante para las personas.La masificación de las redes sociales ha permitido que cada persona pueda comunicar sus mensajes sin más barreras que la que el dispositivo que tienen le permita y aun así encuentre audiencia para su mensaje; recientemente esta tendencia ha sido utilizada en el sector político, donde los hacedores de políticas públicas han encontrado en las plataformas virtuales para transmitir su posición y ejercer política a través de ellos.

En tiempos de redes sociales, los mensajes se acomodan a los gustos de cada usuario. Las nuevas formas de comunicación filtradas por algoritmos generan un desafío a la política. ¿Los candidatos deberían decirle a cada uno lo que quiere escuchar? (Magnani, 2017)

En el presente Problem Set, se trabajará con una base de datos de entrenamiento conformadas por 9349 observaciones o tweets distribuidas en proporciones similares y una base de prueba compuesta por 500 tweets correspondientes a trinos de tres políticos colombianos y de las cuales se pretende predecir cada uno de ellos a qué personaje corresponde.

Para ello se utilizarán herramientas de aprendizaje profundo de machinne learning, en este caso, las redes neurales que permiten generar interconexiones entre los nodos de información generando capas ocultas para procesamiento de datos y posterior predicción de resultados, simulando de alguna manera el comportamiento del cerebro humano, de allí su nombre.

Datos

La información empleada para este estudio, se obtuvo de https://www.kaggle.com/competitions/uniandes-bdml-ps4/data y se compone de dos bases de datos: 1) Entrenamiento y 2) Prueba. La base de datos se va a utilizar para predecir la pertenencia de un conjunto de tweets de tres destacados políticos colombianos: Claudia López, Gustavo Petro y Álvaro Uribe.

Las variables que componen inicialmente las bases de datos corresponde a tres variables:

1. **id**:identificador del tweet
2. **autor**:nombre del autor del tweet
3. **tweet**:contenido textual del tweet

En general, la base de datos de entrenamiento tiene 9349 observaciones y 3 variables (id, autor, tweet), mientras que la base de datos de prueba contiene 1500 observaciones y dos variables (id y tweet). Nota: El conjunto de datos de prueba no contiene la variable de autor, dado que nuestro objetivo es predecir el respectivo autor de cada uno de los tweet de la base de prueba.

Limpieza y manipulación de las bases de datos

Teniendo en cuenta que tenemos dos bases, una de entrenamiento y otra de prueba, iniciamos realizando una inspección a la base de datos de entrenamiento, sobre la relación porcentual de tweets por autor, que se muestra en el Cuadro 1. Podemos observar que el 37.12 % de los tweet en la base de datos de entrenamiento pertenecen a Claudia López, mientras que el 30.77 % pertenecen a Gustavo Petro y el 32.11 % son de Álvaro Uribe.

tanto en la base de entrenamiento como de prueba la variable de superficie total y cubierta tiene un porcentaje alto de valores perdido y ronda el 80 % de las observaciones,

en el caso de las habitaciones y los baños el porcentaje de valores perdidos ronda el 40 % y 25 % de las observaciones, en el caso de las variables de título y descripción el porcentaje de observaciones perdidas es bajo.

Cuadro 1. Porcentajes de tweets en la base de datos de entrenamiento

Autor	**% de tweet**
Claudia Lopez	37.12
Gustavo Petro	30.77
Álvaro Uribe	32.11
Total	**100**

Seguidamente, procedemos a limpiar la variable *tweet*, para esto colocamos todo el texto del tweet en minúscula, se eliminan tildes y caracteres especiales del español, símbolos de puntuación, números y acentos, además de eliminar múltiples espacios en blanco. Posteriormente, se transforma la base de datos de entrenamiento a nivel de palabras, previa creación de un, **id** para cada tweet. Luego se realiza la eliminación de *stopwords* que son las palabras comunes del lenguaje natural que se eliminan por considerarse que tiene poco significado semántico. Para este procedimiento hacemos uso de una lista de *stopwords* en español procedente de tres fuentes distintas: *snowball*, *ntlk* y *stopwords-iso*

Antes de realizar el procedimiento anteriormente descrito, se contabilizó el número de palabras en la variable *tweet* de la base de datos de entrenamiento antes y después de la eliminación de *stopwords*, como se ve en el cuadro 2, había 290,298 palabras y después de la eliminación de los *stopwords* quedaron 139,097 palabras contenidas en la variable *tweet*.

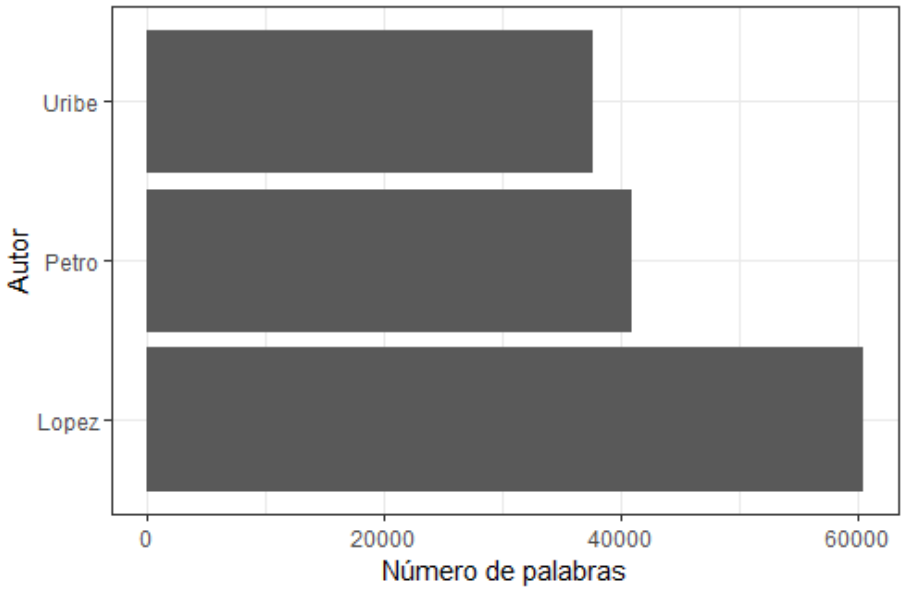
Cuadro 2. Conteo de *stopwords* antes y después de la eliminación dentro de la variable tweet en la base de datos de entrenamiento

Stopwords	**Antes**	**Despues**
Número de palabras	290,298	139,097

Análisis descriptivo

Iniciamos mostrando el número de palabras utilizadas por cada autor. La Figura 1 muestra que la alcaldesa de la ciudad de Bogotá, Claudia López es la persona con mayor número de palabras dentro de la base de datos de entrenamiento con, 60490 palabras, seguidas del presidente de Colombia, Gustavo Petro con, 40889 palabras y finalmente el ex-presidente Álvaro Uribe con 37718 palabras.

Figura 1. Número de palabras utilizadas por cada autor en los tweets



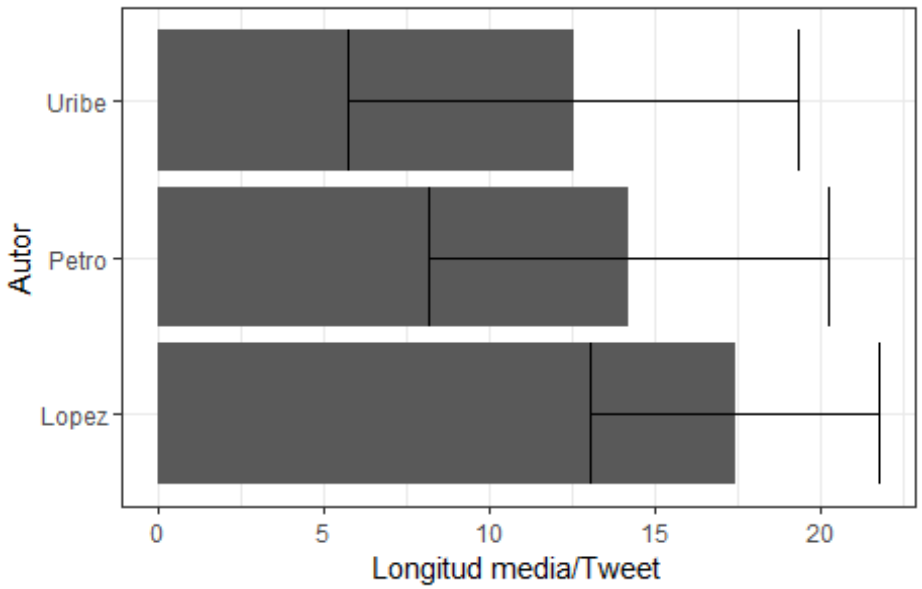
El cuadro 3 muestra la media y desviación estándar de la longitud, los tweet de los autores, se puede ver como la alcaldesa de Bogotá, Claudia López es la autora de los tweet con mayor longitud en promedio dentro de la base de entrenamiento con 17.4 palabras y con menor desviación, seguida del Presidente de la República Gustavo Petro con 14.2 palabras y finalmente el ex-presidente Álvaro Uribe con unos tweets de longitud media de 12.6 palabras.Los tweets de Uribe y Petro son similares en longitud media y desviación, mostrando esta última que suelen alternar entre tweets largos y cortos.

Cuadro 3. Estadísticas media y SD de la longitud de los tweets por autor

Autor	**Longitud media tweet**	**SD longitud tweet**
Claudia Lopez	17.4	4.37
Gustavo Petro	14.2	6.02
Álvaro Uribe	12.6	6.80

La Figura 2 observamos la distribución media de los tweets por autores, donde se nota la mayor longitud hacia los tweet de la Alcaldesa de Bogotá, mientras que el presidente Gustavo Petro y el ex-presidente Álvaro Uribe tiene tweet muy

Figura 2. Longitud media de los tweets por autor



Hacemos una visualización de las nubes de palabras de cada uno de los autores. En el caso de la alcaldesa de Bogota Claudia López se puede apreciar en la Figura 3 la frecuencia de palabras como los, las, gracias, mujeres, jóvenes, todos, pandemia, semana, entre otras.

Figura 3. Nube de palabras para Claudia Lopez

En el caso del Presidente de la República, Gustavo Petro, la nube de palabras que se muestra en la Figura 4, está conformada por palabras como: Colombia, humana, economía, las, los, gobierno, pueblo, pacto, jóvenes, histórico, entre otros.

Figura 4. Nube de palabras para Gustavo Petro

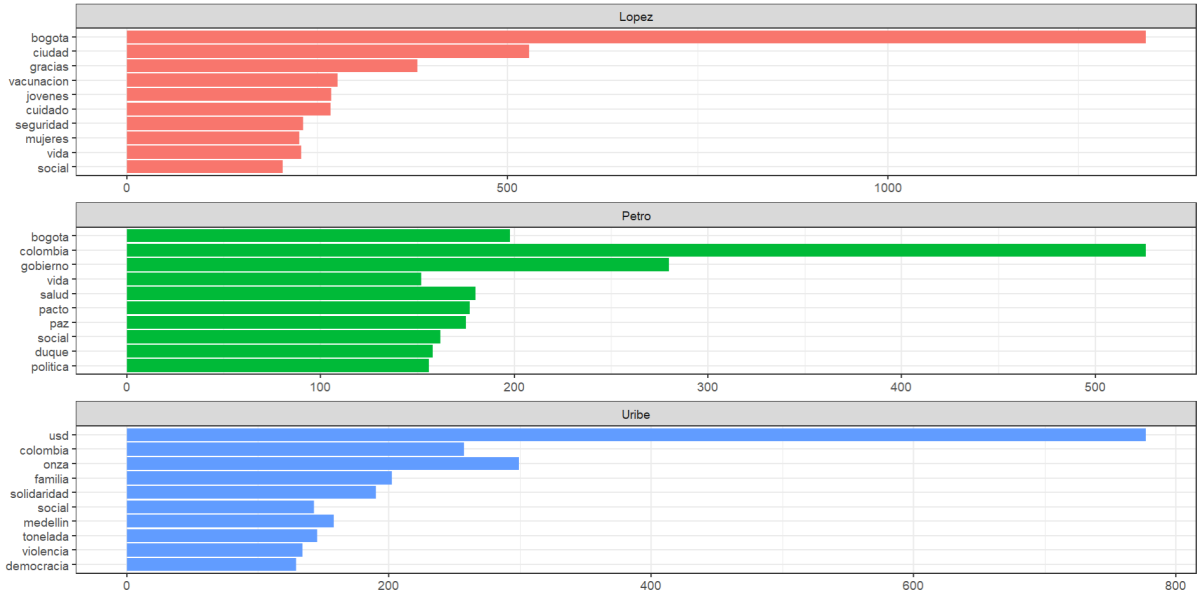
En el caso del ex-Presidente de la República Álvaro Uribe, la nube de palabras que se muestra en la Figura 5, está conformada por palabras como: Colombia, familia, autoridad,

democracia, fuerza, economía, seguridad, entre otros.

Figura 5. Nube de palabras para Gustavo Petro

Igualmente, incluimos la visualización de frecuencias de palabras más utilizadas por cada uno de los autores que se muestra en la Figura 6, en el caso de la alcaldesa de Bogotá, Claudia López, destaca palabras como: Bogotá, ciudad, gracias, vacunación, jóvenes y cuidado, siendo Bogotá la palabra más utilizada. En el caso del presidente Gustavo Petro sus palabras más utilizadas son: Bogotá,Colombia, gobierno, vida, salud, pacto y paz, siendo colombia y gobierno las más utilizadas, mientras que el ex-presidente Alvaro Uribe utiliza con frecuencias en sus tweet las palabras: usd, colombia, familia, solidaridad, medellin, tonelada, violencia y democracia.

Figura 6. Frecuencias de palabras más utilizadas por el autor



Modelos y Resultados

Modelos de clasificación

Para la clasificación de los tweets, se dividió la base de entrenamiento suministrada en dos submuestras de entrenamiento y testeo. Esto con dos objetivos. Por un lado, la submuestra de entrenamiento se utilizó, por medio de validación cruzada tipo K Fold con K igual a 10, para comparar las diferentes combinaciones de modelos propuestos y para hacer la optimización de los parámetros de los mejores modelos derivados de la comparación. Por otro lado, la base de testeo sirvió como último filtro de evaluación, permitiendo calcular la precisión de los modelos no solo por medio de los K Folds, sino con una última base de testeo que no se vio involucrada en el ajuste del modelo. Dicha división se hizo por medio de la estratificación de la variable dependiente a predecir (variable categórica con los nombres de los autores de los tweets). Lo anterior con el objetivo de mantener una misma proporción de dicha variable en las submuestras generadas.

Una vez definidas la base de entrenamiento, evaluación y validación cruzada, se se probaron dos abordajes. En primer lugar, se construyó una especificación simple compuesta de variables creadas a partir de la variable de texto original suministrada de la forma:

$$\begin{aligned} Nombre_i = &+\beta_{i1} No.Caracteres + \beta_{i2} No.Menciones + \beta_{i3} No.Hashtags + \\ &= \beta_{i4} Desviación.Caracteres + \beta_{i5} Emoji + \beta_{i6} Exclamación + \\ &= \beta_{i7} Tema.Tf.Idf + \beta_{i8} Tema.Menciones + \beta_{i9} Tema.Frecuencia + \\ &= \beta_{i10} Bogota + \beta_{i11} Macro + \beta_{i12} Familia + \beta_{i13} No.Sustantivos + \\ &= \beta_{i14} No.Adjs + \beta_{i15} No.Verbos + \beta_{i16} No.Otros + \epsilon i \end{aligned} \tag{1}$$

Donde la variable *Nombre* es nuestra variable dependiente, la cual contiene el nombre del político que escribió dicho tweet; No.Caracteres contiene el número de caracteres de cada tweet como medida de longitud; No.Menciones contiene el número de usuarios mencionados en el Tweet; *No.Hashtags* contiene el número de hashtags utilizados en el tweet; *Desviación.Caracteres* es una medida de dispersión, la cual mide qué tan alejado está la longitud del Tweet frente al promedio; *Emoji* es una variable binaria que toma el valor de 1 si el tweet contiene emojis y 0 de lo contrario; *Exclamación* es una variable binaria que toma el valor de 1 si el tweet contiene signos de exclamación y 0 de lo contrario; *Tema.Tf.Idf* es una variable categórica que toma el valor de los nombres de los políticos a clasificar, en función de si el tweet contiene al menos una de las palabras con mayor Tf-Idf del político.[1]; *Tema.Frecuencia* y *Tema.Menciones* mantienen la misma lógica, solo que las categorías se hacen en función de las palabras y menciones más frecuentes de cada político respectivamente. Las variables binarias de *Bogota*, *Macro* y *Familia*, son variables que toman el valor

[1]El tf idf se calcula para cada uno de los políticos, suponiendo que cada político es un documento. Esto nos permite identificar las palabras más relevantes en función de qué tanto la repite un político relativo a los demás.

de 1 si el tweet contiene esa palabra y cero de lo contrario. Estas variables se incluyeron por su potencial de discriminación identificado en las estadísticas descriptivas. Mientras que la palabra ciudad y Bogota son palabras usadas significativamente más por Claudia Lopez, por su rol como alcadesa de Bogotá, relativo a los otros políticos, palabras como onza y usd son palabras que Alvaro Uribe utiliza más que los demás debido a sus tweets informativos de variable macroeconómicas. Por último, se incluyen una seria de variables continuas con el número de sustantivos, adjetivos, verbos y otros tipos de palabras presentes en el Tweet, las cuales corresponden a las últimas 4 variables de la ecuación 3 respectivamente. Para la especificación anterior, se combinaron 3 recetas[2] y dos metodologías de estimación. La primera receta mantiene el modelo original como referencia. La segunda receta normaliza las variables continuas y le aplica un procedimiento de agrandamiento de muestra via SMOTE para corregir por el leve desbalance presente en la variable de clasificación (Más Tweets de Claudia, seguido de Petro y Uribe). La tercera receta mantiene los preprocesamientos de la segunda y le agrega predictores adicionales derivados de los componentes principales que en su conjunto capturan el 90 % de la variación original del modelo. Con respecto a las metodologías de estimación, se evaluó una metodología de bosques aleatorios y un Elastic Net Multinomial. Junto con las 3 recetas expuestas anteriormente, se evaluaron 6 modelos derivados de la combinación de cada una de las recetas con cada una de las metodologías.

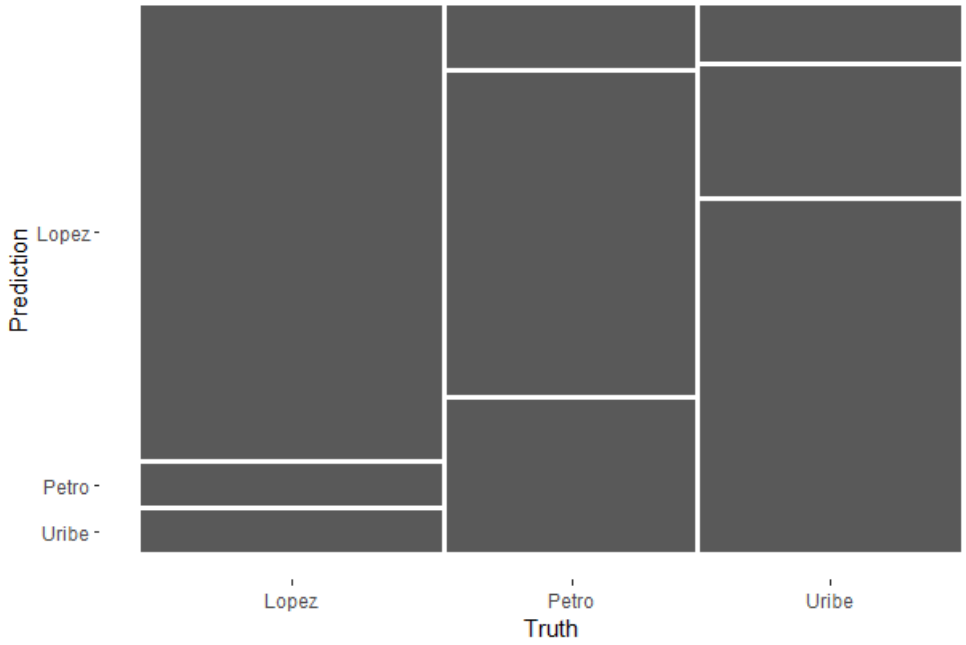
[2]Se utiliza la terminología del paquete Tidy Models, el cual fue usado para el presente taller. Una receta hace referencia a la especificación usada en el modelo y a toda la preparación previa que se le aplica a la base de datos antes de correr el modelo

Cuadro 4. Top 5 modelos con mejor precisión (Accuracy) promedio

	Precisión Promedio	Error Estándar
(1) Random Forest Referencia	0.713	0.0061
(2) Random Forest SMOTE - Normalizado	0.708	0.0059
(3) Elastic Net Multinomial Referencia	0.702	0.0065
(4) Random Forest con PCA	0.702	0.0067
(5) Elastic Net Multinomial SMOTE - Normalizado	0.70	0.0072

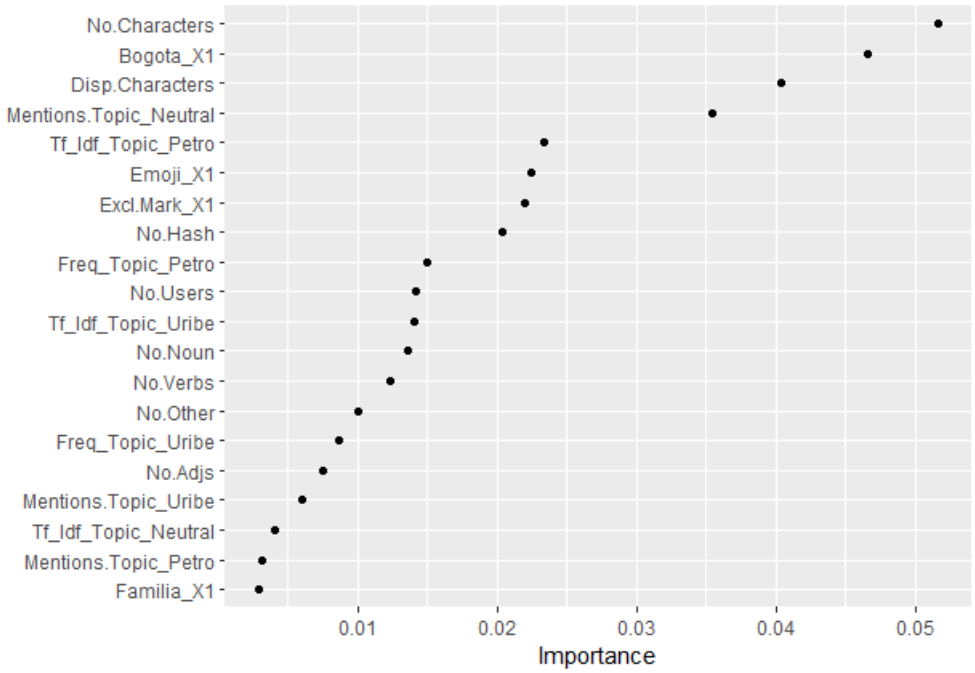
Como se puede ver en el cuadro 5, la precisión promedio de los modelos evaluados estuvo alrededor del 70 %, siendo el mejor modelo la combinación entre la receta básica de referencia y la metodología de estimación por bosques aleatorios. A pesar de que las variables incluidas en el modelo se derivan del análisis de estadísticas descriptivas, capturando así una parte de las unicidades del estilo de cada político, el modelo se queda corto en diferenciar de manera consistente los tweets de Alvaro Uribe y Gustavo Petro (ver figura 6). Por esta razón, se recurre a un modelo de clasificación de texto más complejo.

Figura 7. Matriz de confusión mejor modelo (Random Forest)



Por útltimo, se hace un análisis de importancia de las variables para validar la intuición inicial derivada del análisis por medio de estadísticas descriptivas. Como se puede ver en la figura 7, las variables más importantes no afectan de manera significativa la calidad de la predicción del modelo, siendo las de mayor influencia el número de caracteres y si el tweet contiene la palabra "Bogotá", las cuales reducen la precisión del modelo en cerca de 5 % cada una. Este resultado se alinea con los resultados obtenidos en la métrica de precisión del modelo y denota la importancia de la longitud del Tweet y su deviación de la media en la clasificación de los Tweets.

Figura 8. Importancia de las variables del mejor modelo (Random Forest)



Para el segundo abordaje, se optó por complementar la especificación de la ecuación 3 con una matriz con las palabras más relevantes según la métrica Tf-idf. Lo anterior nos permite definir las palabras más características de cada político para incluirlas como predictores en nuestro modelo inicial, aumentando así su capacidad para identificar consistentemente el estilo particular de cada político. Esta especificación toma la siguiente forma:

$$\begin{aligned} Nombre_i &= +\beta_{i1} No.Caracteres + \beta_{i2} No.Menciones + \beta_{i3} No.Hashtags+ \\ &= \beta_{i4} Desviación.Caracteres + \beta_{i5} Emoji + \beta_{i6} Exclamación+ \\ &= \beta_{i7} Tema.Tf.Idf + \beta_{i8} Tema.Menciones + \beta_{i9} Tema.Frecuencia+ \\ &= \beta_{i10} Bogota + \beta_{i11} Macro + \beta_{i12} Familia + \beta_{i13} No.Sustantivos+ \\ &= \beta_{i14} No.Adjs + \beta_{i15} No.Verbos + \beta_{i16} No.Otros + \beta_{i17} Matriz.Tfidf + \epsilon i \end{aligned} \tag{2}$$

Para esta especificación se probaron 3 recetas y 3 metodologías de estimación. En cuando a las recetas, la primera define la especificación de la ecuación 3, con una matriz con loa primeras 2000 palabras con el mayor valor TF-IDF, normalizando las variables continuas, aumentando la muestra via SMOTE y ejecutando un PCA para incluir los componentes principales que sumados capturen el 90 % de la variación del modelo como predictores adicionales. La segunda receta mantiene los preprocesamientos de la primera, pero utiliza la variable de texto lematizada después de la limpieza en lugar de solo la variable de texto limpia. La tercera receta mantiene los preprocesamientos de la primera, pero en lugar de generar una matriz con las primeras 2000 palabras según su valor TF-IDF, se toman las primeras 3000 palabras. Con respecto a las metodologías de estimación, se evaluó un Elastic Net Multinomial, un Lasso y una metodología por árboles aleatorios. Todas las metodologías se corren con los parámetros por default de cada paquete para alivianar la carga de cómputo.

Cuadro 5. Top 5 modelos con mejor precisión (Accuracy) promedio

	Precisión Promedio	Error Estándar
(1) Elastic Net Multinomial con matriz Tf-Idf = 3000	0.828	0.0038
(2) Elastic Net Multinomial con matriz Tf-Idf = 2000	0.819	0.0060
(3) Elastic Net Multinomial Lematizado	0.819	0.0060
(4) Lasso Multinomial con matriz Tf-Idf = 3000	0.815	0.00489
(5) Lasso Multinomial con matriz Tf-Idf = 2000	0.807	0.0053

Como se puede ver en el cuadro 6 la precisión promedio aumenta significativamente a cerca de 80 %, 10 p.p. más que la especificación inicial expuesta en la ecuación 3. También se observa que, a diferencia de la primera especificación, los modelos con Elastic Net y Lasso presentan mejores resultados que los con árboles aleatorios. Es posible que esto sea

producto del aumento significativo en el número de predictores, donde las primeras metodologías, a diferencia de los árboles aleatorios, se benefician de la selección de variables, regularizando las variables o dejando solo aquellas con los efectos más grandes sobre la variable dependiente. Es importante resaltar que, antes de correr estos modelos, se probó la pertinencia de la normalización y el PCA a la luz de su desempeño en la primera especificación. Sin embargo, a diferencia de este primer abordaje, los modelos con normalización y PCA tuvieron mejor desempeño que aquellos que no, probablemente por el aumento significativo en el número de predictores.

En este orden de ideas, y teniendo en cuenta los resultados del cuadro 6, el modelo con el mejor resultado fue el Multinomial con regularización tipo Elastic Net con la receta número 3, alcanzando una precisión promedio del 82,8 % en la validación cruzada tipo k-fold. Por lo tanto, se procede a optimizar este modelo con el objetivo de mejorar la precisión obtenida con el modelo por default. Los parámetros optimizados se encuentran en el cuadro 7.

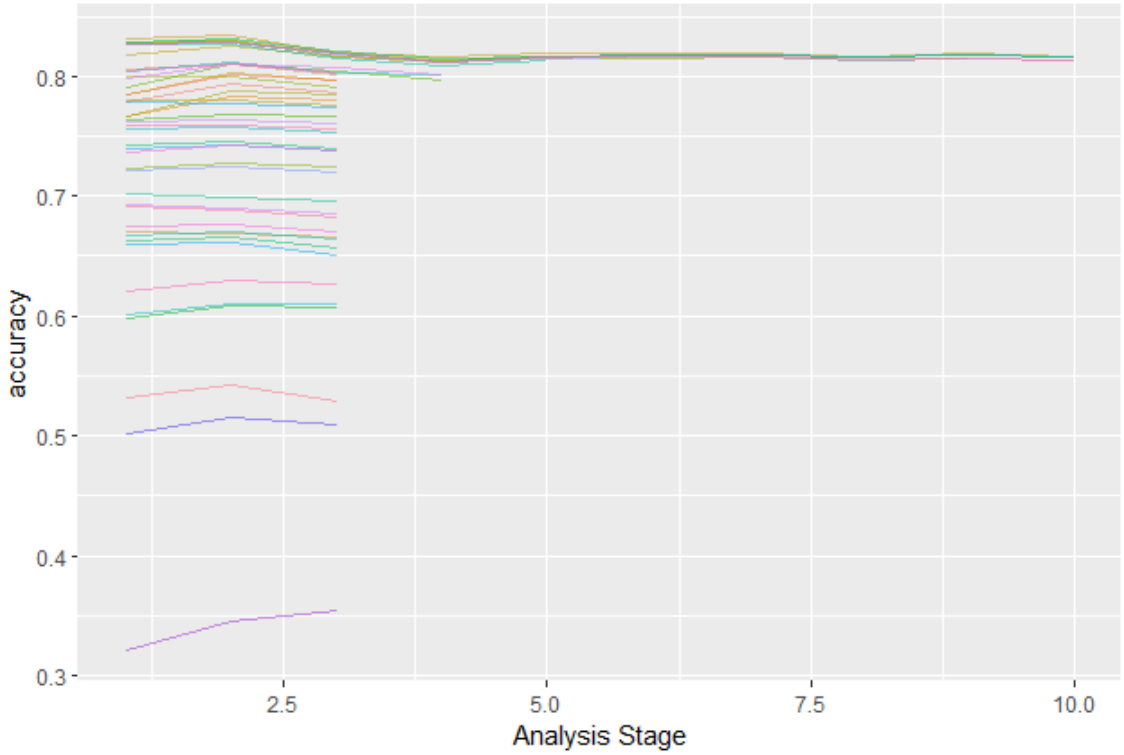
Cuadro 6. Rangos tenidos en cuenta para el proceso de optimización de parámetros

	Elastic Net Multinomial	
Parámetro	Límite inferior	Límite Superior
Penalidad	0.001	1
Alpha	0	1
Tamaño de la grilla		50

Dado que se iban a optimizar una cantidad muy alta de parametros, se implementó un proceso de optimización por modelos ANOVA, que elimina las combinaciones de parámetros que son menos probables de arrojar los mejores resultados, reduciendo así los tiempo de cómputo de la optimización. En la Figura 8 se puede observar los combinaciones que fueron filtradas en los primeros resampleos de la optimización del Elastic Net Multinomial

en función de la Precisión como métrica de referencia, mostrando los recursos y el tiempo ahorrados en no seguir sampleando dichas combinaciones en las submuestras aletorias restantes.

Figura 9. Proceso de optimización de parametros, proceso de eliminación de combinaciones no prometedoras

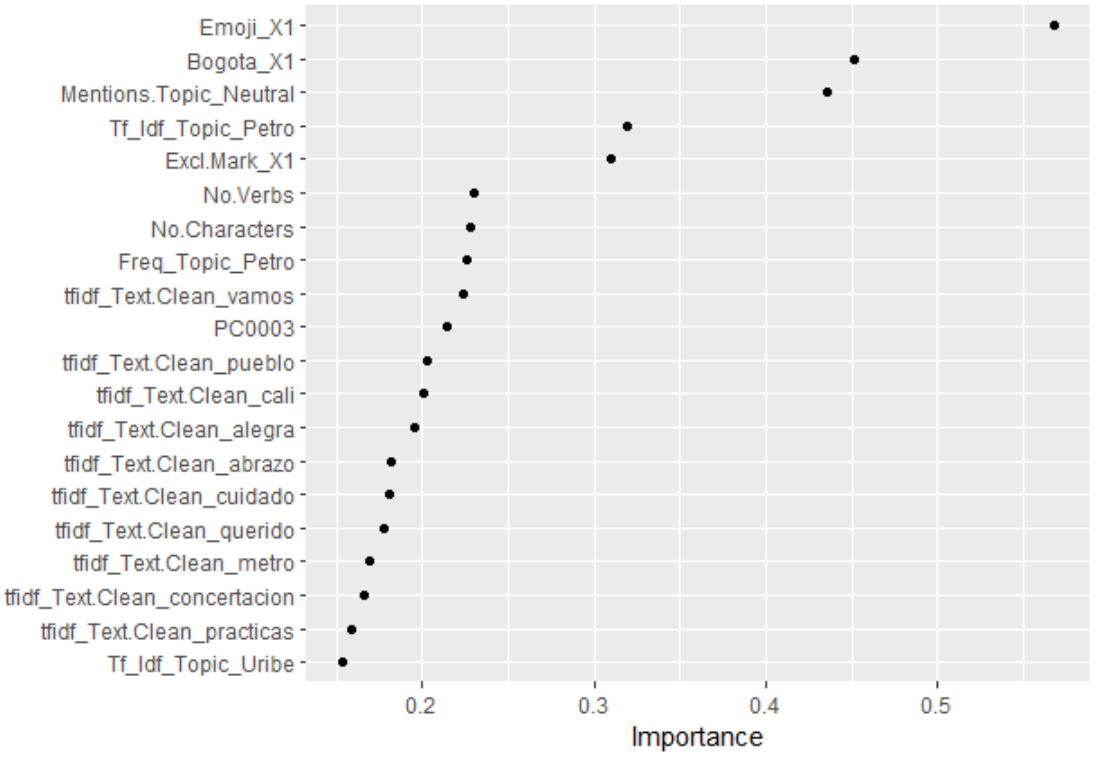


Sin embargo, después de la optimización se obtiene una precisión cercana al 81,7 %, es decir, menor al modelo no optimizado. Es probable que esto sea una señal de sobreajuste del modelo a la base de entrenamiento. Teniendo en cuenta los pocos hiperparámetros disponibles para optimizar en esta metodología, no se prueban más especificaciones y se sube el modelo no optimizado a Kaggle, el cual obtiene una precisión del 82 %, el mejor obtenido.

Al final, se revisa la importancia de las variables una vez más para ver qué cambios

hubo con respecto a la primera especificación. Como se puede ver en la figura 9, las variables creadas toman una mayor relevancia, especialmente las binarias que identifican la presencia de emojis y la palabra Bogotá en los tweets. También resalta la importancia de la categoría "neutral.[en] las categorías creadas por las menciones. Dicha categoría se creó para clasificar los tweets que no tenían menciones relevantes de ninguno de los políticos. En general, se evidencia una mayor importancia relativa de las variables creadas para la primera especificación, proablemnete potenciadas en su capacidad de discriminación por la matriz TD IDF de la segunda especificación.

Figura 10. Importancia de las variables del mejor modelo final (Elastic Net Multinomial)



El anterior análisis sugiere que, para seguir mejorando la precisión del modelo, resultaba necesario seguir creando más variables específicas que lograran capturar las diferencias entre los tweets mayoritarios que fueron asignados a la categoría neutral por las variables

personalizadas creadas, especialemente para los tweets de Uribe y Petro, los cuales siguen siendo los que tienen más error de clasificación.

Conclusiones

Los modelos de clasificación con datos de texto resultan ser significativamente diferentes a los modelos tradicionales de clasificación. El entendimiento profundo de la base de datos resulta esencial, así como la prueba de múltiples metodologías para hacer cada vez mejores predicciones.

Aunque por temas de tiempo y eficiencia en el documento no se documento el proceso completo para llegar a los resultados finales, las predicciones en Kaggle son el fruto de proceso de mejora continua en la que se aplicó casi que todos los conceptos aplicados en clase con el objetivo de mejorar el modelo lo más posible. En primer lugar se empezó con un modelo simple, construyendo variables a partir de la variable original de tweets, haciendo uso de las herramientas de identificación y extracción de texto. En segundo lugar, se probó con modelos más complejos que involucran tokenización de texto, matrices TD IDF. Con el objetivo de reducir los tiempos de cómputo, se limpió la variable de texto y se hizo un proceso de lematización de texto. Sin obtener muchas mejoras.

Acto seguido se utilizó un algoritmo de Componentes Principales para reducir la dimensionalidad del problema, dado que la introducción de la matriz TD IDF introducía muchos nuevos predictores. Su inclusión, junto con las variables originales mejoraron el modelo marginalmente.

Por último, se hizo la estimación del modelo por medio de diferentes metodologías aprendidas en el curso como árboles aleatorios, Xgboost, Elastic Net Multinomial, Lasso Multinomial y Redes Neuronales. Se identificó que con muchos predictores las metodologías con árboles pierden poder de predicción, mientras que las metodologías con regularización mejoran. A pesar de que se intentó correr y tunear un modelo de redes neuronales con Tidy

Models, sin optimizar y optimizado, el modelo no arrojó resultados competitivos (precisiones cercanas al 50 %). Resulta necesario revisar esa metodología y entender mejor su funcionamiento. Después de todo el proceso, quedaron por validar más pasos de preprocesamiento de la data de entrenamiento, así como la creación de variables adicionales para capturar mejor las diferencias entre los tweets de Petro y Uribe, los cuales eran en los que el modelo se veía más limitado. Adicionalmente, hizo falta un análisis de sentimientos, el cual pudo beneficiar mejor la caracterización de los tweets. Por último, también faltó por explorar una potencial combinación de los modelos de clasificación de texto LDA (el cual también se exploró y se puede ver en el código) con los modelos finales con matrices TD IDF. Queda el sin sabor de no haber llegado a 85 % o más después de haber intentado casi todo lo aprendido. Valoraríamos mucho un feedback de qué faltó o qué se pudo haber hecho mejor para llegar a esos valores.

Nota: Github

En el siguiente link https://github.com/ra-carrillo/20231_BDML_Problem_Set4_Predicting_Tweets se encuentran el acceso al repositorio del Problem set 4, donde se almacena tanto el documento en PDF, como el código y los gráficos.

Referencias

[1] Magnani, E,(2017)., *Big data y política* NUSO N 269.https://nuso.org/articulo/big-data-y-politica/

Sarmiento-Barbieri, I,(2023)., *Texto como datos* Tutorial.