Tecnológico de Monterrey

Evidencia 1. Artículo de investigación PBL1

MA2014: Análisis de métodos de razonamiento e incertidumbre

Adrián Mateos |A01722496
Antonio Pena | A017226888
Miguel González | A01198604
José Torres | A00835737
Karla Cantú | A01285550

18 de agosto de 2024

1. Problematización

Los correos electrónicos y mensajería digital son fundamentales para la comunicación de hoy en día, desafortunadamente estos medios se encuentran constantemente amenazados por el problema de correos/mensajes no deseados (spam).

Muchas personas hemos recibido mensajes de correo que no hemos solicitado, y a pesar de que los proveedores de correos electrónicos realizan estrategias "anti-spam", este sigue pasando a nuestras bandejas principales con información no solicitada y potencialmente peligrosa.

Los correos spam no solo inundan las bandejas de entrada de los usuarios, sino que también pueden servir como medios para realizar ataques cibernéticos a través de técnicas como phishing o distribución de malware. Es por esto que realizar detectores de spam eficientes es crucial para mejorar la proteccion y experiencia de los usuarios de estos servicios.

2. Enfoque

El enfoque del proyecto consiste en construir un clasificador de spam "genuo" haciendo uso de la probabilidad condicional mediante el teorema de Bayes, el cual nos porporciona una formula para calcular la probabilidad de un evento A dado que otro evento B haya ocurrido:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Y viceversa:

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$

Si, reorganizamos los términos en las formulas tenemos que:
$P(A|B)P(B) = P(B|A)P(A) = P(A \cap B)$. Por lo tanto el teorema de Bayes se puede reescribir como:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

Ahora, sea W el conjunto de todas las palabras que existen y m (el mail) un subconjunto de palabras que pertenecen a W: $m = \{w_1, w_2, \ldots, w_n\}$. Si queremos conocer la probabilidad de que m sea spam podemos utilizar el teorema de Bayes:

$$P(spam|m) = \frac{P(m|spam)P(spam)}{P(m)}$$
$$= \frac{P(w_1 \cap w_2 \cap \cdots \cap w_n|spam)P(spam)}{P(w_1 \cap w_2 \cap \cdots \cap w_n)}$$

$$= \frac{P(w_1 \cap w_2 \cap \cdots \cap w_n|spam)P(spam)}{P(w_1 \cap w_2 \cap \cdots \cap w_n|spam)P(spam) + P(w_1 \cap w_2 \cap \cdots \cap w_n|not\ spam)P(not\ spam)}$$

Al observar el numerador de la ecuación anterior $P(w_1 \cap w_2 \cap \cdots \cap w_n|spam)P(spam)$, el cual es equivalente a la distribución de probabilidad conjunta de $P(w_1 \cap w_2 \cap \cdots \cap w_n \cap spam)$. Por la regla de la multiplicación, la expresión puede ser reescrita de la siguiente manera:

$$P(w_1 \cap w_2 \cap \cdots \cap w_n \cap spam) = P(spam)P(w_1|spam)P(w_2|w_1 \cap spam) \cdots P(w_n|\cap_{i=1}^{n-1} w_i \cap spam).$$

Es aquí donde entra la suposición "ingenua" del modelo, vamos a asumir que las palabras de los mensajes son mutuamente y condicionalmente independientes entre si, es decir que la probabilidad de que aparezca una palabra en particular no se ve afectada por la presencia de otras palabras en el mensaje. Esto se resume mediante la siguiente ecuación:

$$P(w_i|w_1 \cap \cdots \cap w_{i-1} \cap spam) = P(w_i|spam).$$

Sin embargo, lo opuesto es lo que generalmente ocurre, sabemos que las palabras que utilizamos SI dependen del resto de palabras que aparecen en el enunciado. Por ejemplo si un mensaje fuera: "Mi nombre es", la palabra que esperaríamos encontrar inmediatamente después sería un sustantivo propio. Es por esta razón que decimos que el clasificador es "ingenuo". A pesar de ello, este resulta funcionar bien en muchas situaciones.

Consecuentemente, para realizar la predicción, utilizaremos un modelo de estimación de probabilidad máxima a posteriori (MAP, por sus siglas en inglés). En donde un mensaje será categorizado como spam si se cumple que:

$$P(spam|w_1 \cap w_2 \cap \cdots \cap w_n) > P(not\ spam|w_1 \cap w_2 \cap \cdots \cap w_n),$$

Lo cual es equivalente a escribir:

$$P(w_1 \mid \text{spam})P(w_2 \mid \text{spam}) \cdots P(w_n \mid \text{spam})P(\text{spam})$$
$$> P(w_1 \mid \text{not spam})P(w_2 \mid \text{not spam}) \cdots P(w_n \mid \text{not spam})P(\text{not spam})$$

Para calcular las probabilidades anteriores utilizaremos las siguientes ecuaciones, sea W_t el conjunto de todas las palabras de todos los emails que pertenecen

a los datos predefinidos como entrenamiento y sean W_{ts}, W_{tn} los subconjuntos que contienen las palabras de todos los emails de spam y non-spam respectivamente. Entonces, para calcular la probabilidad de que una palabra w_i se encuentre en un email, dado que sabemos que es spam, realizamos:

$$P(w_i|spam) = \frac{\text{\# de ocurrencias de } w_i \text{ en spam emails}}{\text{\# total de palabras en spam emails}}.$$

Similarmente, para calcular la probabilidad de que una palabra w_i se encuentre en un email, dado que sabemos que no es spam, hacemos el calculo:

$$P(w_i|not\ spam) = \frac{\text{\# de ocurrencias de } w_i \text{ en non-spam emails}}{\text{\# total de palabras en non-spam emails}}.$$

Finalmente, necesitamos una manera de calcular las probabilidades P(spam) y P(not spam):

$$P(\text{spam}) = \frac{|W_{ts}|}{|W_t|}$$

$$P(\text{not spam}) = \frac{|W_{tn}|}{|W_t|}$$

3. Propósito

Este proyecto tiene como objetivo el utilizar herramientas avanzadas de inteligencia artificial, como el aprendizaje automático y el procesamiento del lenguaje natural, para crear un sistema capaz de proporcionar una defensa robusta contra el spam en diversos contextos, dirigido principalmente a crear una alternativa más segura para los usuarios de mensajería digital.

4. Información

El conjunto de datos utilizado para este proyecto fue cargado inicialmente desde un archivo CSV que contenía una combinación de correos etiquetados como "ham" (no spam) o "spam". Al cargar el archivo, se observaron varias columnas que no aportaban información útil para el análisis; y que por lo tanto debían ser eliminadas.

Para facilitar la interpretación y el procesamiento por parte del algoritmo se transformaron las etiquetas de los correos ("ham" y "spam") en una variable binaria, donde "0" representa un correo no spam y "1" representa un correo spam.

Los correos electrónicos contienen texto en diversas formas, es por eso que, para reducir la complejidad del modelo, se convirtieron todas las palabras a minúsculas, se eliminaron los caracteres especiales y se eliminaron palabras comunes que no aportan información relevante en el mensaje, por ejemplo: "y", "la", "los", "a", entre otras.

Antes de pasar al entrenamiento del modelo, se realizó un proceso de lematización, el cual reduce la caracterización de palabras y las pasa a su forma base. Por ejemplo: Transformar las palabras "necesita", "necesitó" y "necesitaba" a "necesitar", creando más eficiencia en el modelo.

El entrenamiento del modelo fue esencial para poder llegar a solucionar los correos spam, se utilizó el 0.8 de los datos para el entrenamiento y el 0.2 para las pruebas. Específicamente se entrenó un modelo para clasificar spam basado en el teorema de Bayes, asumiendo que las palabras dentro de un mensaje son independientes entre sí. Se calcula la probabilidad de que un correo sea spam evaluando cada palabra dentro de el correo, matemáticamente se expresa como:

$$P(spam|m) = \frac{P(m|spam)P(spam)}{P(m)}$$

En donde:

- $P(m|spam)$ es la probabilidad de observar el mensaje m dado que es $spam$.
- $P(spam)$ es la probabilidad a priori de que cualquier correa sea spam.
- $P(m)$ es la probabilidad de observar el mensaje, independientemente si es contenido spam o no.

El clasificador Naïve-Bayes nos permite clasificar la probabilidad conjunta $P(spam|m)$ como el producto de la probabilidad de que cada una de las palabras se relacionen con spam, creando independencia entre las palabras:

$$P(spam|m) = \prod_{i=1}^{n} P(w_i|spam)$$

Donde $P(w_i|spam)$ se calcula durante el entrenamiento del modelo, tomando en cuenta las palabras independientes dentro del mensajes.

De misma manera, las probabilidades $P(spam)$ y $P(no\ spam)$ son calculadas para el entrenamiento de datos, ya que reflejan la proporción de correos spam y no spam en los datos.

Un problema al entrenar el modelo es que existen ciertas palabras en los datos reales que no estaban presentes en el entrenamiento, lo que naturalmente afecta

el calculo del producto de la probabilidad de que una palabra se encuentre en un correo spam. Por lo que se utiliza la suavización de Laplace:

$$P(w_i|spam) = \frac{count(w_i, spam + 1)}{count(spam) + |V|}$$

En donde $|V|$ representa el tamaño del vocabulario.

Posteriormente de haberse entrenado el modelo, se realizaron pruebas con un conjunto de datos sin entrenar para evaluar el desempeño del modelo. Se utilizó una matriz de confusión, la cual ayuda a observar el desempeño del modelo de manera cuantificada. A continuación las descripciones y métricas obtenidas a partir del modelo:

- **Verdaderos positivos (TP):** Número de correos spam identificados correctamente como spam.
- **Falsos positivos (FP):** Número de correos no spam identificados incorrectamente como spam.
- **Falsos negativos (FN):** Número de correos spam identificados incorrectamente como no spam.
- **Verdaderos negativos (TN):** Número de correos no spam identificados correctamente como no spam.

En donde se obtuvieron los siguientes resultados en la clasificación:

	Positivos Predichos	**Negativos Predichos**
Positivos Reales	128 (TP)	16 (FN)
Negativos Reales	6 (FP)	964 (TN)

Cuadro 1: Tabla de predicciones

Después, se realiza una evaluación del modelo con las siguientes métricas estadísticas:

- **Accuracy:** Porcentaje de correos spam y no spam clasificados correctamente.
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
- **Precisión:** Porcentaje de correos spam clasificados correctamente como spam.
$$Presicion = \frac{TP}{TP + FP}$$

- **Recall:** Sensibilidad del modelo; porcentaje de correos de spam clasificados como tal.

$$Recall = \frac{TP}{TP + FN}$$

- **Puntaje F1:** Media armónica de presición y recall (balance entre las métricas).

$$F1Score = 2 \cdot \frac{Presicion \cdot Recall}{Presicion + Recall}$$

Calculando los valores para cada métrica, se obtuvo lo siguiente:

Métricas	**Valor**
Accuracy	0.980
Precisión	0.955
Recall	0.889
Puntaje F1	0.920

Cuadro 2: Métricas de Rendimiento

5. Razonamiento

Como teoría, se esperaba que el clasificador de Naïve-Bayes fuera una buena herramienta para determinar correos spam y evitarlos, analizando las palabras que construyen un mensaje de correo electrónico mediante probabilidades y empleando el teorema de Bayes. Tras observar las métricas del modelo elaborado, se obtuvieron buenos resultados. Cabe añadir que a pesar de tener buenas métricas, el modelo predijo 16 falsos positivos y 6 falsos negativos, equivocándose en estos casos aislados.

Por ejemplo, un caso de falso positivo clasificado se encuentra en un mensaje con las palabras `''yavnt''`, `''tri''`, `''yet''`, `''never''`, `''play''`, `''origin''`, `''either''`. Por otro lado, existe un talón de aquiles dentro de la clasificación del modelo, y es que dentro de los resultados de los falsos negativos, la mayoría de mensajes incluyen textos que carecen de sentido. Un ejemplo de un correo spam clasificado como no spam contiene las palabras `''email''`, `''alertfrom''`, `''jeri''`, `''stewarts''`, `''2kbsubject''`. La mayoría de estos casos se tratan de mensajes con palabras de distintos contextos que juntas en un mismo mensaje carecen de sentido y estructura, que a su vez un humano podría detectarlos como spam con menor dificultad, sin embargo, la ingenuidad del modelo de clasificación Naïve-Bayes no le permitió reconocer estos mensajes como spam, siendo esta una posible limitación que se esperaba.

A pesar de que el clasificador Naïve Bayes sea eficiente y veloz, utiliza la suposición de independencia condicional en las palabras, pudiendo llegar a errores de clasificaciones. Esto es una limitación en el modelo, ya que existe un orden en el acomodamiento de las palabras dentro de un mensaje, y asumir la independencia es una simplificación importante que no captura la relación de las palabras con el mensaje, haciendo que la predicción no pueda ser perfecta.

6. Conclusiones

Sin duda alguna fue interesante elaborar un modelo de clasificación, y que mejor que con un uso práctico y real como lo es la detección de mensajería spam. El considerar que el uso del teorema de Bayes se basa en la independencia de todas las variables, nos permitió suponer los posibles errores del modelo, que sí terminaron por suceder, pero afortunadamente en una escala mínima. A su vez, y a pesar de los errores, quedó demostrado que el uso de la inteligencia artificial nos puede volver más eficaces a la hora de realizar una labor en específico, pudiendo detectar 128 casos de más de 1000 mensajes, con un potencial riesgo en cuestión de minutos. Sin duda alguna, el modelo realizó una tarea que inclusive al trabajador más capacitado en detectar correos maliciosos le pudiese haber tomado horas. Dentro de una empresa con la misma proporción, se hubiesen garantizado el proteger a sus empleados de 128 potenciales ataques o fraudes. Cabe destacar que este modelo se elaboró con fines educativos, estudiando el uso de la probabilidad desde otro punto de vista. No obstante, sería interesante ver el comportamiento de este modelo con una escala mayor dentro del ámbito laboral.

Referencias

[1] Russell, S. J., & Norvig, P. (1996). *Inteligencia artificial: un enfoque moderno*. Prentice Hall Hispanoamericana.