PBL 1. CLASIFICADOR DE MENSAJES DE SPAM EN PYTHON

Instituto Tecnológico y de Estudios Superiores de Mty
Escuela de Ingeniería y Ciencias
Ingeniería en Ciencias de Datos y Matemáticas
Profesor: Marco Otilio Peña Díaz
Monterrey, Nuevo León. Fecha, 15 de Agosto de 2022.

Enrique García Varela A01705747
Ricardo Camacho Castillo A01654132
Grace Aviance Silva Aróstegui A01285158
Michelle Yareni Morales Ramón A01552627
David Esquer Ramos A01114940

Palabras clave:
Probabilidad y estadística
Diseño de algoritmos
Teoría de decisiones
Regla de Bayes

General
El propósito de esta actividad es desarrollar bajo la metodología de SCRUM y PBL, una serie de pasos que te permitan reforzar las competencias de organización, comunicación y colaboración y que a su vez les permita construir el conocimiento necesario para resolver el problema.

1 Problematización

Los mensajes spam hacen referencia a correos y comunicaciones no solicitadas que se envían de forma masiva por internet o por otros medios de mensajería electrónica que pueden saturar la bandeja de entrada, por lo que es necesario detectar estos mensajes para que los correos que sean realmente importantes no se pierdan entre la basura informática.

2 Enfoque

El enfoque o la perspectiva a la cual va dirigida este texto que se basa en la creación de un algoritmo en este caso del teorema de bayes para la detección de mensajes o correos electrónicos que sean o no spam, seria un enfoque educativo, matemático y estadístico.

Además de estos ya mencionados se pudiera considerar que parte del enfoque seria de aprendizaje al momento de realizar el texto.

3 Propósito

El fin de esta actividad es que realicemos un código en Python para que a partir de una base de datos la cual contiene un registro de mensajes y su respectiva clasificación de si es "spam" o "ham"; limpiar la base, analizar las palabras de cada mensaje, y junto con el conocimiento de la Teoría de Bayes estimar la probabilidad de que un mensaje que sea ingresado se clasifique como spam o ham.

4 Información

Naive Bayes es un algoritmo de clasificación de aprendizaje automático probabilístico basado en el Teorema de Bayes.

El Teorema de Bayes es una proposición matemática utilizada para calcular probabilidad condicional, es decir la probabilidad de que ocurra un evento dado que otro evento ha ocurrido [Nagesh Singh Chauhan, 2022].

Al ser una formula matemática basada en la probabilidad condicional, una de sus mayores utilidades en ML es para el procesamiento de Lenguaje Natural. De este modo su formula es la siguiente:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \tag{1}$$

Es necesario para que esta formula se cumpla que las variables sean independientes entre sí y que las variables tienen la misma importancia sobre el resultado de predicción. [Petrov and Ernesto, 2008]

Si tomamos en cuenta la posibilidad de que X, sea un vector con múltiples variables. Entonces el numerador de la fracción anterior estaría dado por el producto de todas las probabilidades condicionales y la probabilidad total, mientras que el numerador por el producto de la probabilidad de cada variable. De modo que:

$$P(y|x_1, x_2, ..., x_n) \ \alpha \ P(y) \prod_{i=1}^{n} P(x_i|y) \qquad (2)$$

Ahora para el caso particular de la clasificación de spam, la probabilidad sería para una clasificación de dos resultados, si es spam o si no lo es. De tal forma que el algoritmo estaría completo al conocer que probabilidad es mayor, la de ser o no ser spam.

5 Razonamiento

A través del código, buscamos darle solución a la problemática planteada. En esta obtuvimos una base de datos que venía ya con un asunto de un correo y con la clasificación que se había hecho si era spam o ham.

Con esto realizamos un procedimiento de análisis de los mismos y los formatos con los que se estarían trabajando. Para lo mismo, realizamos una limpieza de los mismos y seguido a ello el preprocesamiento de la información, para poder trabajar texto. Entre estas técnicas, realizamos un proceso de stemmer, en el que se hace solo una palabra para las que tuvieran misma semántica, quitar los stopwords, la cual es una serie de palabras muy comunes en el inglés y que no proporciona información para nuestro análisis.

Más adelante continuamos con la forma de minería de datos. De modo que se divide la información en un dataset de entrenamiento y uno de prueba. Para el primero, se obtuvo un diccionario para las palabras que estaban en los asuntos y para cada una se cálculo su probabilidad, respecto a si estaba clasificado como ham o spam. Cabe destacar, que para las palabras que no aparecían dentro de estas clases se definían con una probabilidad de 1 de modo que al momento de realizar la multiplicación no afectara al resultado.

Ya con los diccionarios creados, pudimos realizar la predicción calculando la probabilidad bayesiana y comparando entre la mayor, ya sea para spam o ham, de modo que como se explico anteriormente, este sería la clasificación a mantener. Se aplicó para todo el set de prueba, de modo que se pudiera hacer una evaluación con las métricas de matriz de confusión. Y finalmente, se realizó una sección donde se pueda meter texto y que la prediga.

6 Conclusiones

A pesar de haber tenido un código en el cual pudimos realizar la clasificación con base en las probabilidades de condicionales que nos da resultados muy acertados, no es posible estar seguros que va a funcionar todo el tiempo ya que hay muchas maneras de mejorar o de cambiar los resultados dependiendo de que es lo que queremos hacer. Por ejemplo, si queremos que no nos lleguen mensajes de SPAM, y tal vez se nos pueda pasar algún correo que sea HAM al SPAM o viceversa, que se nos pueda pasar un correo de SPAM a HAM. Todo esto depende del enfoque del problema. En nuestro caso utilizamos las stopwords pero se pudieron haber eliminado algunos verbos, tomar solo uno de los grupos que conforman las stopwords.

Tomando en cuenta lo anterior, para futuras consideraciones el código se podría mejorar utilizando comparaciones con otros métodos, tanto para más análisis descriptivo de los datos, como para considerar algunos factores como la especificidad o sensitividad. Un método de estos es el Tf-idf (del inglés Term frequency – Inverse document frequency) el cual podría ayudar considerablemente ya que es una medida numérica que ayuda a expresar cuán relevante es una palabra para un documento en una colección. Este valor aumenta proporcionalmente al número de veces que una palabra aparece en el documento, en nuestro caso en los correos recibidos, pero es compensada por la frecuencia de la palabra en la colección de documentos, lo que permite manejar el hecho de que algunas palabras son generalmente más comunes que otras y así mejorar considerablemente la precisión de nuestro clasificador.

References

[Nagesh Singh Chauhan, 2022] Nagesh Singh Chauhan (08 de Abril de 2022). `www.marte.mar`.

[Petrov and Ernesto, 2008] Petrov, V. V. and Ernesto, M. P. (2008). *bayes*. Number 519.2 PET. Dirac.