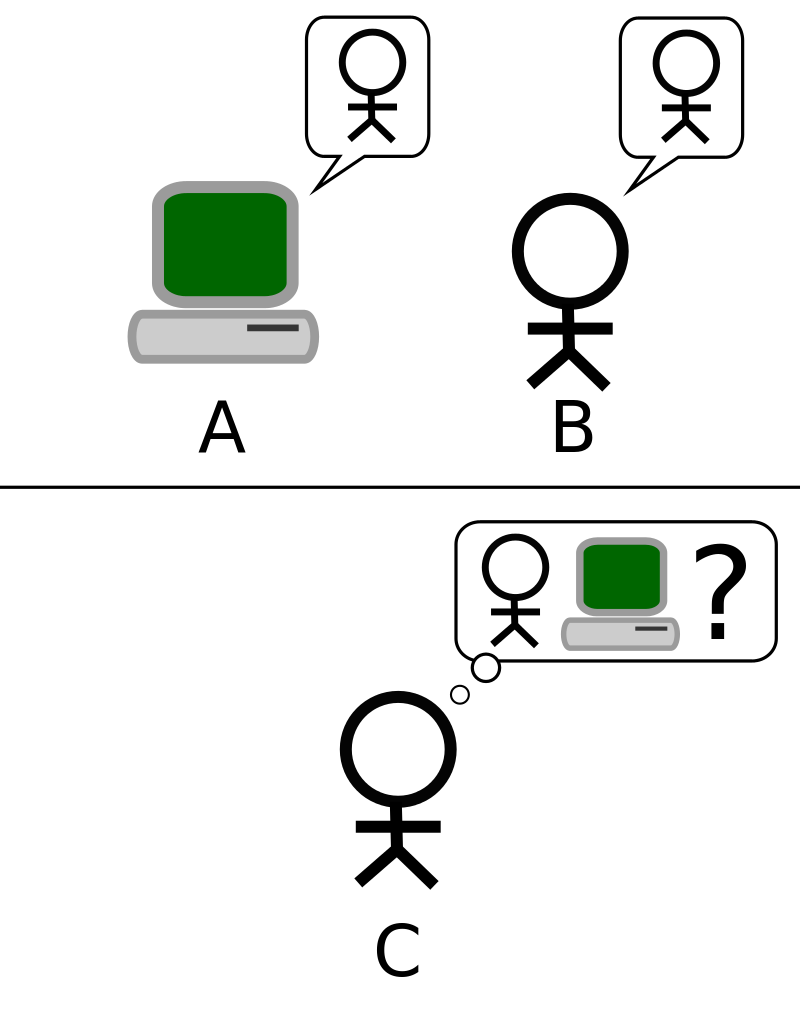
**Procesamiento del Lenguaje Natural**

Para comenzar a hablar acerca de lo que es el Procesamiento del Lenguaje Natural, debemos mencionar de qué manera surgió, o mejor dicho a través de qué experimentos comenzó a conocerse a esto como Procesamiento del Lenguaje Natural. 

Surgió alrededor de la década de 1950.Uno de los experimentos que dio inicio al Procesamiento del Lenguaje Natural fue el de Alan Turing, por ejemplo, el cual fue titulado Prueba o Test de Turing. Este experimento consistía en un examen que se realizaba para evaluar la capacidad de una máquina para poder demostrar un comportamiento de inteligencia similar o prácticamente indistinguible del que tiene un ser humano.

Otro experimento fue el de Georgetown en el año 1954 , el cual consistía en la traducción automática de más de sesenta oraciones rusas al inglés. Los autores afirmaron que dentro de tres o cinco años, la traducción automática sería un problema resuelto. Sin embargo, el progreso real fue mucho más lento, y después del informe ALPAC en 1966, que encontró que la investigación de diez años no había cumplido con las expectativas, la financiación para la traducción automática se redujo drásticamente. Poco más investigación en traducción automática se llevó a cabo hasta finales de la década de 1980, cuando se desarrollaron los primeros sistemas estadísticos de traducción automática.

El ALPAC era un comité conformado por varios científicos, establecido en 1964 por el gobierno de los Estados Unidos para evaluar el progreso en lingüística computacional en general y en traducción automática en particular.

Entonces para obtener una definición formal podemos decir que *“El lenguaje natural(LN) es el medio que utilizamos de manera cotidiana para establecer nuestra comunicación con las demás personas. El LN ha venido perfeccionándose a partir de la experiencia a tal punto que puede ser utilizado para analizar situaciones altamente complejas y razonar muy sutilmente. Los lenguajes naturales tienen un gran poder expresivo y su función y valor como una herramienta para razonamiento.”*(Mg. Cortéz Vazquez,2009,pág 46)

El procesamiento del lenguaje natural debe realizarse de forma sistemática, dividiéndolo en partes, agregando elementos gramaticales e identificando elementos interesantes. Los siguientes son algunos de los elementos comunes del procesamiento en NLP:

* Tokenización: consiste en dividir el texto en oraciones y estas, en palabras.
* Normalización: estandariza todas las palabras, por ejemplo, convirtiendo todas las palabras en mayúsculas o minúsculas.
* Eliminación de palabras redundantes: consiste en omitir o eliminar palabras redundantes, como los artículos, que no contribuyen en gran medida a entender el texto.

Esto se puede realizar por ejemplo mediante la utilización de librerías como NLTK. En primer lugar importamos las librerías:

**>>import nltk**

**>>from nltk import word\_tokenize**

**>>from nltk.corpus import stopwords**

Tokenización. Con el archivo que utilizaremos para el procesamiento:

**>>archivo\_tok = word\_tokenize(archivo)**

Normalización:

**>>archivo\_tokenizado = [word.lower() for word in archivo\_tok if re.search("\w",word)]**

Eliminación de palabras redundantes.Realizando la siguiente función:

**>>def remover\_stopwords(texto):**

**return [word for word in texto if word not in stopwords.words('spanish')]**

Luego:

**>>archivo\_tok\_sin\_stops = remover\_stopwords(archivo\_tokenizado)**

**Historia de NLTK**

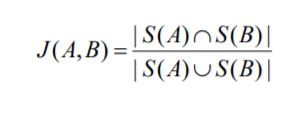
Fue desarrollado por Steven Bird y Edward Loper en el Departamento de Ciencias de la Información y Computación de la Universidad de Pennsylvania. NLTK incluye demostraciones gráficas y datos de muestra.

NLTK está destinado a apoyar la investigación y la enseñanza en PNL o áreas estrechamente relacionadas, incluida la lingüística empírica, la ciencia cognitiva, la inteligencia artificial, la recuperación de información y el aprendizaje automático.

**Metodologías de comparación**

**Jaccard Similarity**

Este método se realiza comparando los trigramas obtenidos del texto procesado, mediante la utilización del índice de Jaccard:

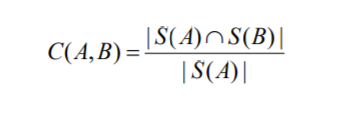


Donde “*S (A) y S (B) representan los conjuntos de trigramas en los documentos sospechosos y originales respectivamente. Su intersección (nominador) representa el conjunto de trigramas coincidentes en aquellos documentos, mientras que su unión (denominador) representa el conjunto de todos los trigramas en esos*

*documentos.”*(Specia, 2010, pág 4)

**Containment measure**

Este es otro método que se realiza también comparando trigramas. *“Similar al coeficiente de Jaccard, Containment measure calcula los trigramas que se cruzan, pero normaliza por los trigramas solamente en el archivo sospechoso.”*(Specia, 2010, pág 4)

**

**Longest Common Subsequence(LCS)**

Este es un método diferente que no utiliza los trigrams sino que“*calcula*

*la secuencia más larga de palabras incluidas en ambos*

*documentos originales y sospechosos.”*(Specia, 2010, pág 5)

**Gráfico de comparación**

Para poder analizar de mejor manera los datos obtenidos de los métodos de medición es conveniente obtener un gráfico comparativo de los mismos. Estos se pueden realizar mediante la utilización de librerías como matplotlib y numpy. Por ejemplo:

**>>import matplotlib.pyplot as plt; plt.rcdefaults()**

**>>import numpy as np**

**>>import matplotlib.pyplot as plt**

Luego se definen los elementos a graficar asi como su orientación, nombre del grafico, etc.

**>>objects = ('score\_containment','score jaccard','score lcs')**

**>>y\_pos = np.arange(len(objects))**

**>>performance = [score\_containment,score\_jaccard,score\_lcs]**

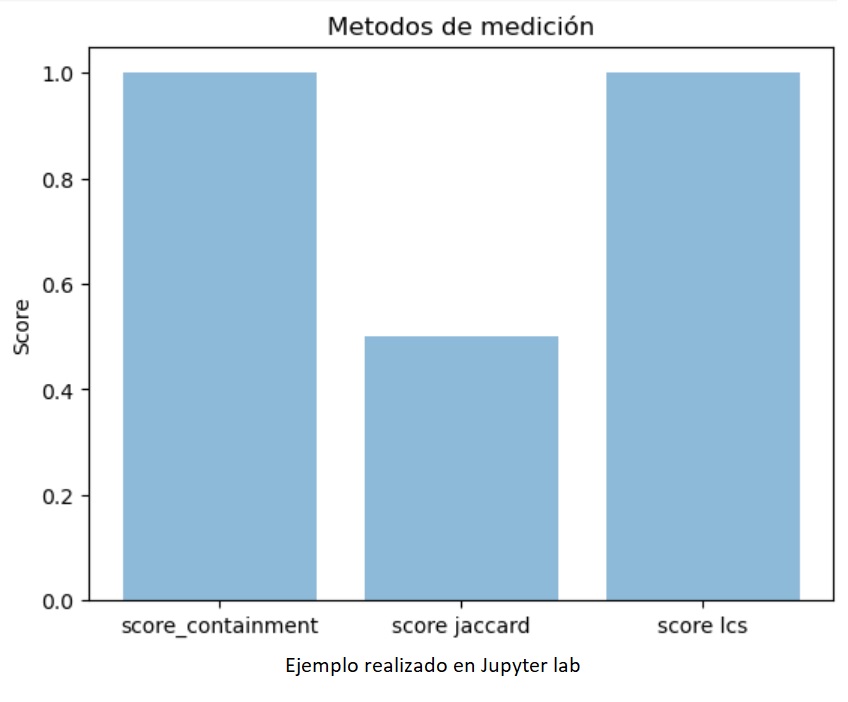
**>>plt.bar(y\_pos, performance, align='center', alpha=0.5)**

**>>plt.xticks(y\_pos, objects)**

**>>plt.ylabel('Score')**

**>>plt.title('Metodos de medición')**

**>>plt.show()**



**Bibliografía:**

https://es.wikipedia.org/wiki/Prueba\_de\_Turing

<https://www.hisour.com/es/natural-language-processing-42789/>

<https://en.wikipedia.org/wiki/ALPAC>

<https://core.ac.uk/download/pdf/304898423.pdf>

<http://datascience.recursos.uoc.edu/es/procesamiento-del-lenguaje-natural-nlp/>

<https://d1wqtxts1xzle7.cloudfront.net/40077408/Using_Natural_Language_Processing_for_Au20151116-27205-18j9w7z.pdf?1447706034=&response-content-disposition=inline%3B+filename%3DUsing_Natural_Language_Processing_for_Au.pdf&Expires=1606504822&Signature=REAlwB7NIDkvPM~tNgUfEjnANv8otHcIv2pv88qmD3EIlhUipjKoYq9QnpVhSZhQDSNnhqOMi53lFgNqnhNvB7cgF5J8HbTDH1cAbHJiCdbf3FhO538nuk3XYTjSddtI60zniGvMQg9n-igDm0mU2T1sb61yBx8ZCZMGdYyS~Ct2YP1wlLFaMUzgDRgGjpgtXcJQs8RV0Nk-DEaeG8uW~hqVio2t2xYdw7Ry50SqnTzPeA0eI-O6bczAduah9yw8pgtMJHVAsOEjSSKgT9HtGokyPrbtQhl~FsLRaxQo3LYI7KtQeisZnbbll3xsJ-WIPozG1ijmT3Z0corJDFyBnw__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA>

<https://pythonspot.com/matplotlib-bar-chart/>

https://en.wikipedia.org/wiki/Natural\_Language\_Toolkit#:~:text=It%20was%20developed%20by%20Steven,graphical%20demonstrations%20and%20sample%20data.&text=NLTK%20supports%20classification%2C%20tokenization%2C%20stemming,parsing%2C%20and%20semantic%20reasoning%20functionalities.