

**Universidad Nacional Autónoma de México**





**Facultad de Ingeniería**



División de Ingeniería Eléctrica

Asignatura: Análisis y Procesamiento Inteligente de Textos

Profesor (a):M.P. Octavio Augusto Sánchez Velázquez

PROYECTO

**“RESUMEN DE TEXTOS AUTOMÁTICO”**

Alumna(s): Buendía Valdivia María Angélica

Espinosa Curiel Oscar

Isabel Gómez Yareli Elizabeth

Zagoya Mellado Roberto Uriel

Grupo 1

Semestre 2020-1

Fecha de entrega: 2020

Índice

[**I.** **Introducción** 3](#_Toc41473980)

[Motivación (interna o Externa) 4](#_Toc41473981)

[Justificación (que es lo que justifica la existencia de ese trabajo) 4](#_Toc41473982)

[Problema/Pregunta (¿Qué es lo que pretende responder con este trabajo?) 5](#_Toc41473983)

[Solución Propuesta 5](#_Toc41473984)

[**II. Marco teórico** 10](#_Toc41473985)

[Antecedentes (Por qué creo que mi solución Propuesta va a funcionar). Algo parecido a lo que estoy haciendo o lo mismo, pero con variaciones, ¿de dónde saqué mi solución propuesta va a funcionar. 10](#_Toc41473986)

[Trabajos relacionados. (Comparación con los trabajos y el nuestro) 11](#_Toc41473987)

[**III. Método experimental** 13](#_Toc41473988)

[III.I. Datos ¿Cómo se obtuvieron? 13](#_Toc41473989)

[III.II. Descripción de su método. 14](#_Toc41473990)

[Descripción del experimento (Responder a la pregunta). 28](#_Toc41473991)

[Presentación de resultados. 28](#_Toc41473992)

[**IV. Conclusión** 35](#_Toc41473993)

[Discusión de resultados 35](#_Toc41473994)

[Descripción de observaciones 35](#_Toc41473995)

[Trabajo futuro 35](#_Toc41473996)

[**V. Referencias** 35](#_Toc41473997)

[https://redhistoria.com/historia-y-evolucion-del-resumen-de-textos/ 35](#_Toc41473998)

# **Introducción**

Durante la existencia de la humanidad el conocimiento se ha convertido esencial para su desarrollo y crecimiento, por lo cual la lectura de información ha sido un punto primordial para desempeñar y llegar a cumplir cualquiera de sus objetivos.

Para comprender la mayoría de la información se tiene que leer un texto sumamente largo, lleno de conceptos y con una letra muy pequeña que en algunas ocasiones no se alcanza a ver. Para algunos, la lectura no es su hobby, además consume mucho tiempo y energía para comprender una idea, así que surge la necesidad de acortar la información más relevante. Dado esto, nace el resumen, este es un escrito que logra captar todas las ideas principales de un texto y sintetizarlas en un escrito mucho más pequeño. La extensión de un resumen varia, ya que no todos los textos son de la misma extensión, pero la mayoría tiene el 25% de la extensión original.

El objetivo del resumen es lograr reducir información a la más importante de cualquier tema, demostrar que se entiende lo resumido del texto original y que sea de utilidad para las personas. Los resúmenes han sido de gran ayuda ya que han servido para la realización de presentaciones, para estudiar algún tema, para realizar alguna exposición y para un trabajo académico.

Existen tipos de resúmenes como el indicativo y el informativo el indicativo es aquel donde se incluyen los temas del documento resumido y la metodología y no supera la extensión de las 100 o 200 palabras. El resumen informativo es aquel en donde se profundiza sobre el texto original siendo un método científico (objetivos, metodología, resultados y conclusiones), en el proyecto se implementa más el resumen indicativo.

En décadas anteriores aún se veía que se elaboraban con más frecuencia los resúmenes a mano, es decir, la persona leía todo el texto y al final o durante la lectura hacia anotaciones y subrayaba los puntos que a esa persona se le hacían relevantes e importantes en la actualidad aún se sigue utilizando pero conforme paso el tiempo se ha hecho se suma importancia el elaborar resúmenes en gran volumen así que gracias a que la tecnología ha ido evolucionando hoy en día existen diferentes softwares que nos permiten realizar estos resúmenes de una manera más práctica, automática. Estas herramientas ahora permiten ingresar el texto y resumirlo en cuestión de milisegundos, por lo que así se puede acortar mucho más el tiempo que inviertes en realizar esta actividad al lector y obtener el mismo resultado de haber captado la misma información.

Algunas tecnologías actuales pueden hacer un resumen coherente de un texto al tener en cuenta variables como la longitud, la escritura y la sintaxis.

El resumen automático es parte del área del aprendizaje automático y de la mimería de datos. Existen dos enfoques del resumen automático: la extracción y la abstracción, la extracción funciona seleccionando un subconjunto de palabras, frases u oraciones existentes en el texto original para formar el resumen, en la abstracción es un método que construye una representación semántica interna, a la que le aplican técnicas de generación de lenguaje natural para crear un resumen más similar al que realizaría un humano que más adelante se explicarán a más detalle.

Para este proyecto realizaremos una aplicación la cual podrá resumir un texto conservando las ideas más importantes que hay en este, se podrá ingresar (copiar y pegar) el texto, directamente en esta aplicación o, con la URL de la página que contiene un texto del cual tú quieres obtener un resumen. Este no será un sistema sin fallas y puede ser que, en los resúmenes, contenga equivocaciones porque funcionará con base en los algoritmos propios de las librerías de Python, las cuales se describirán en secciones posteriores, y no usa un pensamiento lógico y crítico del ser humano, por lo tanto, deberá estar atento al resumen obtenido y se deberá revisar los párrafos finales, para verificar la coherencia y verificar si es el resultado.

### Motivación (interna o Externa)

La motivación para realizar este proyecto fue ver la necesidad principalmente de los estudiantes para realizar los resúmenes, muchas veces se junta la carga de trabajo y no da tiempo de leer todos los documentos completos porque algunos tienen gran extensión de contenido y la mayoría de los profesores dejan muchos artículos y algunos libros para leer, pero para eso también sirven los resúmenes para reducir el tiempo y poner captar las ideas esenciales del documento sin tener que haber leído todo. Al elaborar este proyecto lo que se pretende es que los alumnos tengan una herramienta útil para y así facilitar y analizar su contenido desde artículos a textos de internet, de igual manera el ayudar a tener una mejor comprensión lectora.

### Justificación (que es lo que justifica la existencia de ese trabajo)

El proyecto fue elegido entre tres diferentes proyectos, las opciones que existían eran “El análisis de sentimientos”, “Chatbot” y “El resumen de textos”. Estos tres fueron elegidos por diferentes motivos, lo que nos llamó la atención en el análisis de sentimientos es que este es muy utilizado en redes sociales para verificar la opinión que se tiene de las comunidades sobre ciertos temas, también es utilizada en organizaciones mundiales, un ejemplo es Obama, él lo utilizo para sondear la opinión en las elecciones. Lo que encontramos de este proyecto es que el entender los sentimientos no es una tarea fácil ya que se requiere de conocer culturas, interpretar palabras dentro de un contexto, asumir faltas de ortografía y a partir de ello determinar diferentes reglas que es a lo que el usuario se quiere referir, etc., en pocas palabras es un proyecto que requería de mucho tiempo y de algunos conceptos extra-clase. Analizando el proyecto de chatbot, este nos pareció interesante ya que este es un programa informático con el que es posible mantener una conversación, tanto si queremos pedirle algún tipo de información o que lleve a cabo una acción. Para la elaboración de este proyecto el profesor dijo que teníamos poco tiempo para su elaboración ya que en un semestre no lo íbamos a poder terminar a menos que obtuviéramos ciertos programas ya elaborados, pero ese no era el objetivo, ya que el profesor quería que nosotros empezáramos el proyecto desde cero, considerando la opinión del profesor opinamos que la dificultad era muy alta para elaborar este proyecto en un semestre teniendo más materias por las cuales se debe de poner atención al igual que en la que estamos elaborando el proyecto, así que se llegó a la conclusión que los tres proyectos desde nuestro punto de vista valían la pena realizarlos, los consideramos muy interesantes para implementarlo en la materia.

Dado que nuestras opciones fueron aprobadas por el profesor, pero tomando en cuenta el factor de tiempo, el profesor nos sugirió realizar el óptimo para realizarlo durante el semestre y el que se eligió fue el de “Resumen de textos”.

### Problema/Pregunta (¿Qué es lo que pretende responder con este trabajo?)

Hoy en día existen muchas actividades que hacer, la sociedad cada vez trata de realizar todo de una forma más práctica. Conforme pasa el tiempo el internet de las cosas (IoT) y todo lo que se refiere a TIC ha ido entrando más en nuestros hogares, esto ha provocado que la mayoría de las personas quieran realizar sus actividades de una manera más fácil y automática por esta razón ha surgido este proyecto el “El resumidor automático”. Lo que se pretende es precisamente eso, es permitirles a las personas realizar sus resúmenes de una manera más fácil, práctica y en poco tiempo, sin tener que escribir a mano y sin invertir mucho tiempo.

### Solución Propuesta

Para la solución a nuestro proyecto estuvimos investigando en varias fuentes por lo que encontramos varias opciones para la solución. La opción que nos pareció como la mejor es la siguiente.

En lo que primero que se pensó fue en el lenguaje de programación, estuvimos investigando y “Python” fue el elegido ya que cuenta con Bibliotecas útiles como NLTK, esta sirve para el análisis de procesamiento de lenguaje natural (PLN) por lo cual fue de suma importancia para realizar el proyecto.

Otro punto que se tomó en cuenta fue el proceso para la elaboración del resumen paso a paso, el siguiente procedimiento es el que se tomó en cuenta para la solución y llegar al resumen del texto.

1.-Elegir el texto

Este paso consiste en elegir dos opciones un texto que el usuario quiera o que tenga escrito en computadora, y la segunda es elegir un texto que se encuentre en una página web.

Para la explicación de la solución propuesta pondremos un ejemplo de un párrafo de un discurso del famoso Denzel Washington en los premios de Imagen NNAACP.

**“Entonces, sigue trabajando. Sigue esforzándote. Nunca te rindas. Cáete siete veces, levántate ocho. La facilidad es una mayor amenaza para el progreso que las dificultades. La facilidad es una mayor amenaza para el progreso que las dificultades. Entonces, sigue moviéndote, sigue creciendo, sigue aprendiendo. Nos vemos en el trabajo”.**

En el párrafo anterior básicamente lo que está diciendo el famoso es que está motivando a otros para que trabajen duro y nunca se rindan. Para resumir el párrafo anterior se utilizan las técnicas basadas en NLP mencionadas anteriormente, por las que para hacer el resumen se deben de realizar los siguientes pasos.

2.-Convertir párrafos a oraciones

Primero necesitamos convertir todo el párrafo en oraciones es dividir el párrafo cada vez que se encuentra un período. Entonces, si dividimos el párrafo en discusión en oraciones, obtenemos las siguientes oraciones:

* Así que sigue trabajando
* Sigue esforzándote
* Nunca te rindas
* Cáete siete veces, levántate ocho
* La facilidad es una mayor amenaza para el progreso que las dificultades
* La facilidad es una mayor amenaza para el progreso que las dificultades
* Así que sigue moviéndote, sigue creciendo, sigue aprendiendo.
* Nos vemos en el trabajo

3.-Preprocesamiento de texto

Después de convertir el párrafo en oraciones, debemos eliminar todos los caracteres especiales, detener palabras y números de todas las oraciones. Después del preprocesamiento, obtenemos las siguientes frases:

sigue trabajando

sigue esforzándote

nunca dar

caer siete veces conseguir ocho

aliviar la amenaza mayor progreso dificultades

aliviar la amenaza mayor progreso dificultades

sigue moviendo sigue creciendo sigue aprendiendo

ver trabajo

4.-Tokenizado las oraciones

Necesitamos tokenizar todas las oraciones para obtener todas las palabras que existen en las oraciones. Después de tokenizing las oraciones, obtenemos una lista de las siguientes palabras:

['mantener',

 'trabajando',

 'mantener',

 'esforzarse',

 'Nunca',

 'dar',

 'otoño',

 'Siete',

 'hora',

 'obtener',

 'ocho',

 'facilitar',

 'mayor',

 'amenaza',

 'Progreso',

 'privación',

 'facilitar',

 'mayor',

 'amenaza',

 'Progreso',

 'privación',

 'mantener',

 'Moviente',

 'mantener',

 'creciente',

 'mantener',

 'aprendizaje',

 'ver',

 'trabajo']

5.-Encontrar la frecuencia ponderada de ocurrencia

A continuación, necesitamos encontrar la frecuencia ponderada de las ocurrencias de todas las palabras. Podemos encontrar la frecuencia ponderada de cada palabra al dividir su frecuencia por la frecuencia de la palabra que aparece con mayor frecuencia. La siguiente tabla contiene las frecuencias ponderadas para cada palabra:

| Palabra | Frecuencia | Frecuencia ponderada |
| --- | --- | --- |
| facilitar | 2 | 0.40 |
| ocho | 1 | 0,20 |
| otoño | 1 | 0,20 |
| obtener | 1 | 0,20 |
| dar | 1 | 0,20 |
| mayor | 2 | 0.40 |
| creciente | 1 | 0,20 |
| privación | 2 | 0.40 |
| mantener | 5 | 1.00 |
| aprendizaje | 1 | 0,20 |
| emocionante | 1 | 0,20 |
| Nunca | 1 | 0,20 |
| Progreso | 2 | 0.40 |
| ver | 1 | 0,20 |
| Siete | 1 | 0,20 |
| esforzarse | 1 | 0,20 |
| amenaza | 2 | 0.40 |
| hora | 1 | 0,20 |
| trabajo | 1 | 0,20 |
| trabajando | 1 | 0,20 |

Dado que la palabra "mantener" tiene la frecuencia más alta de 5, por lo tanto, la frecuencia ponderada de todas las palabras se ha calculado al dividir su número de ocurrencias por 5.

6.-Reemplazar palabras por frecuencia ponderada en oraciones originales

El paso final es conectar la frecuencia ponderada en lugar de las palabras correspondientes en las oraciones originales y encontrar su suma. Es importante mencionar que la frecuencia ponderada de las palabras eliminadas durante el preprocesamiento (palabras de detención, puntuación, dígitos, etc.) será cero y, por lo tanto, no es necesario agregarlas, como se menciona a continuación:

| Frase | Suma de frecuencias ponderadas |
| --- | --- |
| Así que sigue trabajando | 1 + 0.20 = 1.20 |
| Sigue esforzándote | 1 + 0.20 = 1.20 |
| Nunca te rindas | 0.20 + 0.20 = 0.40 |
| Cáete siete veces, levántate ocho | 0.20 + 0.20 + 0.20 + 0.20 + 0.20 = 1.0 |
| La facilidad es una mayor amenaza para el progreso que las dificultades | 0.40 + 0.40 + 0.40 + 0.40 + 0.40 = 2.0 |
| La facilidad es una mayor amenaza para el progreso que las dificultades | 0.40 + 0.40 + 0.40 + 0.40 + 0.40 = 2.0 |
| Así que sigue moviéndote, sigue creciendo, sigue aprendiendo. | 1 + 0.20 + 1 + 0.20 + 1 + 0.20 = 3.60 |
| Nos vemos en el trabajo | 0.20 + 0.20 = 0.40 |

7.-Ordenar oraciones en orden descendente de la suma

El paso final es ordenar las oraciones en orden inverso de su suma. Las oraciones con las frecuencias más altas resumen el texto. Por ejemplo, mire la oración con la suma más alta de frecuencias ponderadas:

*Así que sigue moviéndote, sigue creciendo, sigue aprendiendo.*

Puedes juzgar fácilmente de qué se trata el párrafo. De manera similar, puede agregar la oración con la segunda suma más alta de frecuencias ponderadas para tener un resumen más informativo. Echa un vistazo a las siguientes oraciones:

*Entonces, sigue moviéndote, sigue creciendo, sigue aprendiendo. La facilidad es una mayor amenaza para el progreso que las dificultades.*

Estas dos oraciones dan un resumen bastante bueno de lo que se dijo en el párrafo.

# **II. Marco teórico**

### Antecedentes (Por qué creo que mi solución Propuesta va a funcionar). Algo parecido a lo que estoy haciendo o lo mismo, pero con variaciones, ¿de dónde saqué mi solución propuesta va a funcionar.

Investigando acerca de los resúmenes automáticos y de sus inicios, han tenido una gran evolución desde que se comenzó la idea.

En los años finales de 1950 y 1960 sugería que generar resúmenes de texto por computadora era viable, aunque no sencillo.

En el año de 1960 aparecen algunas bibliotecas en EUA, donde se quería agregar digitalmente el contenido de la biblioteca para facilitar su búsqueda, al existir límites de volumen de almacenamiento fue imposible almacenar todos los documentos, para ello se creó la idea de realizar resúmenes de estos, para quien los consultara pudiera conocer más acerca de ellos. Algunos documentos ya proporcionaban este pequeño resumen, pero otros no, esto hizo que se despertará el interés por técnicas que permitieran realizar resúmenes automáticos.

En los 30 años siguientes se continuó usando esa técnica de resúmenes en otros ámbitos, pero hasta los años 90’s con el surgimiento y la extensión de WWW (World Wide Web) y la aparición de nuevas fuentes de información, es cuando los resúmenes automáticos cobran más importancia.

Actualmente, es muy diferente a la situación de los años 60’s ya que ahora contamos con más capacidad de almacenamiento y más barata y seguramente irá incrementando durante los años siguientes. El problema que surgió hoy en día es filtrar la información abundante en pequeños resúmenes para que esta puede ser entendible. De esto surge la importancia de inventar nuevas técnicas para poder solucionar este problema. Tras algunas décadas, los avances en el PLN, texto en línea y especialmente en la web, han renovado el interés y las técnicas en la generación automática de resúmenes de texto.

### Trabajos relacionados. (Comparación con los trabajos y el nuestro)

Para cualquier problema planteado con el análisis y procesamiento de textos existe un campo amplio de información en el cual nos basamos para realizar nuestro trabajo y esperamos que con lo realizado los resultados obtenidos contribuyan y aporten a dicho tema. En el trabajo que se presenta se mostrarán diversos trabajos que nos permitirán o aportarán en mayor o menor medida recursos necesarios para el desarrollo de nuestra aplicación.

Uno de los principales documentos que consultamos fue la tesis titulada *Sistema Resumidor-Traductor Automático* del Ing. Jouse Antonio Careaga Moya publicada en el año 2012 por la UNAM, la cual nos permite reforzar la parte que el resumidor que construyamos debe contener las ideas principales y dar una idea completa del texto. Nos guiamos en este trabajo ya que emplea la librería NLTK la cual brinda la posibilidad de segmentar el texto en enunciados, separados por comas, puntos o saltos de línea, por lo tanto, podemos indicar que el trabajo citado nos orienta en cómo llevar a cabo el proceso de elaboración ya mencionado. [[1]](#footnote-1)

A lo largo de que el lector conozca más sobre el trabajo presentado se dará cuenta que el trabajo citado en el párrafo anterior es de gran ayuda para la solución de nuestro planteamiento, sin embargo, se analizaron otros trabajos como el de Pedro Luis Mateo, José Carlos González, Julio Villena y José Luis Martínez titulado *Un sistema para resumen automático de textos en castellano* en el cual se presenta las técnicas que deben ser aplicadas para detectar palabras clave, dar ponderaciones con el fin de no solo crear extracciones simples y sin coherencia ya que el fin de un resumen es mostrar la idea del texto basado en el original y no sólo copiar las frases idénticamente. El trabajo nos resulta útil e interesante ya que consideramos que nos permite evaluar si el resumen creado está cumpliendo con el concepto, si este está siendo analizado correctamente para crear finalmente un resumen de la mejor calidad, los cuales son puntos que hemos considerados importantes y necesarios en nuestro trabajo. [[2]](#footnote-2)

Consideramos que los dos trabajos citados son los más relevantes ya que contienen una amplia descripción y brindan una o más soluciones al problema planteado, además que coinciden con la solución propuesta.

Algunos de los trabajos relacionados son los mencionados anteriormente, para el primero de ellos se implementa Python y algunas de las librerías que son útiles para la creación de un resumidor de textos, el segundo es un trabajo internacional que de igual manera nos permite tener una guía para dar solución a nuestro problema. En general como ya se mencionó en un inicio contamos con una amplia variedad de artículos y trabajos que nos proporcionan ideas y soluciones que debemos implementar en nuestro propio proyecto.

Además, tenemos como guía las ventajas que son definidas en todas las aplicaciones disponibles online o en aplicaciones, las cuales nos permiten a través de una comparativa identificar cuáles son los principales puntos con los que debe contar nuestro programa.

**Descripción.**

**-Resumen de textos automático.**

Un resumen de textos automático es aquel que se ejecutará por el usuario pero que no requerirá la intervención de este en la generación del documento final, dicho documento debe contener una idea concisa del documento original el cual será procesado por diferentes funciones del programa clasificando las ideas más relevantes y ordenándolas con el fin de generar un documento que sea lo más coherente posible. El hecho de generar un resumen de textos es una tarea que corresponde a la disciplina del procesamiento de lenguaje natural.

**-Procesamiento de Lenguaje Natural.**

Corresponde a una disciplina que emplea diversas técnicas con el fin de que el lenguaje natural, es decir el lenguaje hablado por los humanos sea procesado por las computadoras o una máquina, dicho problema no tiene una solución fácil ya que se requiere que el programador conozca tanto el lenguaje natural como el de la máquina con el fin de hacer que una computadora genere los procesos necesarios con el fin de obtener resultados que los humanos realizamos como el comparar textos, resumir textos o encontrar una idea principal.

**-Recuperación de la información.**

Indica el problema que se tiene con poder recuperar la información extraída por el documento en un inicio, con proporcionar la información solicitada y mostrarla de manera exitosa.

**-Técnicas para generar un resumen de textos.**

Podemos encontrarnos con diversas técnicas que van desde generarlo a partir de temas específicos, otro ejemplo es introducir un aprendizaje para que compare posteriormente los textos solicitados o bien los que se basan en análisis discursivos a partir de árboles, por ejemplo, en el trabajo presenta se pretende generar un análisis a partir de los tokens identificados a través de NLTK de Python. En general en una aplicación de resumen de textos veremos como punto clave el uso de la lingüística y sus conceptos la cual nos permitirá identificar los elementos necesarios en un texto y a partir de esto generar un resumen lo más adecuado y completo posible.

# **III. Método experimental**

### III.I. Datos ¿Cómo se obtuvieron?

Los datos para el desarrollo de nuestro proyecto se obtuvieron de diversos trabajos ya realizados y consultados para basarnos de ahí y realizar nuestro proyecto. Dado que Python cuenta con la biblioteca nltk, nosotros decidimos usar este lenguaje por su utilidad y facilidad para obtener más rápido y eficientes los resultados.

La mayor parte de los datos aquí presentados fueron obtenidos de las fuentes ya consultadas y mencionadas en la sección del Marco Teórico del proyecto y de la solución propuesta.

### III.II. Descripción de su método.

Para la descripción de nuestro método es importante conocer lo que realizan cada una de las bibliotecas que utilizamos del lenguaje de programación Python. A continuación, se detallará que es lo que hacen estas librerías.

**III.III. Librería NLTK**: Por sus siglas en inglés (Natural Language Toolkit) , es una plataforma que se utiliza para realizar programas para análisis de texto. Esta plataforma fue liberada por Steven Bird y Edward Loper, gracias a un concurso de lingüística computacional en la Universidad de Pensilvania en el año 2001.

Es un conjunto de herramientas para el procesamiento del lenguaje natural, simbólico y estadístico para el lenguaje de programación Python. También incluye demostraciones gráficas, pero en este proyecto no se verá su utilidad.

Este está destinado para apoyar a la enseñanza como una herramienta de estudio individual y como plataforma en los sistemas de investigación de PLN y en áreas muy relacionadas como lingüística empírica, las ciencias cognitivas, inteligencia artificial y aprendizaje automático.

Para su instalación en Windows 10 se debe de realizar desde el cmd, escribiendo

pip install nltk

Para su instalación en Linux distribución Ubuntu se ejecuta el comando desde la terminal.

sudo apt-get update

sudo apt-get install python-nltk

**III.IV. ¿Por qué utilizar NLTK?**

Primero debemos entender que el lenguaje natural es el que usamos los humanos para comunicarnos diariamente, a diferencia de los lenguajes de programación. El análisis de lenguaje natural puede ir desde solo contar frecuencias o bien desde analizar expresiones completas y darles utilidad.

**III.IV.I . ¿Qué nos proporciona NLTK?**

NLTK nos proporciona herramientas para el procesamiento del lenguaje natural, es decir, el lenguaje que utilizamos diariamente para la comunicación humana. El procesamiento del lenguaje natural puede ser sencillo como el caso de contar frecuencias como es el ejemplo del proyecto presentado en este trabajo, que muestra como a partir del conteo de frecuencias y valorización de oraciones se genera un resumen o bien el hecho de tomar los resultados para destacar el contenido de una oración, su relevancia, entre otros factores.

Por lo que para este trabajo se recalca nuevamente que se hará uso de Natural Lenguaje Toolkit (NLTK ) la cual es una biblioteca de código abierto de Python, con el fin de manipular los datos para obtener un resultado útil.

**III.IV.II. NLTK con 4 objetivos principales.**

|  |  |
| --- | --- |
| **Sencillez.** | Brindar a los usuarios un uso práctico sin agobiarlos con la depuración asociada al procesamiento del lenguaje natural. |
| **Consistencia.** | Proporcionar interfaces y estructuras fáciles de deducir. |
| **Extensibilidad.** | Proporcionar estructura en la que los nuevos módulos de software puedan acomodarse fácilmente, tomando en cuenta enfoques competitivos. |
| **Modularidad.** | Proporcionar componentes que pueda usarse de manera independiente sin necesidad de comprender el resto del kit de herramientas. |

**III .IV. III. ¿Cómo vemos que se cumplen los objetivos de NLTK en nuestro proyecto?**

Para el proyecto implementado del cual se habla en el documento presente identificamos que cumple al menos con tres objetivos como lo son la sencillez la cual la identificamos porque los paquetes como stopwords o punkt son fáciles de asociar a la actividad que proveen bien la consistencia toda nos resultó fácil de utilizar y el modularidad podemos considerarla como verdadera ya que de NLTK solo usamos dos herramientas que son stopwords y punkt.

**III.V. ¿Cómo funciona NLTK en nuestro proyecto?**

Nos ayuda a realizar un conteo de las veces que aparece una palabra en un texto, el ver cuántas veces se encuentra en un texto podría no resultarnos aún del todo útil. Ya que podría arrojarnos un resultado sin sentido de palabras y números. Para poderle dar sentido a las palabras es necesario ver su importancia en el texto completo y por oración. La manera que esto posible es a partir de las herramientas que nos proporciona nltk para el desarrollo de nuestro proyecto.

**III.V.I. Token.**

En el programa realizado se tokenizará todo el texto en un inicio, para lo cual haremos uso del módulo punkt y en específico de word\_tokenizer, el cual nos dará subcadenas de una cadena, estos tokens son cada una de las formas que conforman al texto en general como lo pueden ser palabras, puntuación incluso números o caracteres numéricos, word\_tokenize recibe lo que queremos tokenizar y el idioma.

Para el proyecto se utilizó word\_tokenize el cual nos arroja cada una de las formas que se encuentren en un texto como se muestra en la imagen siguiente, donde apreciamos una lista que contiene cada una de las subcadenas que se pueden generar a partir del texto.



**III.V.II. Frecuencias**

Es necesario indicar que para el proyecto en curso se hará uso de la frecuencia ponderada que es el valor obtenido entre la frecuencia de una palabra dividida entre el numero mayor de frecuencia de una palabra dentro del texto.

**III.V.III. ¿Qué hay con la tabla de frecuencias?**

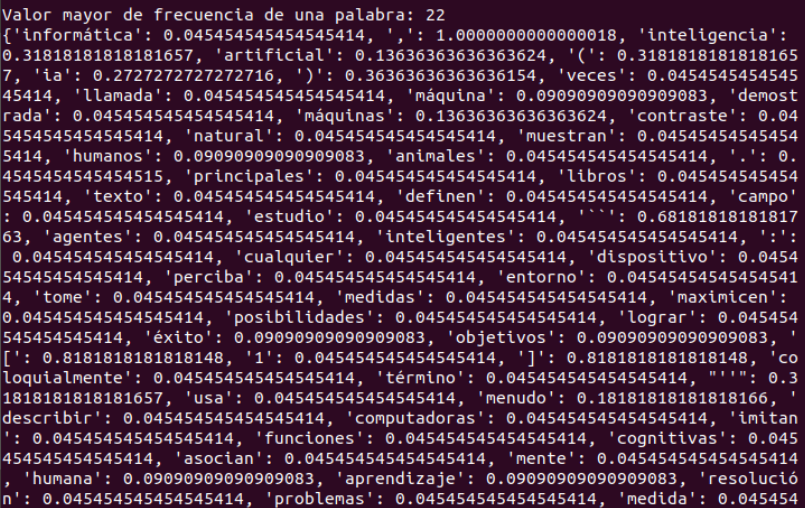
La tabla de frecuencias es un diccionario, el cual se usa con el fin de poder almacenar una clave con su respectivo valor en este caso las claves son cada una de las formas o tokens que están presentes en el texto y los valores son los números correspondientes a las frecuencias, como se muestra en la imagen siguiente.



Como podemos observar se está ocupando un diccionario esto debido a que tenemos lo que se llama **lista simple**, es decir un token acompañado de su frecuencia, podemos encontrar otro tipo de listas como las que describen en la tabla siguiente.

|  |  |  |
| --- | --- | --- |
| **Tipo de lista.** | **Descripción.** | **Ejemplo.** |
| **Simple.** | Token acompañado de su frecuencia. | Clasificadas: 1 |
| **Canónicas.** | Lista ordenada alfabéticamente y a cada línea le suceden las palabras que corresponden a dicha forma canónica. | el trabajo de Juan López Chávez y Marina Arjona de *Lexicometría y fonometría del Primero Sueño de Sor Juana Inés de la Cruz*. |
| **Lemas.** | Ordenadas por lemas o raíces , sin importar la parte de la oración de la que trate. | Se puede agrupar el lema nación en:  Naciones, nacionalizar, nacionalmente y nacionalización. |
| **Dos o más palabras.** | No son palabras simples sino pares, terceras de palabras que ocurren contiguamente. Cada una se acompaña de su frecuencia. | Informática:1 sin: 2 amor:4 |
| **Partes de la oración.** | Lista que solo contenga partes de la oración. | Se usan solo referencias de categorías para fines estadísticos. |

Una vez obtenida la frecuencia absoluta de nuestras palabras o ya tokens, obtenemos la frecuencia ponderada de cada una de ellas que como ya se mencionó corresponde al número total de apariciones en el texto entre el mayor número correspondiente a una palabra, dichas frecuencias ponderadas se muestran en la siguiente imagen.



**III.V.III.I. ¿De qué nos sirve la forma en que se crea la tabla de frecuencias?**

Además la tabla que podemos apreciar en este proyecto tiene un **orden por aparición,**  ya que si nos percatamos no veremos los tokens con mayor frecuencia primero o en orden inverso, tampoco se muestran de manera alfabética o por la longitud de la palabra. Lo que nos indica que la tabla se va construyendo según vayan apareciendo en el texto.

Entonces el hecho de que se vayan acomodando las palabras por aparición y por tanto llenando de esta manera la tabla de frecuencias nos beneficia en el hecho de que las palabras solo aparecerán una vez de acuerdo con el texto y de ahí el número de veces que estén el texto, lo que se asemeja a la manera en que lo haríamos humanamente ya que primero contabilizaríamos y finalmente sol escribiríamos la palabra con su respectivo valor total. Adicionalmente nos permite conocer la distribución de las palabas y ver so esta indica un cambio en el sentido de la oración por el hecho de aparecer en un lugar distinto. Independientemente del tipo de lista con el que contemos nos permitirá en sentido general del proyecto conocer cuál es el sentido de la palabra, sin embargo los tipos de orden que les demos nos harán más fácil su análisis o lo complicarán en menor medida.

**III.VI. Problemas al realizar los diccionarios o listas de palabras.**

Podemos enfrentarnos a la perdida de información relevante o no, es decir, podemos obtener listas de palabras que contengan guiones que son esenciales en palabras compuestas, signos de puntuación eliminados que quiten sentido a las oraciones e incluso eliminación de mayúsculas que correspondan a determinadas siglas. Además de la eliminación podemos tener confusión en el sentido de números que estén escritos con letras y no con números, lo que quitaría el sentido a dichas cifras.

Entonces la pregunta relevante es ¿Qué debemos eliminar y de acuerdo con qué?. Para lo cual se han planteado soluciones como:

* Pre- procesamiento del texto: en donde se realcen todas las palabras incluyendo fechas más importantes, así como el uso de expresiones regulares.
* Ajustar el programa: indica que el programa usado o método empleado deberá ajustarse a lo que uno decida conservar o eliminar, dando uno el propio sentido.

**III.VI.I. ¿Cómo saber que se ignora y a qué se le da importancia?**

En un proyecto que desea generar un resumen como es el caso presentado, no es útil darle importancia a todas las palabras del texto que un usuario nos proporcione. Muchas de las veces los humanos al hacer un resumen creemos que es útil toda la información que nos encontramos al ir leyendo, pero no es así, al realizar un resumen debemos tener en cuenta el hecho de que solo queremos dar a conocer la idea principal, lo más importante o relevante de todo el texto.

Los resúmenes de los seres humanos contienen palabras conectoras que repetimos con frecuencia las cuales variarán de acuerdo con el idioma que se use. Para poder dar énfasis a las palabras más significativas de un texto en este proyecto se emplea la librería de python stopwords, lo anterior no nos quiere indicar que no sean necesarias sino que se trata en un análisis de este tipo de encontrar aquellas que dan un aporte sustancial.

Los stopwords son un conjunto de preposiciones, conjunciones, artículos, adverbios, pronombres y algunos verbos y forman alrededor del 30% al 40% de los textos en español, este tipo de palabras se les caracteriza como palabras vacías que si bien no se eliminan, se omiten para poder obtener lo más relevante.

Entonces un stopwords funcionará de la siguiente forma.

|  |
| --- |
| En informática, la inteligencia artificial (IA). |

Al ejecutar el stopwords, lo que obtenemos es:

informática

,

inteligencia

artificial

(

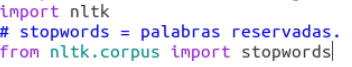
ia

)

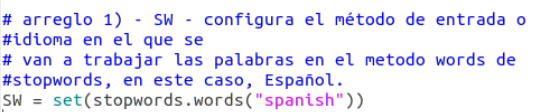
Lo que nos indica que hemos descartado el siguiente conjunto: {En, la}

Que corresponde a artículos y preposiciones propiamente.

El uso de stopwords se emplea de la siguiente manera para python, primero importamos el corpus de stopwords, es decir el conjunto de palabras donde se encuentran contenidas todas aquellas preposiciones, conjunciones, artículos, etc.

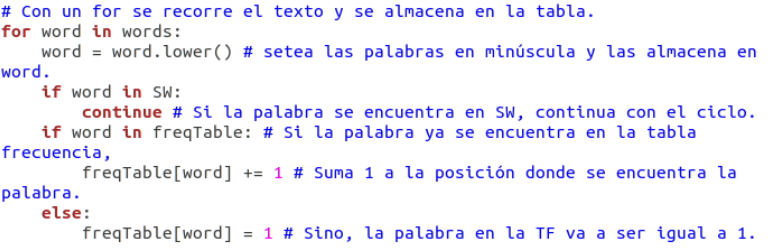


Una vez importado se configura el idioma, para los casos implementados en este proyecto son español e inglés, como se muestra en las siguientes imágenes.



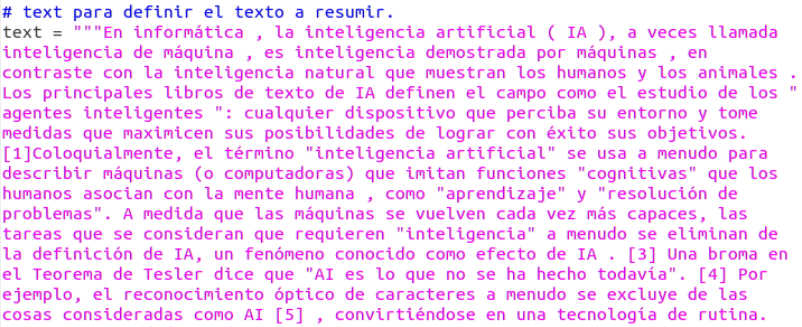


Posteriormente se crea un diccionario de frecuencias llamado tabla de frecuencias que se ha mencionado en la sección anterior en este se guardarán las palabras y su valor asociado a estas, de acuerdo con la condición de encontrarse o no en SW que corresponde al corpus de las stopwords, de la manera siguiente.

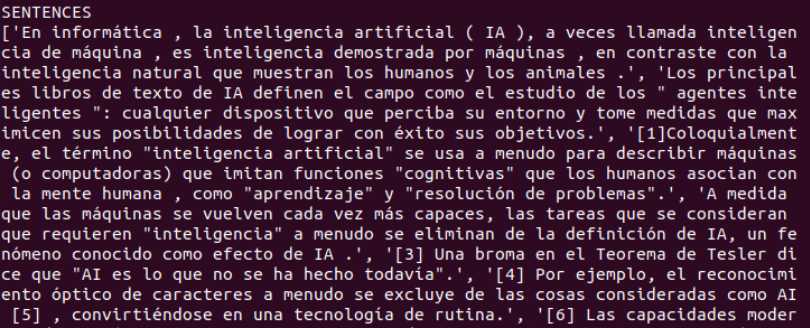


**III.VI.II. ¿Cómo generar las oraciones?**

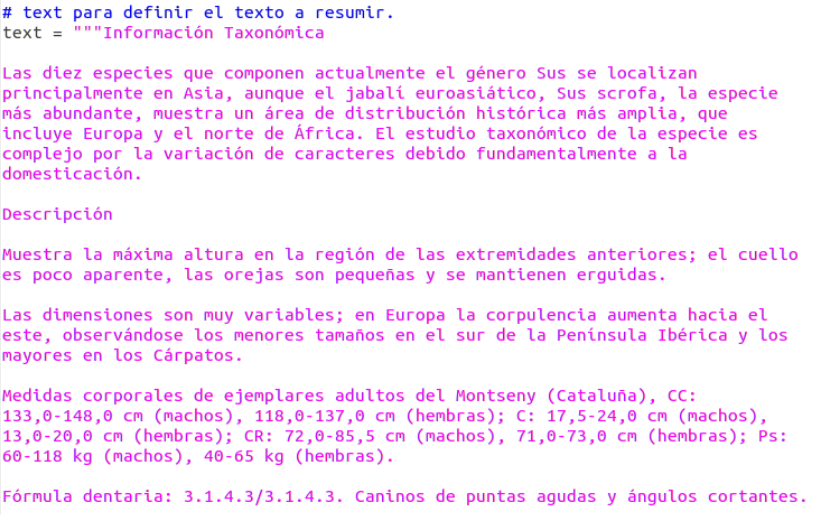
Para el hecho de generar las oraciones o crear una lista donde estas estén albergadas se usa sent\_tokenize la cual permite al igual que Word\_tokenize separar, aunque en este caso serán oraciones. Lo que nos dará un resultado de una lista que contenga las oraciones agrupadas de acuerdo con los signos de puntuación del texto original de dónde son tomadas. Por ejemplo para el caso de un texto como el de la imagen obtendremos al utilizar sent\_tokenize una lista como la que se muestra una imagen posterior. Las oraciones al igual que las palabras por si solas tendrán un valor ponderado que es igual a su frecuencia absoluta más su frecuencia ponderada obtenida en paso previos ya descritos.



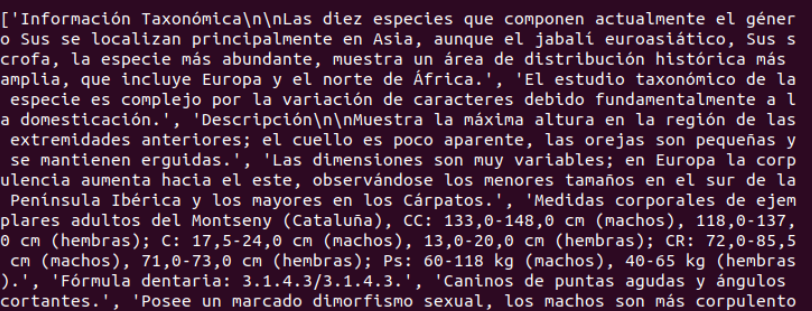
Se obtienen las siguientes oraciones.



Además permite separar las oraciones de acuerdo con los saltos de línea o tabulaciones que se indican en un texto, como en el caso del texto siguiente.

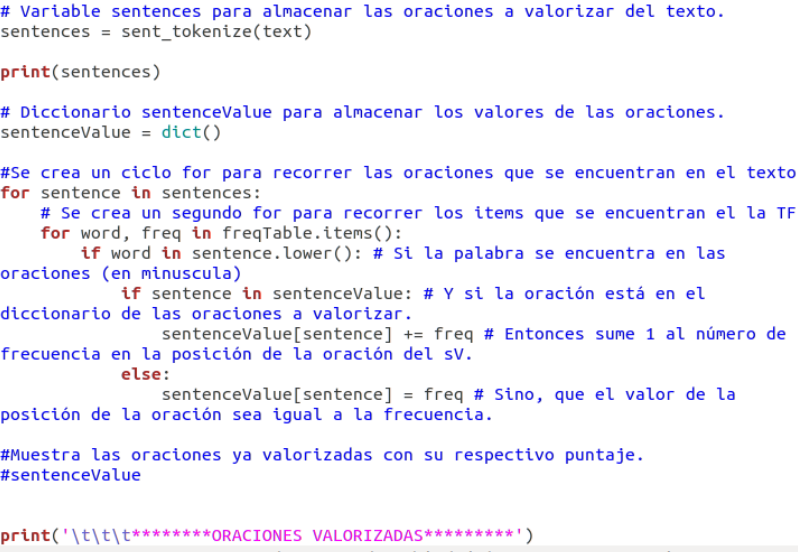


Obteniendo.

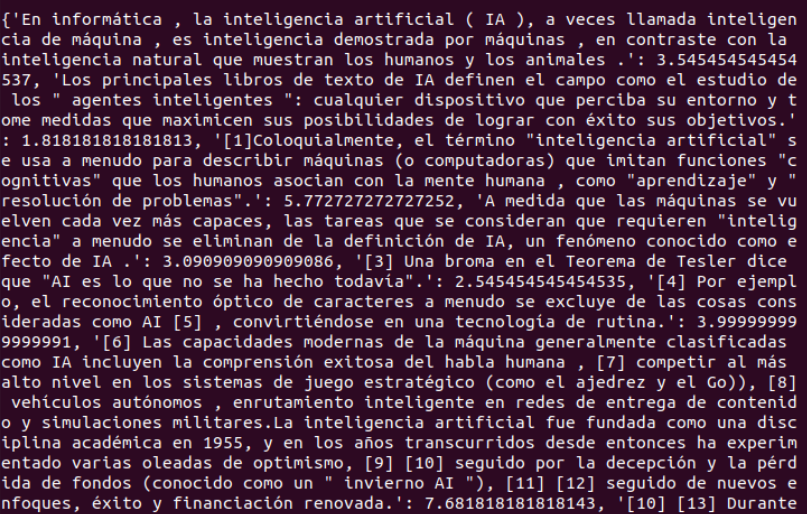


Una vez que tenemos un conjunto de oraciones que aparecen en nuestro texto, es de importancia saber cuál es el proceso que determina como se valorizarán cada una de las oraciones.

Para esto se recorren las oraciones que se encuentran en el texto, posteriormente se recorren las partes que componen a la tabla de frecuencias, entonces si la palabra está en las oraciones en minúscula y además en diccionario de oraciones a valorizar se suma uno a su frecuencia en la posición, en caso contrario el valor asignado para si posición corresponde al de su frecuencia absoluta, de esta forma se obtienen las oraciones con valor, como se muestra en la imagen siguiente.



A continuación se muestran las oraciones ya valorizadas, el ejemplo mostrado es el del texto de la informática, se muestran las oraciones de acuerdo con los valores de frecuencia ponderada obtenidos con anterioridad.



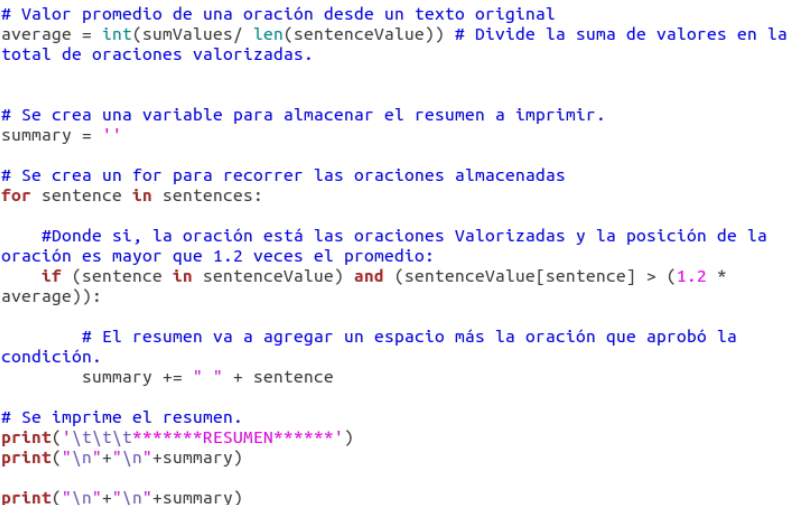
**III.VII. ¿Cómo determinar que oraciones se añaden al resumen final?**

Para llegar a crear el resumen una vez que hemos dado valor a cada oración será necesario hacer la suma total de los valores que se le han dado a estas, para lo cual se recorrerán una por una y en caso de que la oración pertenezca a las oraciones valorizadas sumaremos el valor a una variable que contenga el valor total.

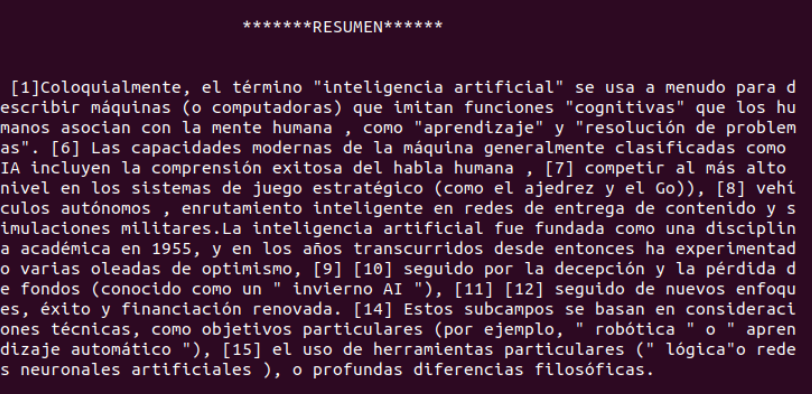
Una vez que hemos obtenido la suma total lo que queda es obtener el promedio dividiendo el valor total entre el número de oraciones valorizadas. Una vez hecho esto podemos proseguir a añadir las oraciones al resumen.

Las oraciones se añadirán siempre y cuándo el valor ponderado de la oración sea mayor a 1.2 por el valor obtenido de tal manera que al multiplicar el promedio por dicho valor priorizamos el que solo las oraciones con mayor se añadan.

El proceso y resultado se muestran en las siguientes imágenes.



El resultado es:



Y es de esta forma que podemos obtener un resumen.

**III.VIII. ¿Qué hay de la interfaz gráfica?**

**Librería PYQT5:** Es una biblioteca gráfica Qt para el lenguaje de programación Python, está desarrollada por la firma británica “Riverbank Computing”. Esta es muy utilizada para la creación de interfaces gráficas en Python, ya que es muy fácil de utilizar y entender en poco tiempo.

Para su instalación en Windows 10 se debe de realizar desde el cmd, escribiendo

$pip install PyQt5

Para su instalación en Linux distribución Ubuntu se ejecuta el comando desde la terminal.

$pip3 install PyQt5-sip

Descomprimiéndolo

$python3 configure.py

$make

$make install

Para su implementación lo que se realizó, fue seguir las siguientes tareas de manera sintetizada.

1. Importar los métodos para trabajar con NLTK.
2. Importar stopwords para poder checar posteriormente las palabras reservadas.
3. Descargar los métodos para palabras reservadas y tokenizar.
4. Configurar el método de entrada para el idioma requerido.
5. Definir el texto que queremos resumir.
6. Tener un arreglo para almacenar los tokens del texto.
7. Crear una tabla para las frecuencias de las palabras.
8. Recorrer el texto y añadir las palabras con su frecuencia a la tabla de frecuencia y actualizarla si es necesario.
9. Mostrar la tabla de frecuencias.
10. Crear un diccionario para almacenar los valores de las oraciones.
11. Recorrer las oraciones en el texto y verificar si está en el diccionario y tabla de frecuencias y sumar puntos a la palabra.
12. Mostrar el valor por oración.
13. Evaluar las oraciones de acuerdo con el diccionario.
14. Obtener el valor promedio de la palabra de acuerdo con el texto original.
15. Añadir el texto a la variable de resumen.
16. Recorrer las oraciones almacenadas y si la oración tiene valor y su posición es mayor que 1.2 veces el promedio añadir al resumen.
17. Imprimir resumen.

### Descripción del experimento (Responder a la pregunta).

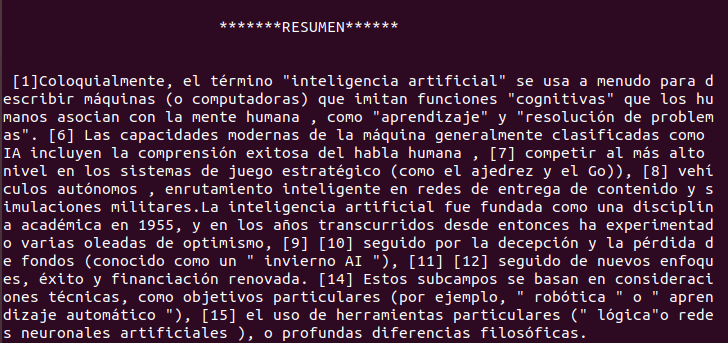
El experimento realizado fue verificar si el resumen obtenido era similar al que un ser humano puede hacer normalmente, es decir, si es coherente, si tiene estructura.

### Presentación de resultados.

Los primeros resultados aquí mostrados corresponden a los resultados de un resumen obtenido desde la terminal, como podemos observar aún no se muestra una interfaz gráfica para el usuario, pero estos resultados fueron los primeros que obtuvimos.

**•Versión en español.**

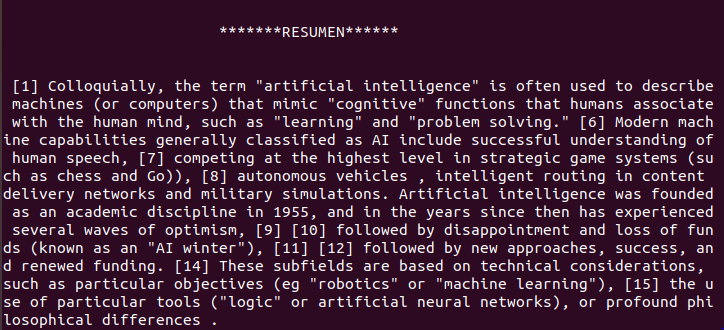
**-Resumen.**



**•Ingles.**

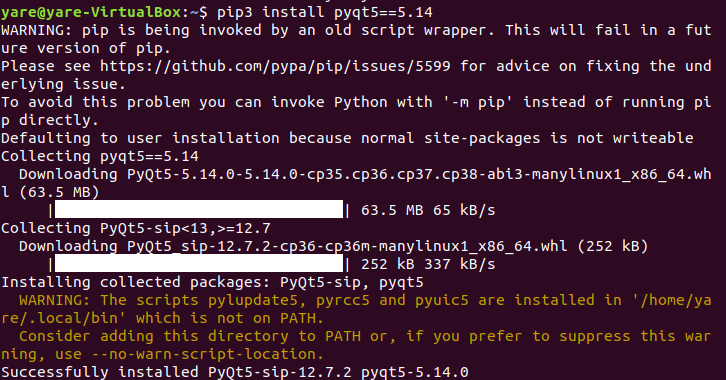
Podemos implementar distintos lenguajes siempre y cuando estos se encuentren dentro del stopwords.

**-Resumen.**



**Figura No.8. Resumen (inglés).**

**¿Qué hacer para tener una interfaz gráfica?**

Primero para poder tener una interfaz gráfica e importar en nuestro código Qt Core, QtGui, QtWidgets es necesario instalar PYQT5 librería que nos permitirá importar lo necesario para poder mostrar al usuario una interfaz.

**Figura No.9. Instalación Librería para la interfaz gráfica.**

**Resultados de la versión 2.**

**•Menú.**

**Para probar el proyecto, lo que se realiza es direccionarse dentro de la terminal en donde se encuentra guardado el proyecto y ejecutarlo para que nos habrá un menú en donde se podrá realizar el resumen.**



**Figura No.10. Ejecución del programa.**

Que nos genera la siguiente ventana.



**Figura No.11. Seleccionar que es lo que se desea realizar.**

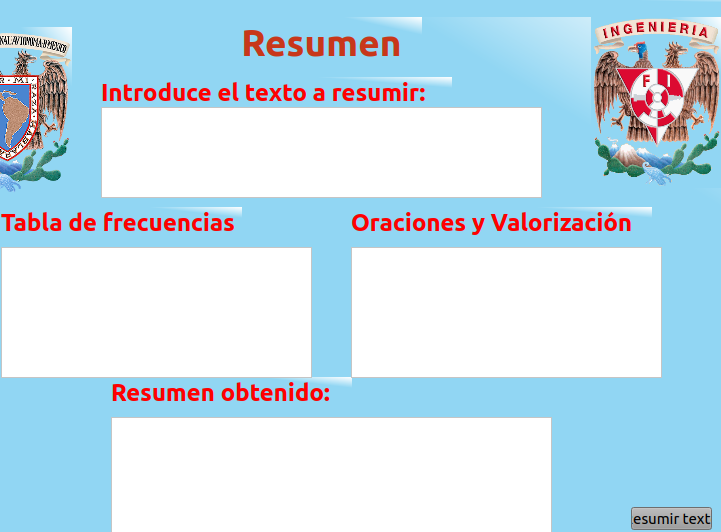
Al seleccionar el resumir texto podemos llegar a una nueva ventana donde podremos introducir nuestro texto a resumir. Como se muestra a continuación.

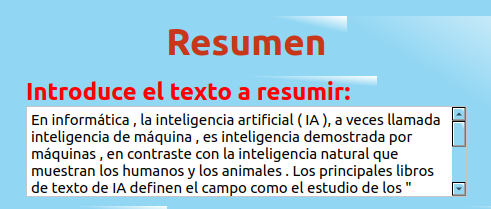
**Texto desde el ordenador.**

Una vez instalada nos debemos mover a la ruta donde se encuentra nuestra interfaz y ejecutar la ventana principal cuya extensión es .py. Como se muestra en la imagen siguiente. Esta es una manera de poder visualizarla o bien nos podemos dirigir a este punto desde la ejecución del menú mostrada en el punto anterior.



Con lo que obtendremos un resultado como el que se muestra a continuación, en la ventana introduciremos el texto a resumir y al presionar el botón de resumir obtendremos una tabla de frecuencias, oraciones valorizadas y el resumen. Así mismo veremos una ventana emergente que nos indicará que nuestro resumen está listo.



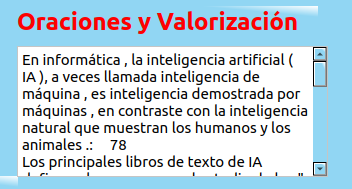




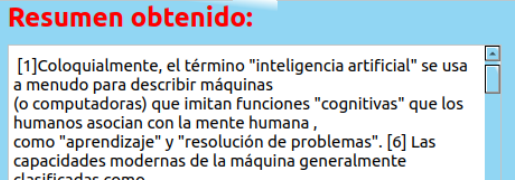
**Figura No.14. El resumen se ha realizado.**

Al presionar ok podremos apreciar los campos ya mencionados.

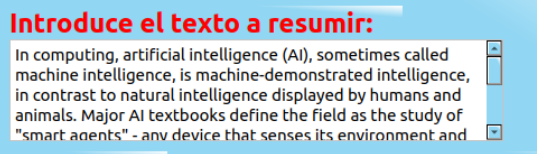
**Figura No.15. Tabla de frecuencias (Interfaz).**

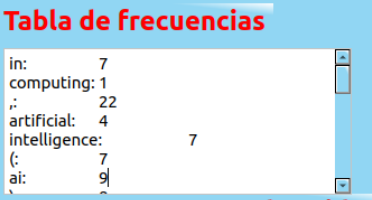
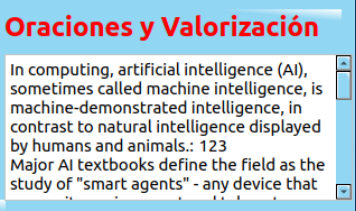


Resumen obtenido.

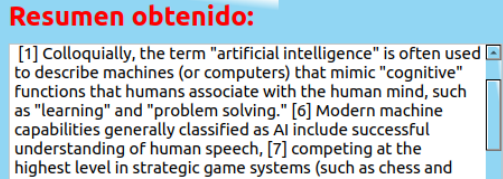


Para un resumen en inglés los resultados son los siguientes.



Del cual obtenemos su tabla de frecuencias, oraciones valorizadas y el resumen en general.

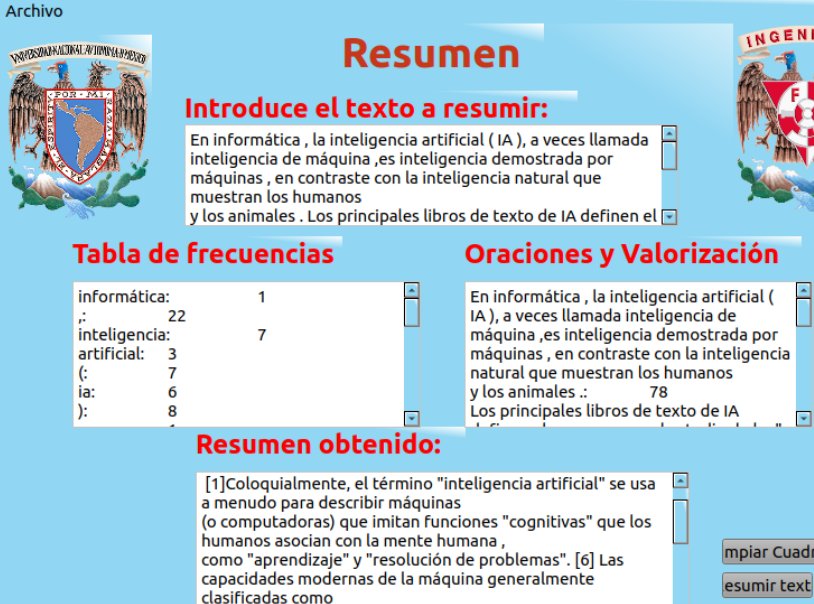
Obtención del resumen.



**Resultados de la versión 3.**

Una vez avanzado el proyecto se visualizaron las áreas de mejora de este y por tal razón para esta versión del proyecto se implementaron botones adicionales como el de limpiar cuadro ubicado en la parte inferior derecha, el cual permite eliminar el contenido de forma automática.

Una visión general de esta versión es la que se muestra en la imagen siguiente. Sigue contando con las ventanas de la versión 2 además de la ventana emergente que indica cuando se ha resumido un texto.



**Mejoras.**

Además nos brinda la posibilidad de poder guardar el resumen en archivo .txt, para lo cual nos debemos colocar en la parte superior izquierda presionar Archivo y Guardar en archivo, lo que nos generara en nuestra carpeta local tener el resumen.



Al ser guardado nos aparecerá una ventana como la siguiente.



Que nos indicará que se ha guardado satisfactoriamente, cada vez que demos clic en Guardar Archivo se creará un archivo que contendrá nuestro resumen, por lo que hay que tener cuidado de no guardarlo muchas veces.

# **IV. Conclusión**

### Discusión de resultados

Para darle una valoración de los resúmenes generados en nuestro proyecto la forma más común sería compararlo con los resúmenes realizados por un humano. Para ello existen varias técnicas para conocer si el resumen que se obtuvo es coherente o lo más coherente posible, una de ellas es la evaluación intrínseca y extrínseca. La intrínseca prueba el sistema de resumen en sí mismo, y la extrínseca prueba el resumen basado en la forma que se afectaría la realización de alguna otra tarea.

Las evaluaciones intrínsecas evalúan la coherencia y la informatividad de los resúmenes.

Las evaluaciones extrínsecas ponen a prueba el impacto del resumen de tareas como la evaluación de la pertinencia y la comprensión de lectura.

Existen otros métodos como los son los intertextuales, estos evalúan la salida de un resumen en específico. Los intra-textuales centran el análisis comparativo con varios sistemas de integración.

Hay infinidad de métodos por lo cuales podemos saber si nuestro proyecto es correcto, pero elaborar un resumen en muy subjetivo, ya que todas las personas realizamos los resúmenes de diferentes maneras, por ejemplo, no todos le damos la misma importancia a las cosas, algunas personas pueden considerar más valiosa cierta información que otra. Por lo tanto, si le damos la valoración comparando el resumen automático con el de un humano, segura existirán semejanzas, pero de igual manera diferencias.

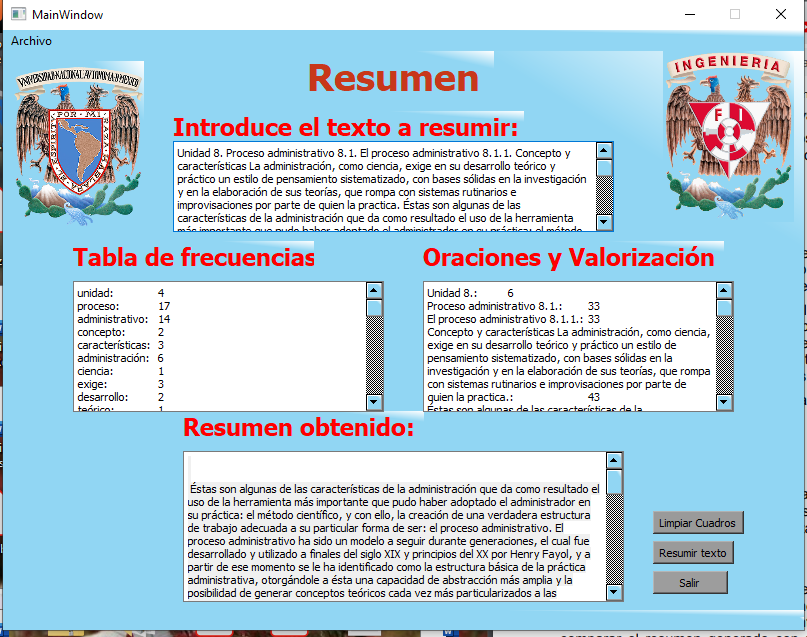
Sabiendo esto consideramos que nuestro proyecto lo que realiza es una valuación intrínseca ya que soló se evalúan la coherencia y la informatividad de los resúmenes.

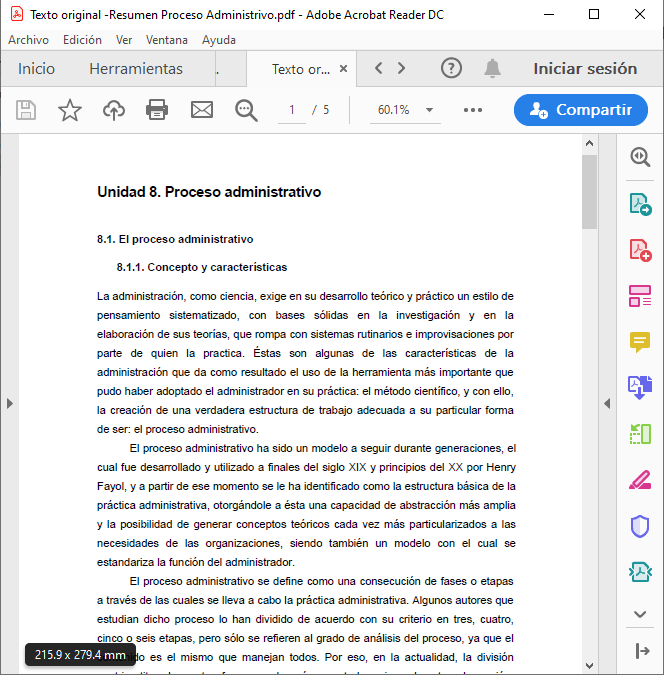
**Coherencia**: Dado que nuestro método para generar el resumen se enfoca en la extracción de la información, donde se extrae la información sin modificarla y se utilizan frases clave, donde el objeto es seleccionar frases enteras, para crear un breve resumen de párrafo. Puede que al tomar una frase se saque del contexto del texto y es posible que este deje de tener sentido en el lugar donde se introduzca, por lo tanto, de debe de verificar que exista coherencia.

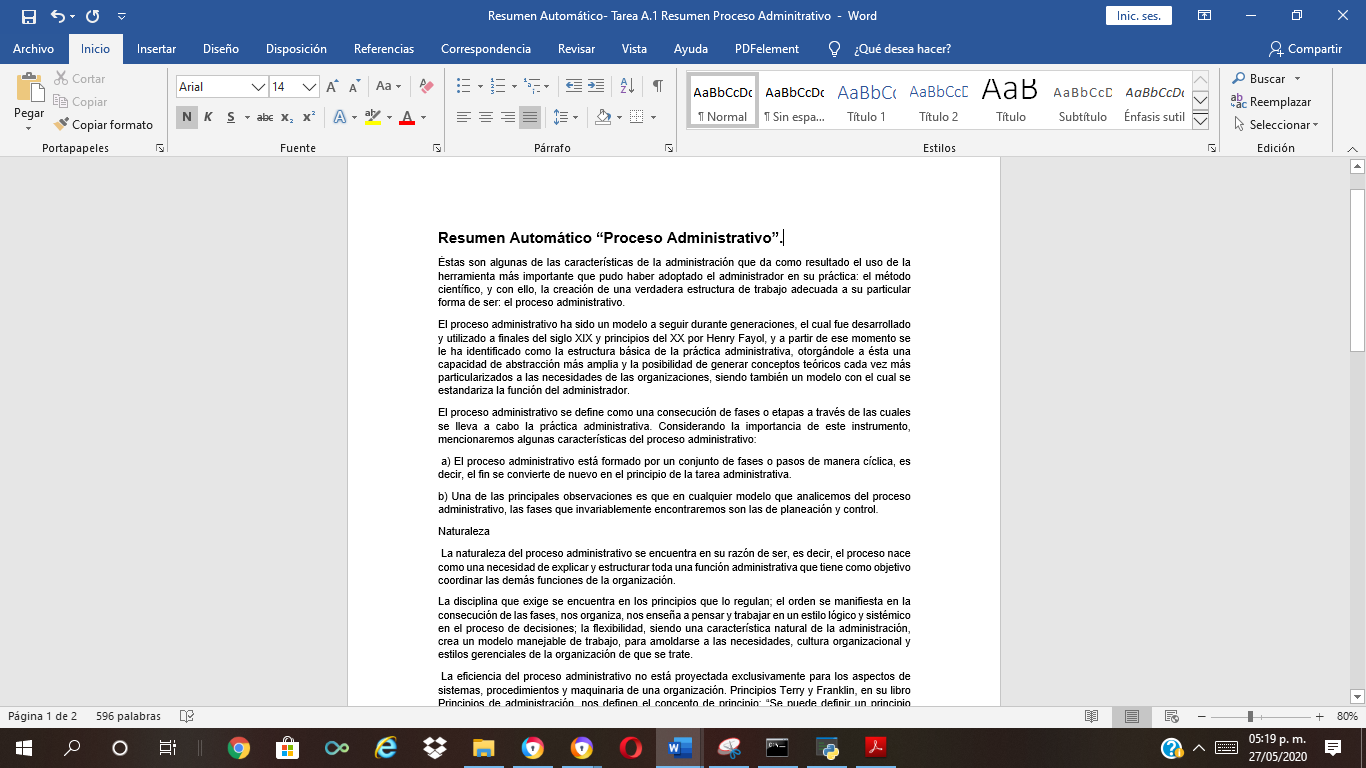
**Informativo:** Calcular el nivel de retención para dar un resultado de cuanta información de la fuente mantiene el resumen.

**Similitud del contenido:** Comparar la similitud del resumen generado con el resumen de referencia.

Tomando en cuenta un texto llamado “Texto original -Resumen Proceso Administrativo”, que se dejó como tarea realizar a un alumno del equipo, se consideró que el resumen generado fue correcto, ya que hubo coherencia en el resultado comparándolo con el texto original.



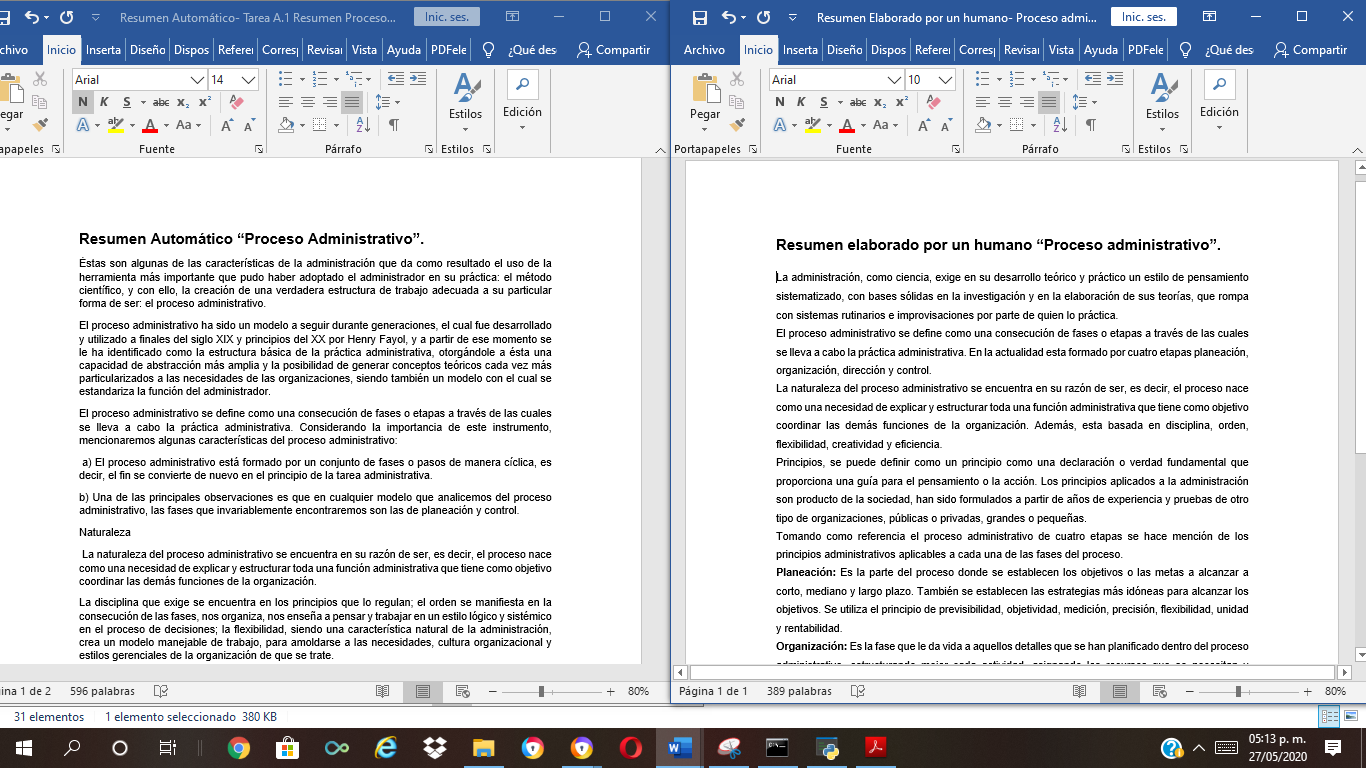




Probando nuestro proyecto con el método de comparación, lo realizamos con la de un humano. Desde nuestro punto de vista consideramos que esté método lo es el óptimo ya que existen muchas diferencias ya que todos pensamos de forma diferente.

Comparando nuestro resultado del resumen automático que generamos con el resumen elaborado por un humano, observamos ciertas diferencias. Cabe destacar que también esta fue una tarea de la vida real, que se dejó a un miembro del equipo en la materia de Administración de Redes.

A continuación, se mostrarán. También los textos se encuentran en la carpeta del Proyecto con el nombre “Resumen-Proceso administrativo-Prueba”, ahí está el texto original, el resumen automático y el resumen elaborado por el humano.



Aquí en esta imagen que se encuentra en la parte de arriba se puede observar algunas de las diferencias y similitudes.

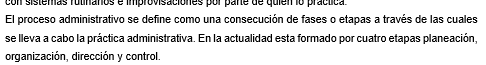
Alguna de la similitud que se encontró, es que algunos de los conceptos mencionados en la lectura tienen coherencia y se parece mucho a lo que el humano también realizo.

**Resumen automático**





**Resumen elaborado por un humano**





De las diferencias que notamos es que no toma en cuenta los subtítulos o palabras remarcadas.

### Descripción de observaciones

Lo que pudimos observar en la realización de los resúmenes automáticos es que se tienen que tener en cuenta ciertas variables. Ya que para resumir desde una página Web, se tienen más consideraciones, por ejemplo: la forma de programación y el lenguaje que se está utilizando

### Trabajo futuro

# **V. Referencias**

# <https://redhistoria.com/historia-y-evolucion-del-resumen-de-textos/>

<https://www.instintoprogramador.com.mx/2019/07/resumen-de-texto-con-nltk-en-python.html?m=1>

<https://code.tutsplus.com/es/tutorials/introducing-the-natural-language-toolkit-nltk--cms-28620>

<http://www.nltk.org/book/>

<https://likegeeks.com/es/tutorial-de-pyqt5/>

<https://es.qwe.wiki/wiki/Automatic_summarization#Applications_and_systems_for_summarization>

Herramientas y técnicas. Consultado el 25 de mayo en:

<http://www.corpus.unam.mx/cursocorpus/4_1_Conteo_de_palabras.html>

1. Careaga, J.2012. *Sistema Resumidor-Traductor Automático. México, CDMX. UNAM. Consultado 27/03/2020 en:* [*http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/2640/Tesis.pdf?sequence=1*](http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/2640/Tesis.pdf?sequence=1) [↑](#footnote-ref-1)
2. Mateo, P. González, J.Villena, J. Martínez, J. *Un sistema para resumen automático de textos en castellano. Centro de Empresas la Arboleda. Madrid. Consultado el 20/03/2020 en:* [*http://rua.ua.es/dspace/handle/10045/1486*](http://rua.ua.es/dspace/handle/10045/1486) [↑](#footnote-ref-2)