**Proyecto final**

**Teorías de Autómatas y Lenguajes Formales**

**“Algoritmo de Porter”**

**Integrantes:**

Jorge Tobar Quintanilla

Felipe Troncoso Ponce

Catalina Bustos Zurita

**Profesor:**

Ricardo Corbinaud Pérez

**Ayudante:**

Cristóbal Saldias Rojas

***Índice:***

Introducción…………………………………………………………………………….3

Algoritmo de Porter:

Definición……………………………………………………………………….4

Explicación……………………………………………………………………...5

Aplicaciones…………………………………………………………………….7

Text Mining……………………………………………………………………………..8

Aplicaciones del Text Mining………………………………………………….11

*Data mining…………………………………………………………………………….13*

*La aplicación de los modelos de minería de datos………………………………..15*

*Web mining……………………………………………………………………………..16*

*Explicación del código…………………………………………………………………………17*

*Conclusión……………………………………………………………………………………….22*

*Referencias………………………………………………………………………………………23*

*Anexo:*

*Avance 1 y 2………………………………………………………………………….24*

***Introducción:***

El problema que se planteó fue el poder crear una manera eficaz y automática de poder resumir y sintetizar el tema principal de un texto cualquiera.

Para ello utilizaremos el llamado Algoritmo de Porter y su método de sintetizar textos en este Proyecto.

El Objetivo Principal de este Informe es determinar la eficacia del algoritmo de Porter para lo Solicitado en este curso.

***Algoritmo de Porter:***

***Definición:***

***“****Martin F. Porter es el inventor del Algoritmo de Porter, donde fue escrito originalmente en 1979 como parte de un proyecto más grande llamado “ IR project ”, (...), apareciendo en el Capítulo 6 de este reporte...”* [1].

Posteriormente en 1980, publicó un Paper llamado “ *“An algorithm for suffix stripping”, (…), el cual ha sido citado 10455 veces.”* [2].

Este Algoritmo, está diseñado para eliminar los sufijos y prefijos más comunes morfológicos e inflexionables de las palabras en inglés, siendo el código más pequeño y simple, posteriormente en 1999 ya era ampliamente utilizado y adaptado a otros idiomas, utilizando un proceso de normalización de términos usado en la Clasificación Automática o en sistemas de Búsqueda y Recuperación de Información (ISR), ésta última consiste en facilitar la localización de datos y la relación que estos tienen dentro del contexto.

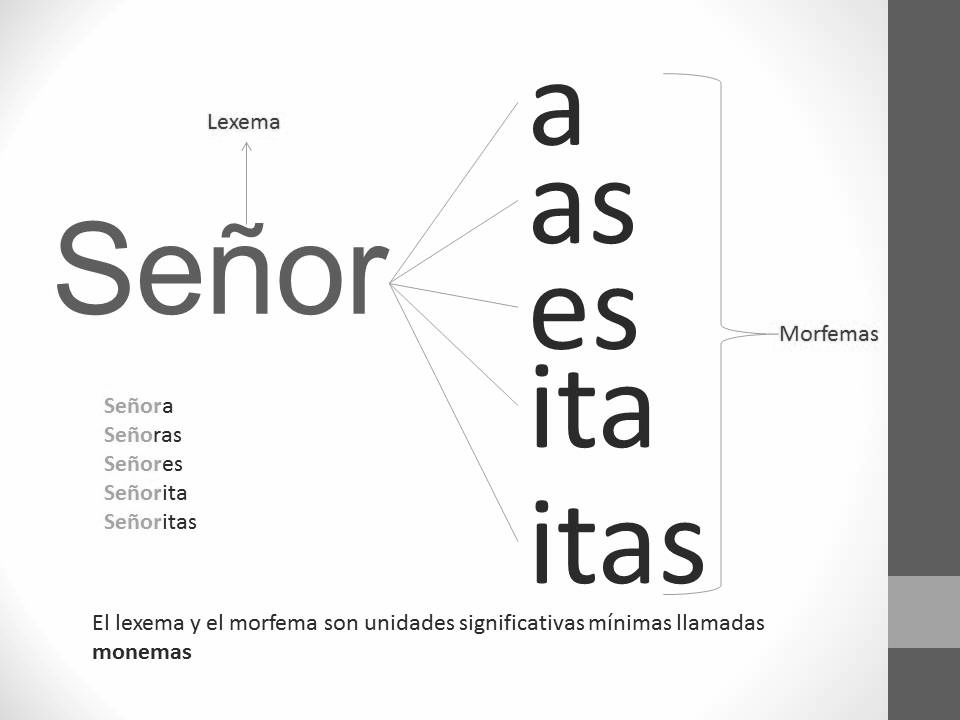
Siendo esto de gran utilidad en la tecnología aplicada a Buscadores, Traductores Automáticos y Extracción de información. Sistemas utilizados ampliamente y siendo casi indispensables en la vida moderna, tal y como veremos más adelante.

El Algoritmo de Porter utiliza un método llamado Stemming que consiste en reducir una palabra a su raíz, en inglés llamado “steam”, por ello se le llama comúnmente “Porter’s Stemmer Algorithm”.

***Explicación:***

“*En lenguaje computacional, un stem es la parte de la palabra que nunca cambia al ser flexionada morfológicamente…*” [3]

Para entender mejor el Stemming tenemos que tener en cuenta el proceso lingüístico conocido como: Lematización, la cual se lleva a cabo cuando de una forma flexionada (sea éste el plural, el género o la conjugación de una palabra, entre otros...) se logre encontrar el lexema, forma que se acepta como representante, o raíz, de todas las formas morfológicas de la palabra en cuestión, siendo este equivalente al stem.



El Stem o Lexema informático sería similar, no necesariamente equivalente al Lexema en Gramática. Siendo este, en resumen, significado común a toda una familia de palabras, como muestra el esquema.

En el Algoritmo de Porter utilizado en este Proyecto, además de implementar los stem de las palabras presentes en el texto lo normalizamos convirtiendo éste a minúscula, y aplicaremos autómatas, utilizando los respectivos regex en Java para la eliminación de caracteres especiales tales como signos de puntuación y dígitos, con esta misma metodología

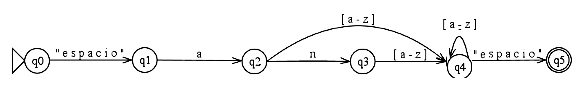
Para aplicar las Expresiones Regulares, o Regex en inglés, primero debemos tener en cuenta que parte de la palabra o qué carácter queremos “seleccionar”, luego debemos hacer un esquema de cómo se compone la palabra, en caso de caracteres como dígitos o signos simplemente se anotan por extensión, por ejemplo:





******

Después de haber hecho el esquema, podemos pasarlo a un autómata, según el ejemplo:

******

Por último para aplicar el Regex en Java, utilizaremos la librería java.util.Regex.Pattern y las construcciones que tiene esta [5], por ejemplo:



***Aplicaciones:***

Una de las principales aplicaciones, tal y como detallaremos más adelante, es **Text Mining** en inglés.

“*Las técnicas de minería de datos se están utilizando desde hace varios años para la obtención de patrones en los datos y para la extracción de información valiosa en el campo de la Ingeniería del Software..*.” [6]

Un ejemplo de este recurso, es el que veremos en este proyecto, crear resúmenes de textos extensos. Esto se puede ocupar, por ejemplo, en periodismo, cuando se debe comunicar gran cantidad de información, será mejor resumir el contexto del texto, allí se aplica el Algoritmo de Porter.

**Buscadores Comerciales**, entre ellos: especializados, motores de búsqueda, meta buscador. Son aplicaciones que utilizamos a diario también funcionan a base de stemming, un ejemplo de esto son los últimos buscadores de Google.

***Text mining***:

La minería de textos permite el descubrimiento de patrones interesantes y nuevos conocimientos en un conjunto de textos, es decir, su objetivo consiste en descubrir tendencias, desviaciones y asociaciones entre una gran cantidad de información textual. Esto nos permite encontrar conocimiento significativo a partir de datos textuales sin estructurar. La minería de textos extrae información nueva por lo que es algo totalmente distinto a la búsqueda web, en la cual se busca información ya conocida, no se extrae información nueva.

¿QUÉ CLASE DE GENTE HACE MINERÍA DE TEXTOS?

La tecnología de la minería de textos es actualmente ampliamente aplicada por una extensa variedad de usuarios, desde organizaciones gubernamentales, instituciones de investigación y empresas para sus necesidades diarias. Estos son algunos ejemplos de uso en diferentes campos:

*Investigación*: por ejemplo, el descubrimiento de conocimientos, la atención médica y sanitaria: en el pasado, a un investigador humano le lleva mucho tiempo analizar y obtener información relevante. En algunos casos, esta información ni siquiera era accesible. La minería de textos permite a los investigadores encontrar más información y de forma más rápida y eficiente.

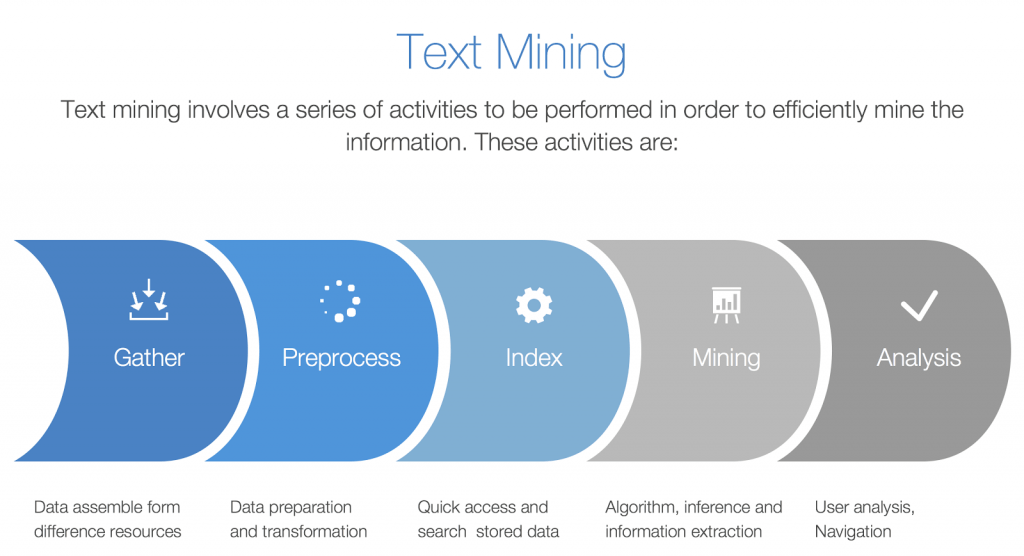
*Negocios*: por ejemplo, las grandes empresas utilizan la minería de textos para ayudar en la toma de decisiones y responder rápidamente a las consultas de los clientes en procesos tales como la gestión de riesgos o el filtrado de currículos

*Seguridad*: En anti-terrorismo, el análisis de los blogs y otras fuentes de texto en línea se utiliza para prevenir delitos en Internet y luchar contra el fraude.

*Diariamente*, La minería de texto es usada por los sitios web de correo electrónico para crear métodos de filtrado más confiables y efectivos, para el filtrado de spam, análisis de datos de medios sociales, etc. También para identificar las relaciones entre los usuarios y ciertos productos o para determinar las opiniones de los usuarios sobre temas particulares

¿Cómo Funciona, Esta Minería De Texto Y Datos?

La extracción de textos puede dividirse en cinco pasos:



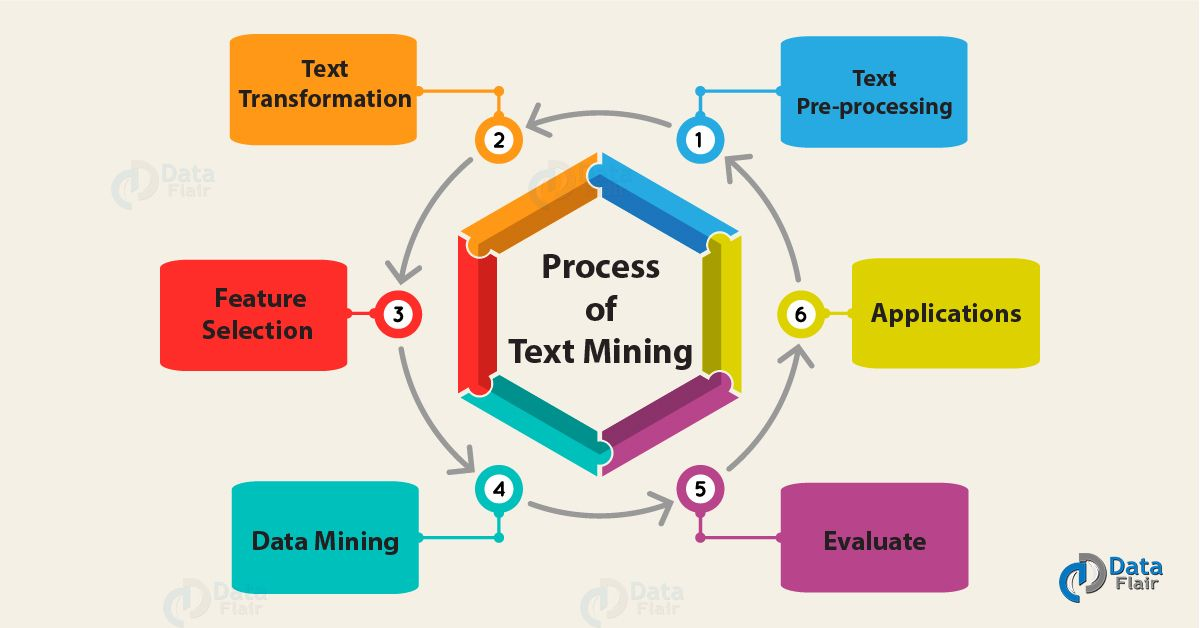
1. *Recolección*: Recopilación de datos de diferentes recursos, tales como sitio web, correos electrónicos, comentarios de clientes, archivo de documentos. Dependiendo de la aplicación, este proceso puede ser completamente automatizado o guiado por una persona encargada de realizar este proceso.

*2. Pre procesamiento:* La identificación del contenido y la extracción de características representativas

*3. Limpieza de textos*: eliminación de cualquier información innecesaria o no deseada, como los anuncios de las páginas.

4. *Tokenización*: un ordenador sólo “ve” una cadena de caracteres, sin poder identificar, por ejemplo, párrafos, frases o palabras. La *Tokenización* divide el texto en entidades significativas (palabras, oraciones, etc.) dados los espacios en blanco presentes y las puntuaciones.

5. *Extracción de características* (también llamada selección de atributos): es el proceso de caracterización.



### ***Las aplicaciones del Text Mining***

La minería de textos puede ser encontrada en campos diferentes, de hecho, podemos encontrar multitud de trabajos de ámbitos diversos como sería el caso de la medicina, el análisis de opiniones, la filología, la biología, etcétera. No obstante, para que estos textos están publicados y han tenido que pasar por un análisis de la minería de datos. Las cinco principales aplicaciones del Text Mining que podemos encontrar son las siguientes:

1. **La Búsqueda de Información**. Son las búsquedas que se hacen a partir de una pregunta, palabra concreta o contenido interesante. Por ejemplo, podemos encontrar esta aplicación en Google y su motor de búsqueda del contenido que podemos encontrar en Internet.
2. **El reconocimiento de entidades mencionadas y referencias**. Se utilizan estadísticas, nombres de lugares, personas, entidades, etcétera. Con este análisis se pretende agrupar una cantidad de textos que tengan la misma palabra o información, como sería el caso de los Trending Topic (Tema de Tendencia) de Twitter.
3. **Clustering**. Se agrupan los textos que reúnen criterios similares que no pueden ser vistos directamente. Sería el caso de reunir los libros por estilo de redacción o vocabulario utilizado.
4. **Clasificación**. Se trata de la etiquetar los textos para dividirlo en categorías de forma más eficaz. Por poner un ejemplo podemos ver los documentos de texto de Facebook o los vídeos de YouTube que están agrupados por ofensivos o no ofensivos, contenido violento, etcétera.
5. **Análisis de sentimientos**. En base a los resultados estadísticos podremos saber si un determinado texto es positivo, negativo o incluso neutral. En este caso podríamos hablar de Tripadvisor, una app donde podemos acceder a los comentarios sobre restaurantes, hoteles etcétera. A través del análisis de texto podemos ver cuántos comentarios son positivos y cuántos negativos.

***Data mining (Minería de datos):***

La **minería de datos** o **exploración de datos** (es la etapa de análisis de "Knowledge Discovery in Databases" o KDD) es un campo de la estadística y las ciencias de la computación referido al proceso que intenta descubrir patrones en grandes volúmenes de conjuntos de datos. Utiliza los métodos de la inteligencia artificial, aprendizaje automático, estadística y sistemas de bases de datos. El objetivo general del proceso de minería de datos consiste en extraer información de un conjunto de datos y transformarla en una estructura comprensible para su uso posterior. Además de la etapa de análisis en bruto, supone aspectos de gestión de datos y de bases de datos, de procesamiento de datos, del modelo y de las consideraciones de inferencia, de métricas de Intereses, de consideraciones de la teoría de la complejidad computacional, de post-procesamiento de las estructuras descubiertas, de la visualización y de la actualización en línea.

Ahora bien, hablando acerca de la estructura de la minería de datos, esta consta básicamente en el uso de un algoritmo o algún programa computacional para realizar las actividades de búsqueda dentro de las grandes cantidades de información contenidas en la base de datos.

El uso de estos programas y algoritmos es con el fin de poder detectar tendencias y patrones que se encuentran de alguna manera ocultos en los datos históricos de las organizaciones.

Estos programas son los que anteriormente llamamos mineros, estos mineros, programas o algoritmos, son creados por los usuarios en los cuales se utilizan varias técnicas de exploración de datos, las técnicas que se pueden utilizar son:

* Cluster
* Asociaciones
* Clasificaciones
* Visualizaciones
* Redes neuronales
* Algoritmos genéricos
* Detección de desviaciones

Todos estos métodos antes mencionados requieren de una base de datos muy grande para que así puedan tener una mayor eficiencia.

La tarea de la minería de datos es **el análisis automático o semi-automático de grandes cantidades de datos**. Esto sirve para extraer patrones interesantes hasta ahora desconocidos. Hablamos así de los grupos de registros de datos (análisis clúster), registros poco usuales (la detección de anomalías) y dependencias (minería por reglas de asociación). Esto generalmente implica el uso de técnicas de bases de datos como los índices espaciales. Así, estos patrones pueden ser vistos como una especie de resumen de los datos de entrada. Además de poder ser utilizados en el análisis adicional o, por ejemplo, en el aprendizaje automático y análisis predictivo.

Uno de los ejemplos que podemos poner es la minería de datos. Esta podría identificar varios grupos en los datos, que luego pueden ser utilizados para obtener resultados más precisos. Pudiendo **predecir los problemas a través de un sistema de soporte de decisiones**. Ni la recolección de datos, preparación de datos, ni la interpretación de los resultados y la información son parte de la etapa de minería de datos. No obstante, pertenecen a todo el proceso KDD como pasos adicionales.

### ***La aplicación de los modelos de minería de datos***

Los patrones y tendencias de la minería de datos se pueden considerar también modelos de Data Mining. Cabe destacar que hay cinco escenarios específicos donde dichos modelos pueden ser aplicados:

* **Previsiones**. Cálculo de ventas y predicción de cargas en los servidores, o incluso cálculo del tiempo que el servidor permanece inactivo.
* **Riesgos y probabilidades**. Se trata de un escenario donde se busca distinguir a los mejores clientes para distribuir el correo. Pudiendo así determinar un equilibrio en cuanto al riesgo y a las probabilidades de resultados.
* **Búsqueda de secuencias**. Se efectúa el análisis de los artículos que los propios clientes introducen en su carro de la compra. A través de las secuencias y datos extraídos se pueden llegar a predecir posibles eventos futuros.
* **Agrupamiento**. En este escenario se busca dividir a los clientes o los eventos en grupos de elementos que están relacionados entre sí. De este modo se puede realizar un análisis y predecir las afinidades.

***Web mining (Minería web):***

Web mining - es la aplicación de técnicas de minería de datos para descubrir los patrones de la Web. De acuerdo a los objetivos de análisis, la minería web se puede dividir en tres tipos diferentes, que son la minería de uso de la Web, minería del contenido de la Web y minería de la estructura de la Web.

En este sentido podemos definir el *Web mining* en tres variantes:

1. Minería del contenido de la Web, o *Web Content Mining*;
2. Minería de la estructura de la Web, o *Web Structure Mining*;
3. Minería de los registro de navegación en la Web, o *Web Usage Mining*.

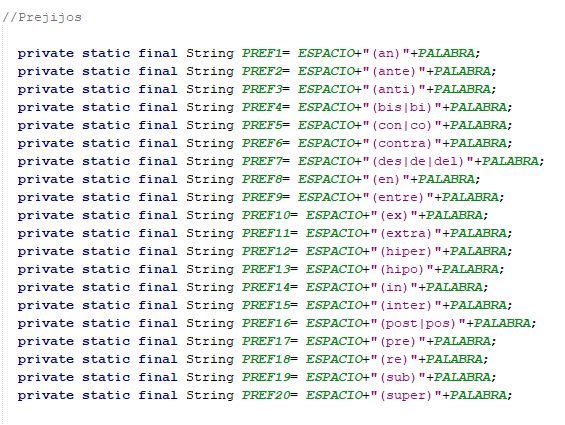
***Explicación del código:***

Regex:

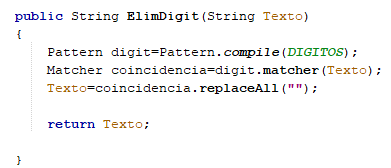
Se abre la librería para implementar los regex:



Se declaran las constantes correspondientes a los respectivos prefijos, tal como se muestra en el ejemplo:

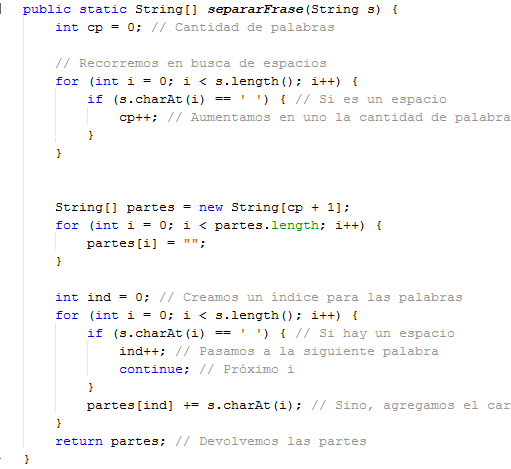


Se repite esta declarativa para los sufijos, adjetivos, artículos y pronombres. Luego se aplican las funciones para eliminar estos mismos.



Donde se crea una variable Pattern y se le da el patrón a buscar, el matcher se encarga de encontrar donde se ubica el patrón en el texto y finalmente se reemplaza este con un vacío. Finalmente se retorna el texto.

separarFrase:



Es una clase que permite convertir un texto de tipo String en una cadena de String, donde cada posición del array representa una palabra. Su iniciación en el main:

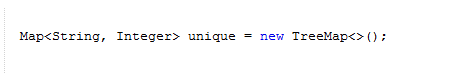
Método para ver palabras repetidas y agregar el diccionario:

El método que usamos fue, del array de Strings de la función anterior, ingresarlos a un mapa de árboles (TreeMap). Que permite automáticamente convertir un texto de tipo String array en un árbol que automáticamente cuenta la cantidad de veces que se repite una palabra.

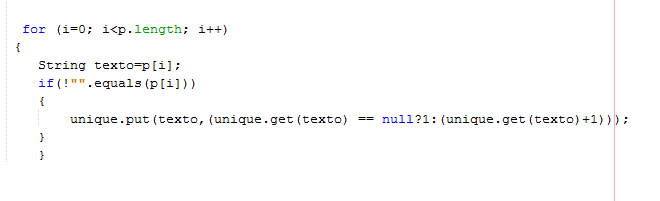
Librerías utilizadas:



Para iniciarlo en el main:

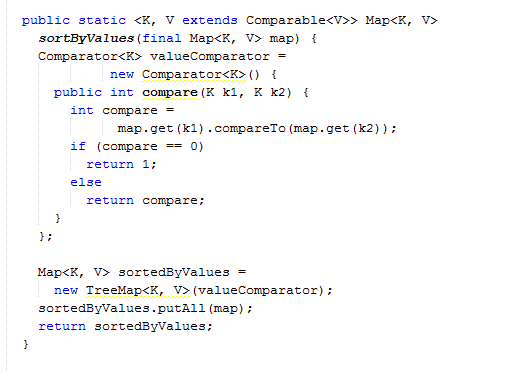


Y para convertir el texto al objeto:

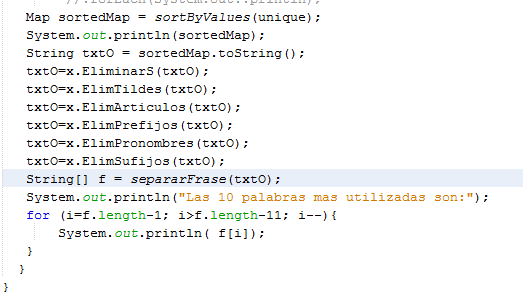


sortByValues:

Al objeto TreeMap,se le crea una clase función para hacer Sort y ordenar de menor a mayor las palabras repetidas en el texto.



Al tenerlo ordenado, hay que convertir el objeto Map a un String con la función .toString del objeto. El string resultante se transforma en un array de Strings con la funcion **separarFrase**, y de ahí mostrar los diez últimos elementos de ésta.

La parte final del main queda:

Con el último for para mostrar las 10 últimas palabras de la cadena, que son las 10 más frecuentes.

***Conclusión:***

El Algoritmo de Porter es una herramienta que puede ser usada para distintos fines, una de los más importantes, la minería de textos.

El uso de REGEX para pasar nuestro idioma a lenguaje de máquina es fundamental para concretar los autómatas ideados para el algoritmo, los cuales simplifican en gran medida la implementación del Algoritmo.

Este algoritmo tiene como “defecto” que fue creado en idioma inglés, idioma en el cual es evidente la menor cantidad de pronombres y adjetivos con respecto a nuestro idioma. Lo que implica que el español tendrá algunos errores puesto que hay más conjugaciones verbales y los Lexemas no son tan distinguibles como en el inglés.

Por lo tanto, a pesar de lo anterior, el resultado entregado por el Algoritmo de Porter utilizado en este proyecto es muy eficaz para los fines propuestos al inicio de este informe ya que nos permitió realizar un resumen adecuado para tener un contexto de que se trata el texto pedido.

***Referencias:***

[1] Martin F. Porter. “The Porter Stemming Algorithm” s.f. Recuperado el 3 de Julio de 2019 en: <https://tartarus.org/martin/PorterStemmer/>

[2] Artículos en Google Scholar. Recuperado el 3 de Julio de 2019 en: <https://scholar.google.com/scholar?hl=es&as_sdt=0%2C5&q=An+algorithm+for+suffix+stripping&btnG=>

[3] Anjali Ganesh Jivani. “A Comparative Study of Stemming Algorithms” s.f. Recuperado el 3 de Julio de 2019 en: <https://pdfs.semanticscholar.org/1c0c/0fa35d4ff8a2f925eb955e48d655494bd167.pdf>

[4] Gómez Díaz, Raquel. “CRÍTICA DE LIBROS: LA LEMATIZACIÓN EN ESPAÑOL: UNA APLICACIÓN PARA LA RECUPERACIÓN DE INFORMACIÓN”. Rev. Esp. Doc. Cient. 29. 1. 2006. en:<http://redc.revistas.csic.es/index.php/redc/article/download/301/348>

[5] Oracle. “Class Pattern”. s.f. Recuperado el 3 de Julio de 2019 en: <https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>

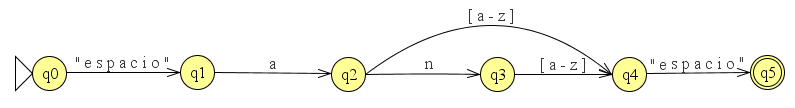
[6] María N. Moreno García. “ APLICACIÓN DE TÉCNICAS DE MINERÍA DE DATOS EN LA CONSTRUCCIÓN Y VALIDACIÓN DE MODELOS PREDICTIVOS Y ASOCIATIVOS A PARTIR DE ESPECIFICACIONES DE REQUISITOS DE SOFTWARE” s.f. Recuperado el 3 de Julio de 2019 en: <http://ceur-ws.org/Vol-84/paper4.pdf>

***Anexo: Avance 1 y 2.***

***Prefijos:***

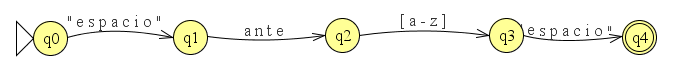
**-a, an:**

**\b((a|n)+(a-z)\*)**

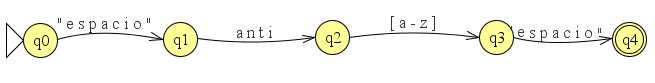
****

**-ante:**

**\b((ante)+(a-z)\*)**

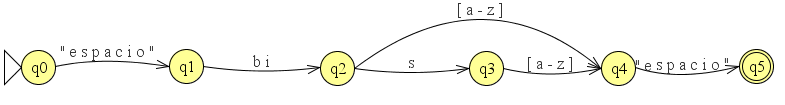
****

**-anti:**

**\b((anti)+(a-z)\*)**

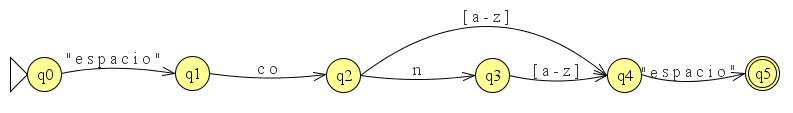
**-bi, bis:**

 **\b((bi|s)+(a-z)\*)**

****

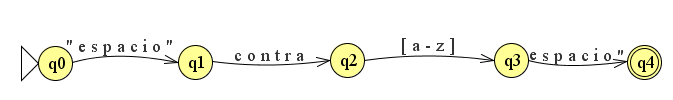
**-co, con:**

**\b((co|n)+(a-z)\*)**

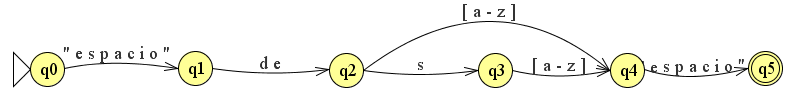
****

**-contra:**

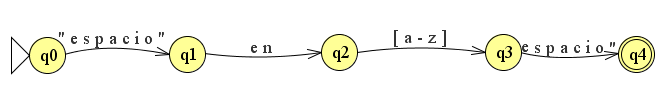
**\b((contra)+(a-z)\*)**

****

**-de, des:**

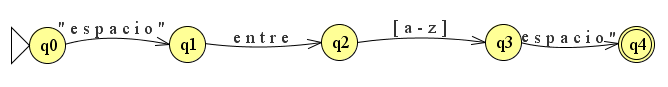
**\b((de|s)+(a-z)\*)**

**-en:**

**\b((en)+(a-z)\*)**

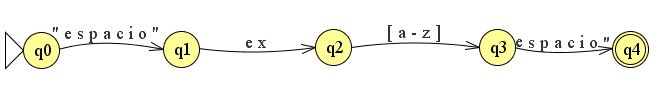
**-entre:**

**\b((entre)+(a-z)\*)**

****

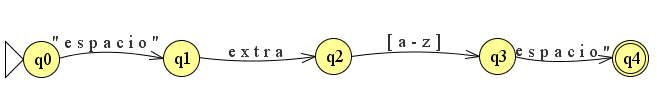
**-Ex:**

**\b((ex)+(a-z)\*)**

****

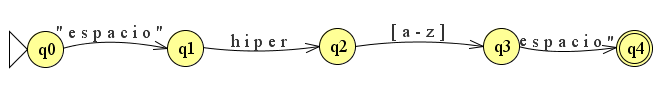
**-extra:**

 **\b((extra)+(a-z)\*)**

****

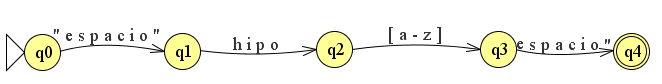
**-hiper:**

 **\b((hiper)+(a-z)\*)**

****

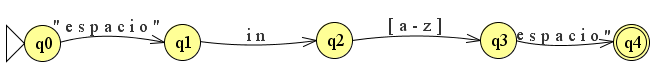
**-hipo:**

**\b((hipo)+(a-z)\*)**

****

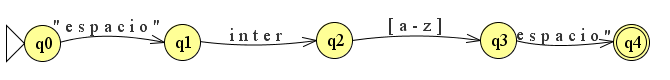
**-in:**

 **\b((in)+(a-z)\*)**

****

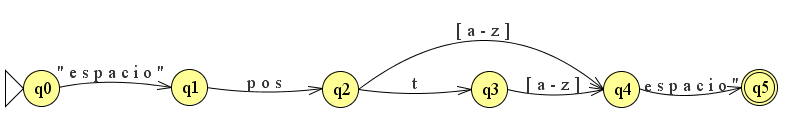
**-inter:**

**\b((inter)+(a-z)\*)**

****

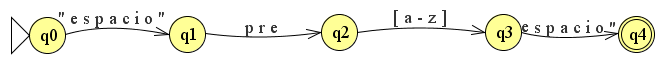
**-pos(t):**

**\b((pos|t)+(a-z)\*)**

****

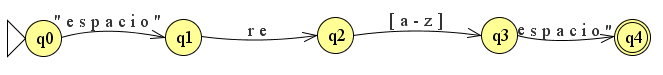
**-pre:**

**\b((pre)+(a-z)\*)**

****

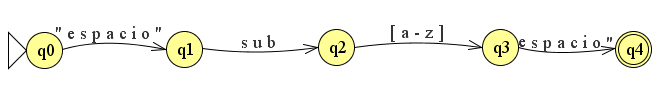
**-re:**

**\b((re)+(a-z)\*)**

****

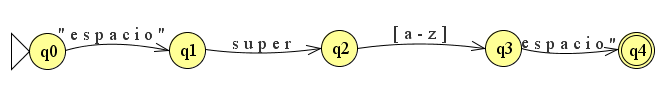
**-sub:**

**\b((sub)+(a-z)\*)**

****

**-Super:**

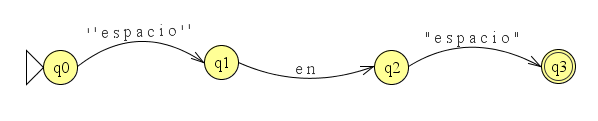
**\b((super)+(a-z)\*)**

****

***Artículos:***

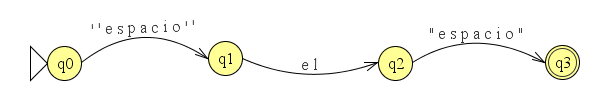
**-en:**

**\b(en) \b**

****

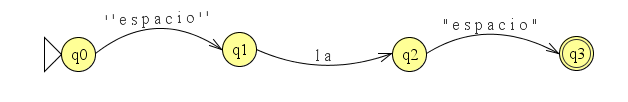
**-el:**

**\b(el) \b**

****

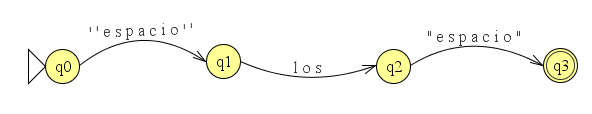
**-la:**

**\b(la) \b**

****

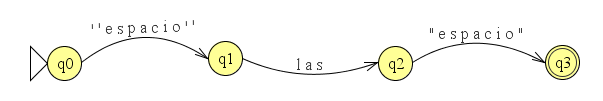
**-los:**

**\b(los) \b**

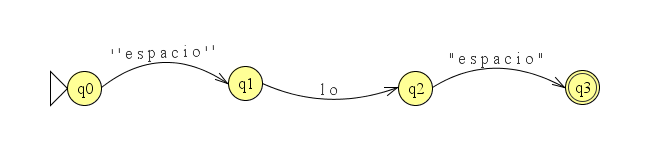
****

**-las:**

**\b(las) \b**

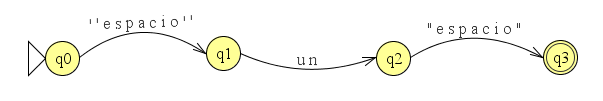
****

**-lo:**

**\b(lo)**

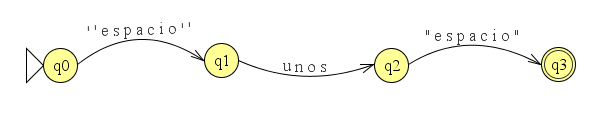
**-un:**

**\b(un) \b**

****

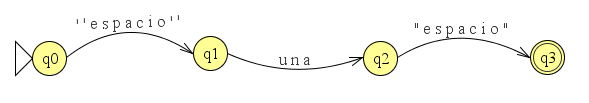
**-unos:**

**\b(unos) \b**

****

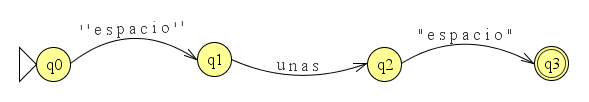
**-una:**

**\b(una) \b**

****

**-unas:**

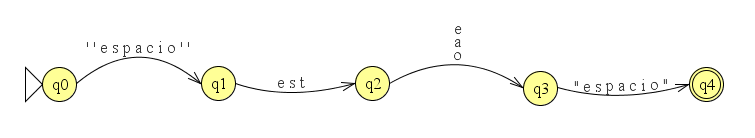
**\b(unas) \b**

****

***Pronombres:***

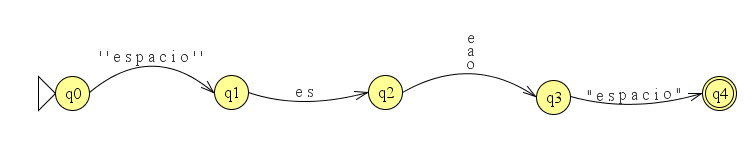
**-esto, este, esta :**

**\b((est+(a|e|o)) \b**

****

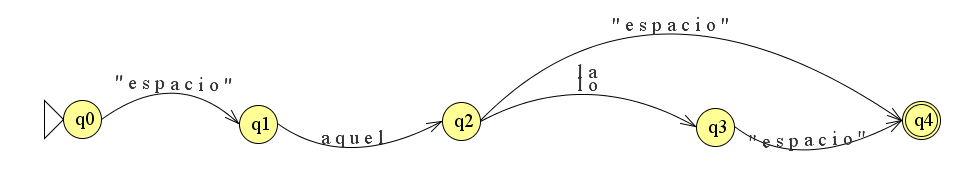
**-ese, eso, esa:**

**\b((es+(a|e|o)) \b**

****

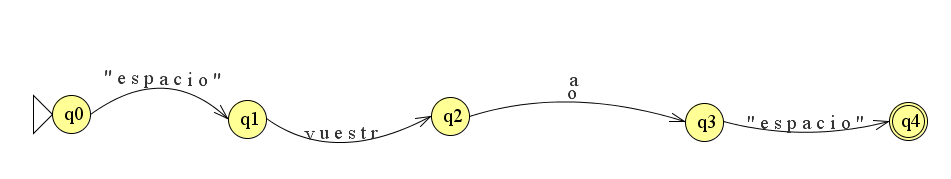
**-aquello, aquella, aquel :**

**\b((aquel+(la|lo)) \b**

****

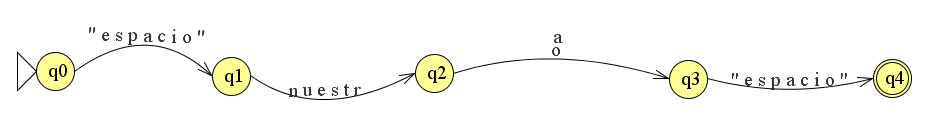
**-vuestro(a):**

**\b((vuestr+(a|o)) \b**

****

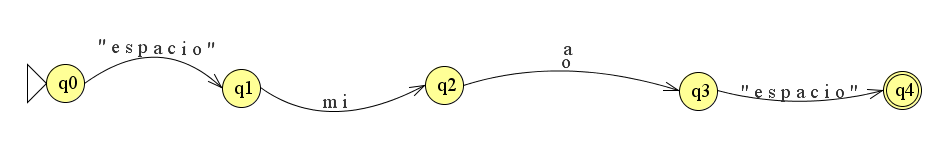
**-nuestro(a):**

**\b((nuestr+(a|o)) \b**

****

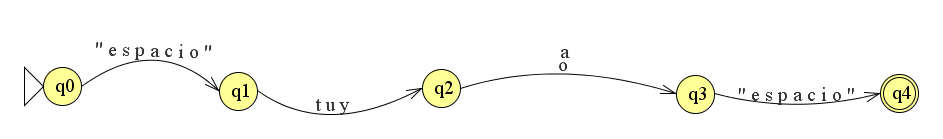
**-mio(a):**

**\b((mi+(a|o)) \b**

****

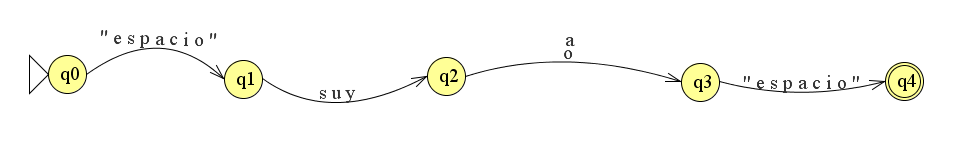
**-tuyo(a):**

**\b((tuy+(a|o)) \b**

****

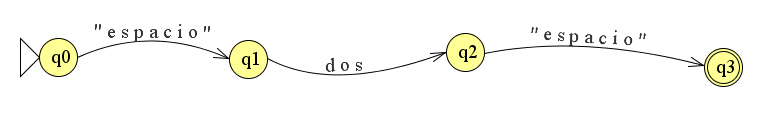
**-suya(o):**

**\b((suy+(a|o)) \b**

****

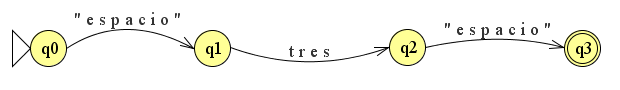
**-Dos:**

**\b(dos)\b**

****

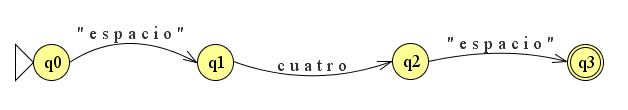
**-Tres:**

**\b(tres)\b**

****

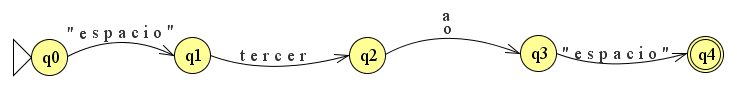
**-Cuatro:**

**\b(cuatro)\b**

****

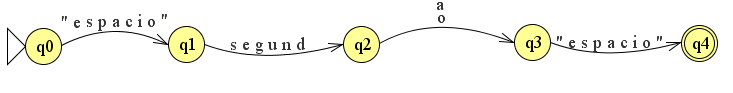
**-Tercero(a):**

**\b((tercer+(a|o)) \b**

****

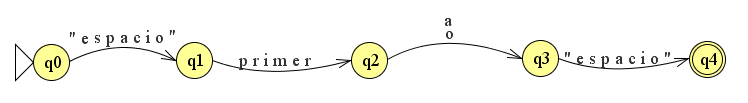
**-Segundo(a):**

**\b((segund+(a|o))\b**

****

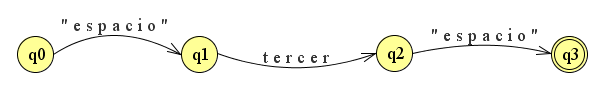
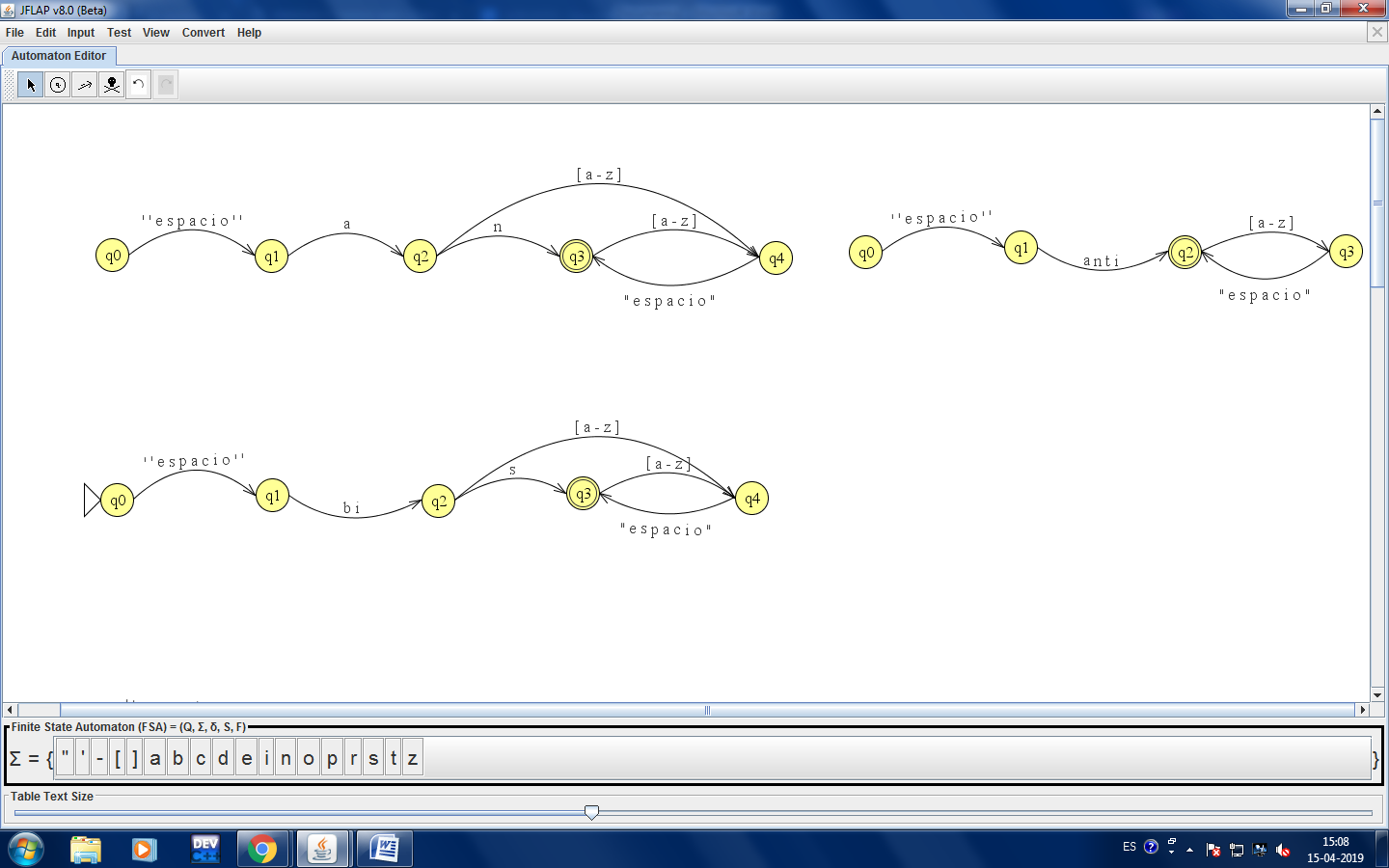
**-primero(a):**

**\b((primer+(a|o))\b**

****

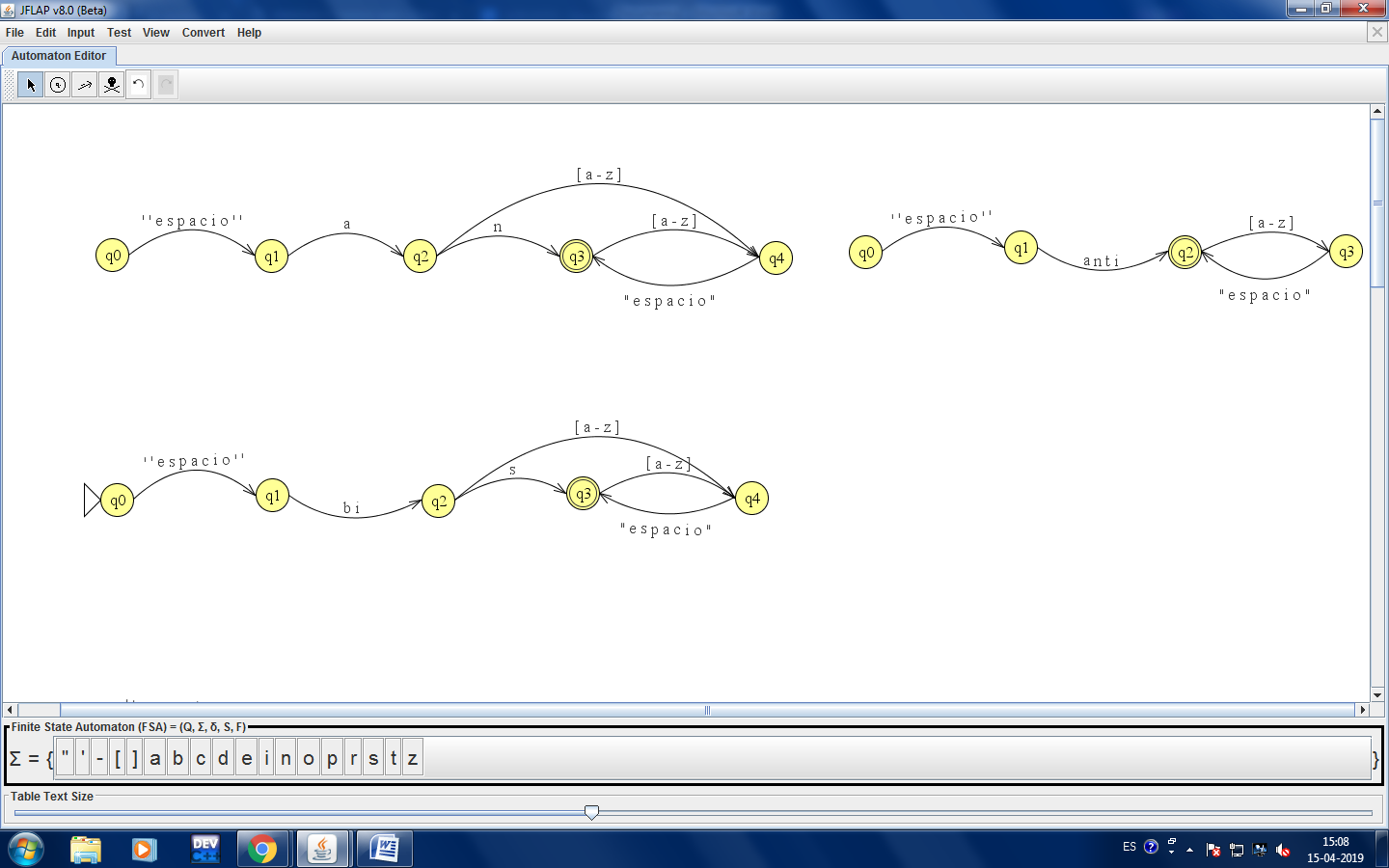
**-tercer:**

**\b(tercer)\b**

****

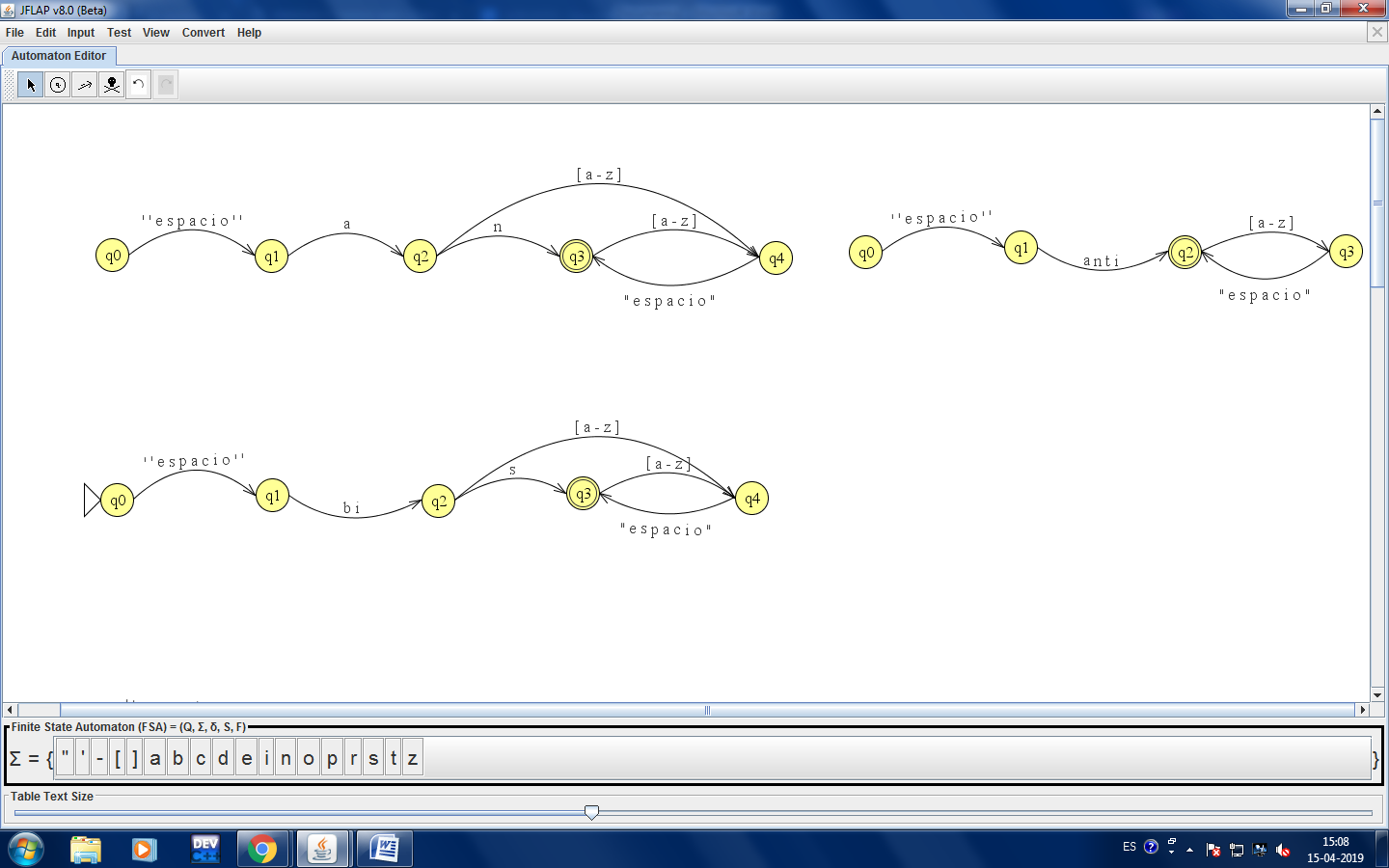
**-primer:**

**\b(primer)\b**

****

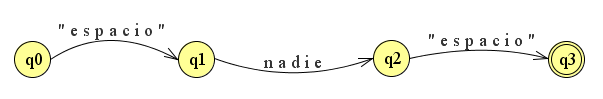
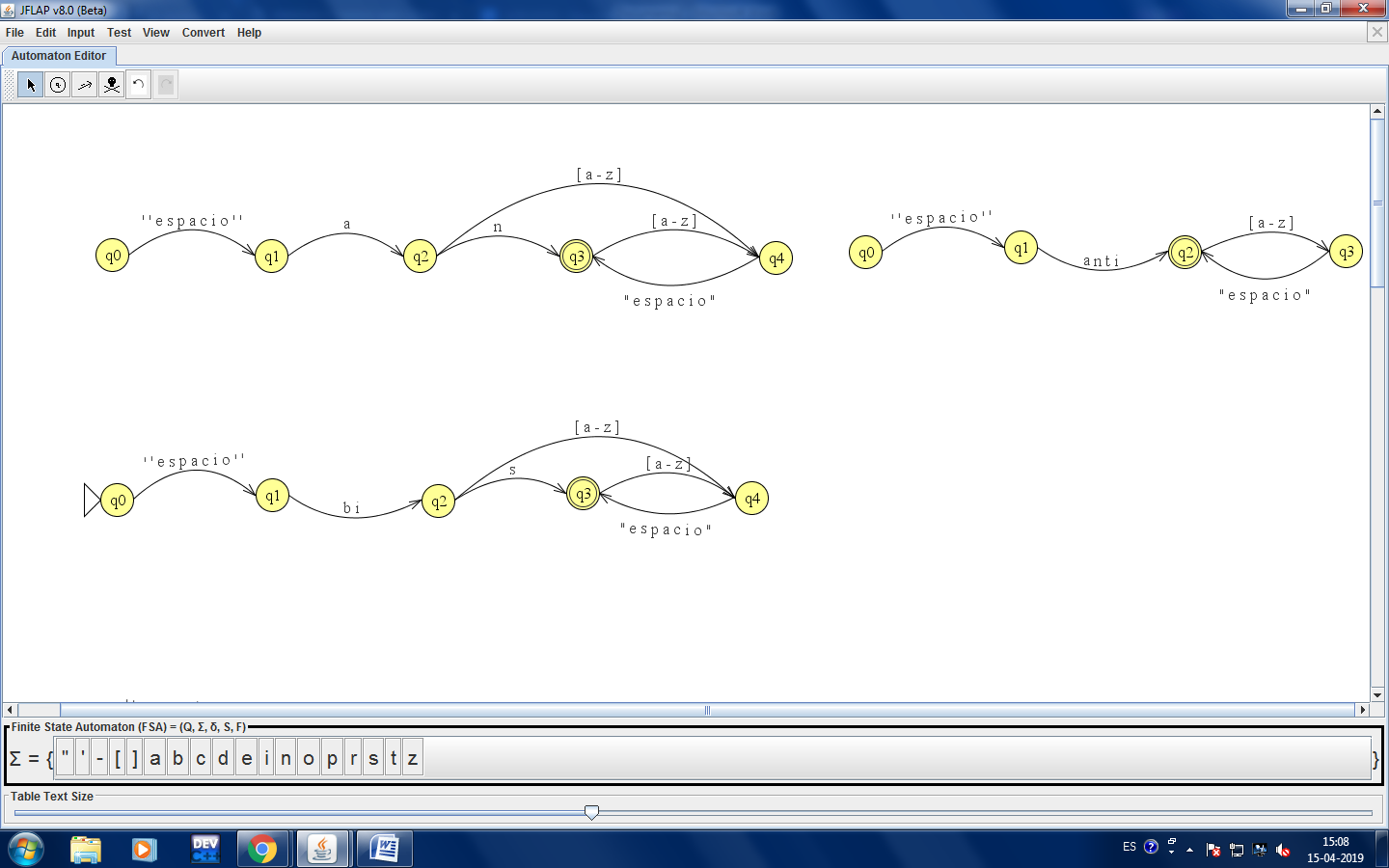
**-alguien:**

**\b(alguien) \b**

****

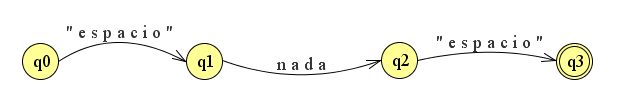
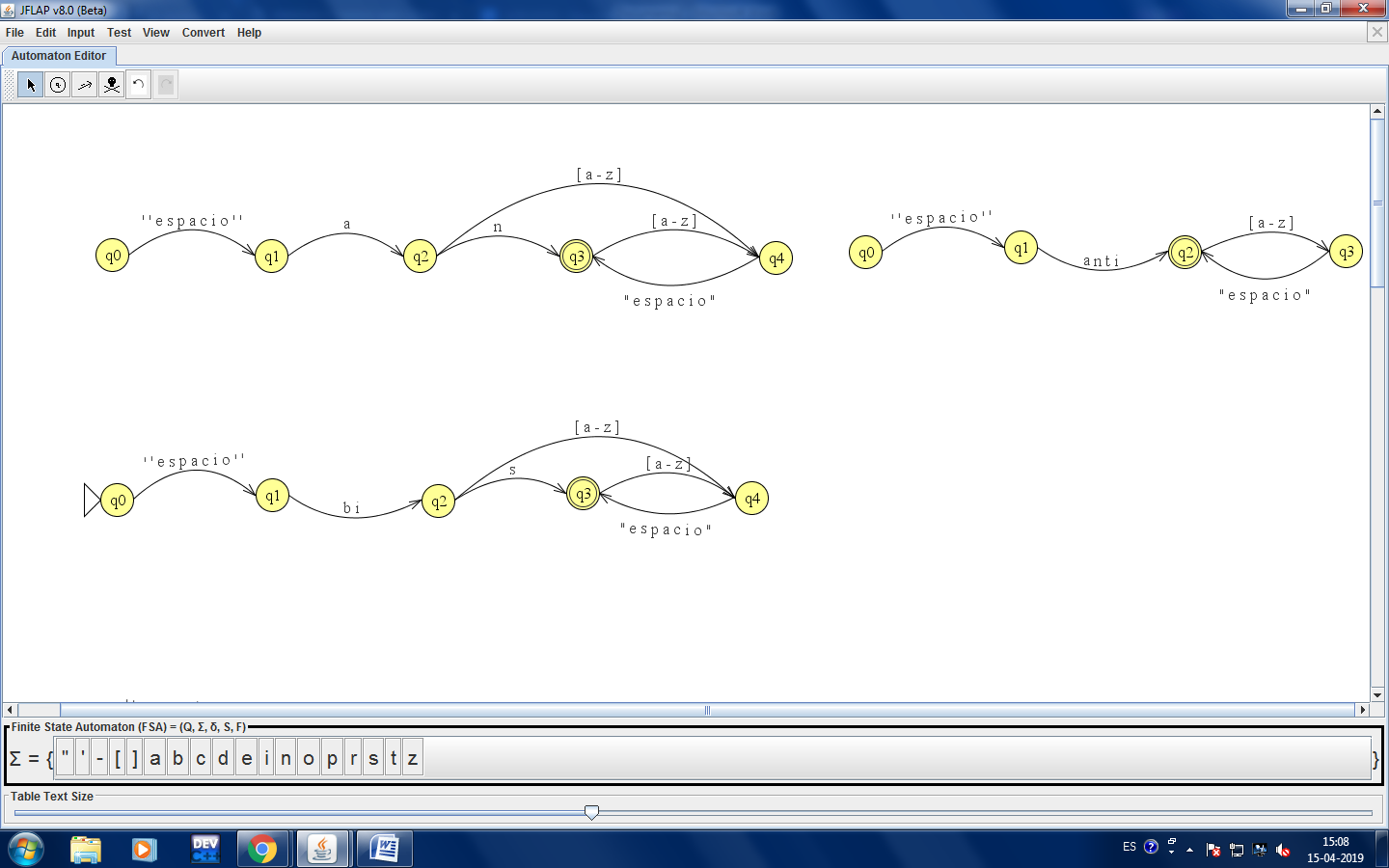
**-nadie:**

**\b(nadie)\b**

****

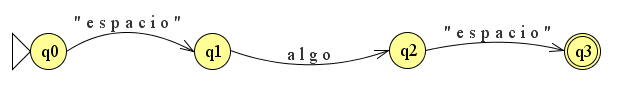
**-nada:**

**\b(nada) \b**

****

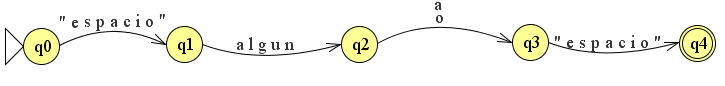
**-algo:**

**\b(algo) \b**

****

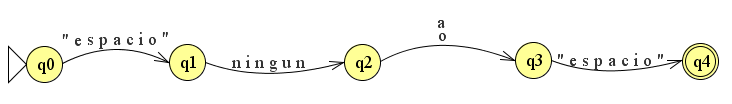
**-alguno(a):**

**\b((algun+(a|o)) \b**

****

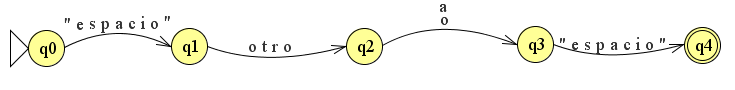
**-ninguno(a):**

**\b((ningun+(a|o)) \b**

****

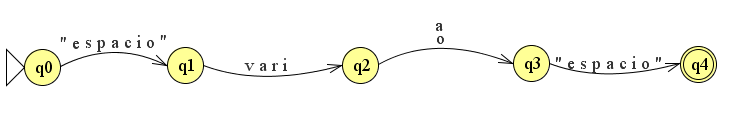
**-otro(a):**

**\b((otr+(a|o)) \b**

****

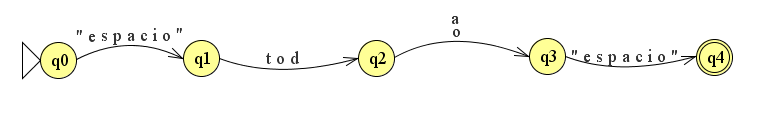
**-vario(a):**

**\b((vari+(a|o))\b**

****

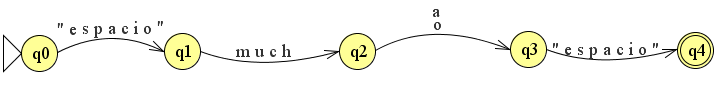
**-todo(a):**

**\b((tod+(a|o))\ b**

****

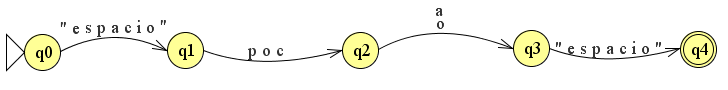
**-mucho(a):**

**\b((much+(a|o)) \b**

****

**-poco(a):**

**\b((poc+(a|o)) \b**

****

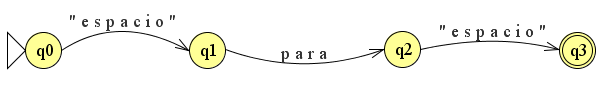
**-bastante:**

**\b(pre)\b**

****

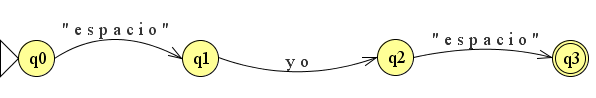
**-para:**

**\b(para)\b**

****

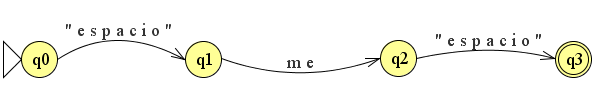
**-yo:**

**\b(yo)\b**

****

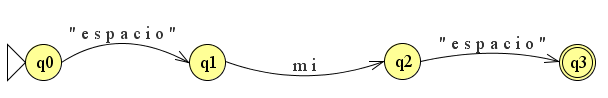
**-me:**

**\b(me)\b**

****

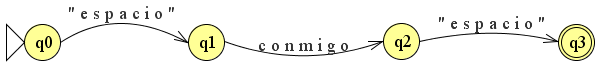
**-mi:**

**\b(mi)\b**

****

**-conmigo:**

**\b(conmigo)\b**

****

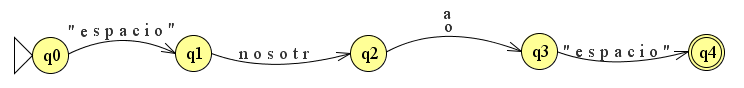
**-consigo:**

**\b(consigo)\b**

****

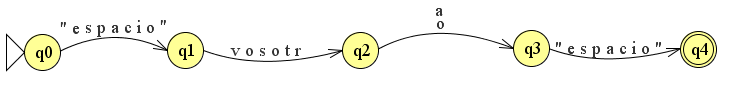
**-nosotro(a):**

**\b((nosotr+(a|o)) \b**

****

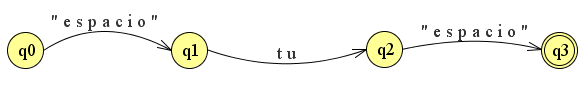
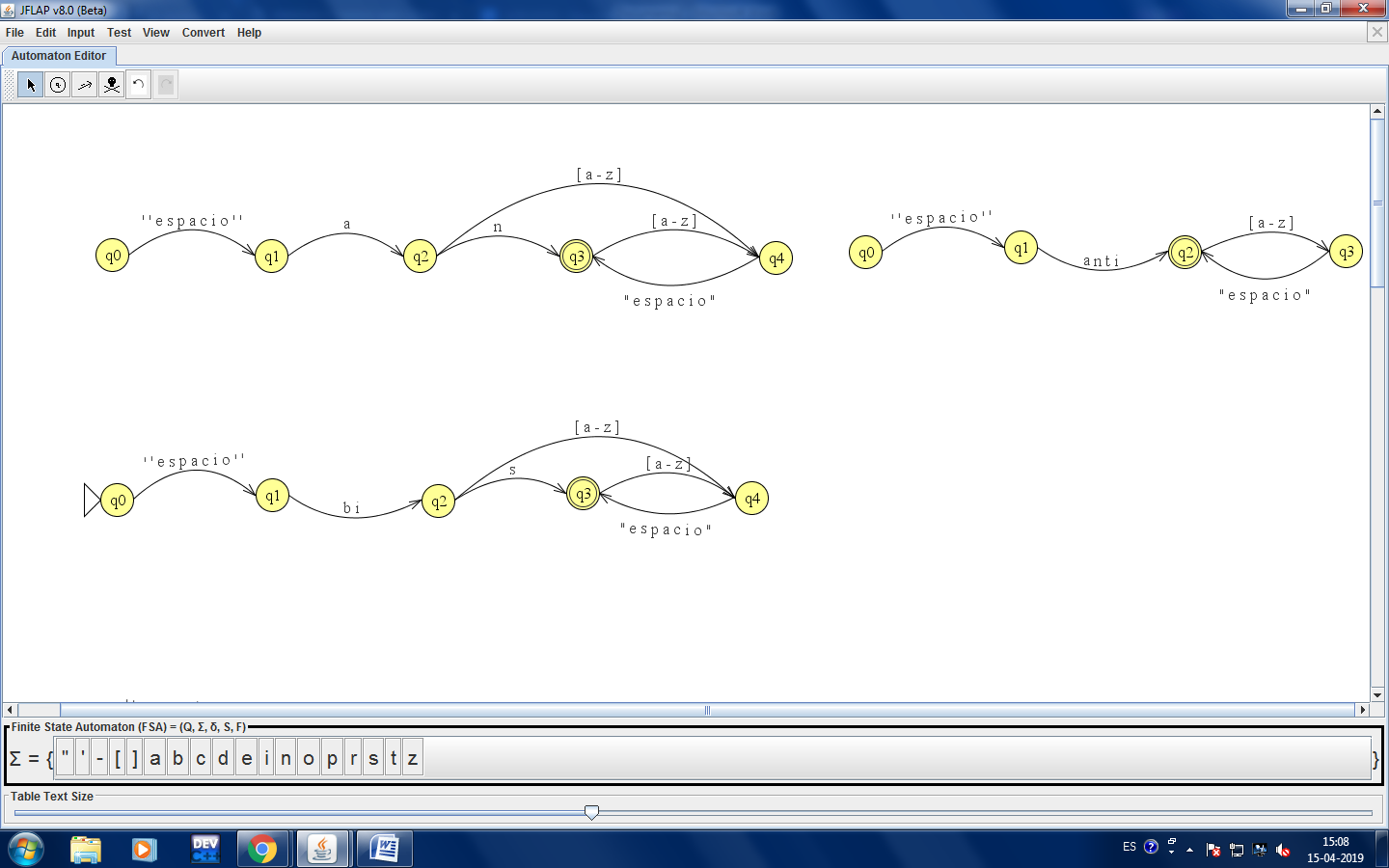
**-vosotro(a):**

**\b((vosotr+(a|o)) \b**

****

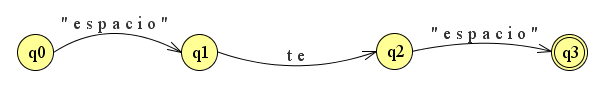
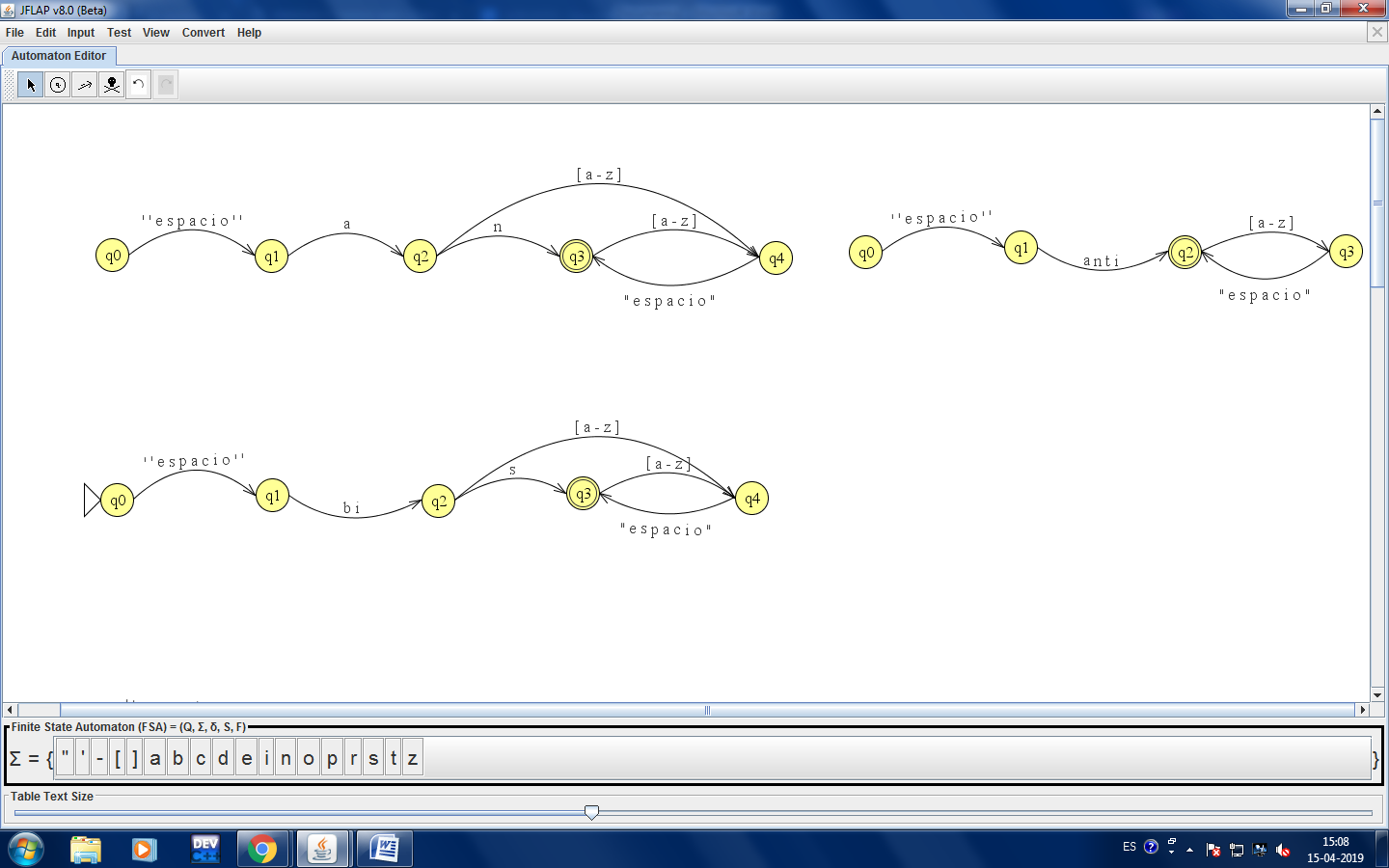
**-tu:**

**\b(tu)\b**

****

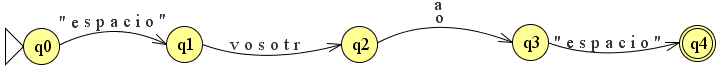
**-te:**

**\b(te)\b**

****

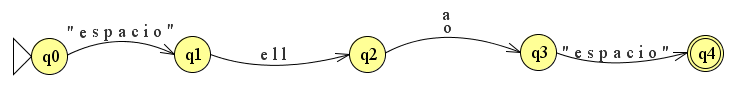
**-ti:**

**\b(ti)\b**

****

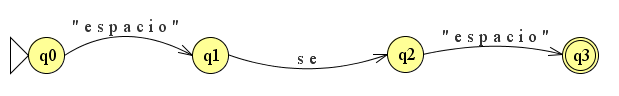
**-ella(o):**

**\b((ell+(a|o)) \b**

****

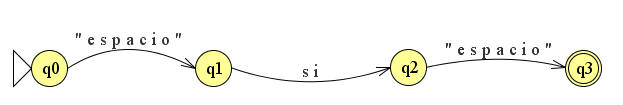
**-se:**

**\b(se)\b**

****

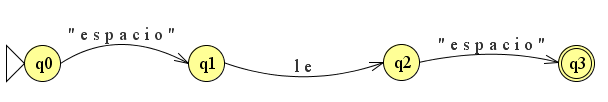
**-si:**

**\b(si)\b**

****

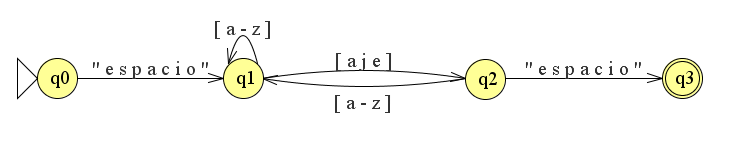
**-le:**

**\b(le)\b**

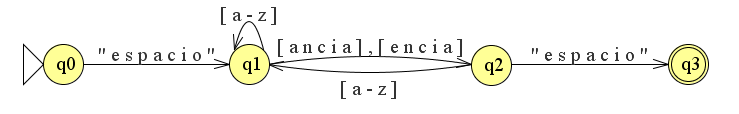
****

***Sufijos Sustantivos:***

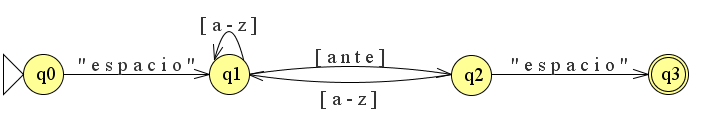
**-aje:**

**(a-z)\*(aje)\b**

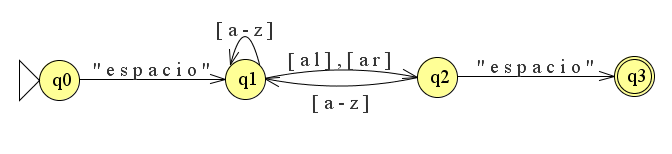
**-ancia,-encia:**

**(a-z)\*((e|a)ncia)\b**

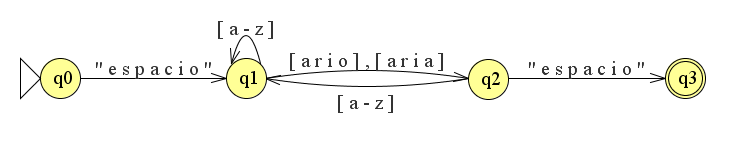
**-ante:**

**(a-z)\*((e|a|ie)nte)\b**

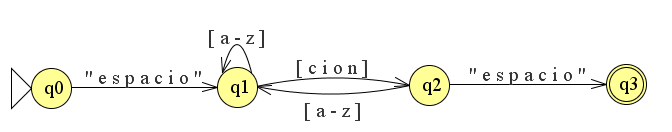
**-al, -ar:**

**(a-z)\*(a(l|r))\b**

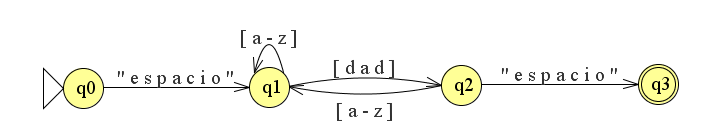
**-ario:**

**(a-z)\*(ari(a|e))\b**

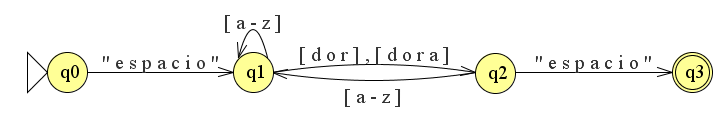
**-cion:**

**(a-z)\*(cion)\b**

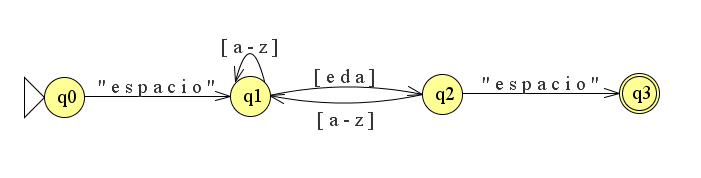
**-dad:**

**(a-z)\*(dad)\b**

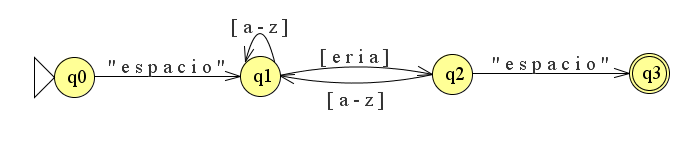
**-dor:**

**(a-z)\*(do(r|ra))\b**

**-eda:**

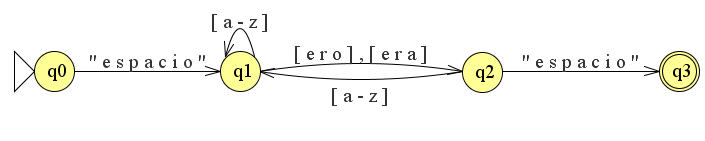
**(a-z)\*(eda)\b**

**-eria:**

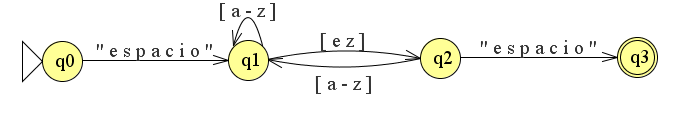
**(a-z)\*(eria)\b **

**-ero:**

**(a-z)\*(er(a|o))\b**

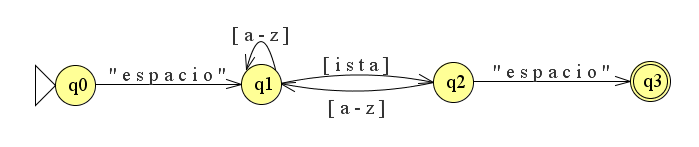
****

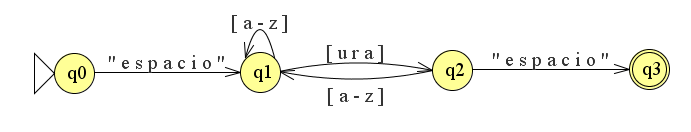
**-ez:**

**(a-z)\*(ez)\b**

**-ista:**

**(a-z)\*(ista)\b**

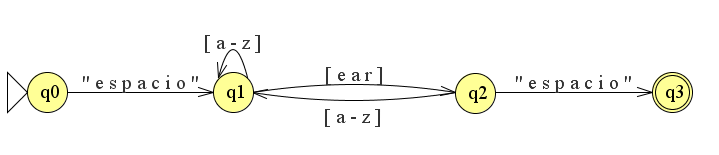
****

**-ura: (a-z)\*(ura)\b**

***Sufijos de Verbos:***

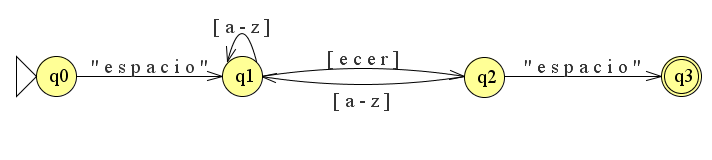
**-ear:**

**(a-z)\*(ear)\b**

****

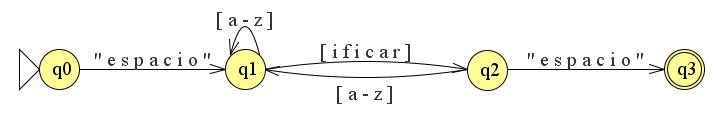
**-ecer:**

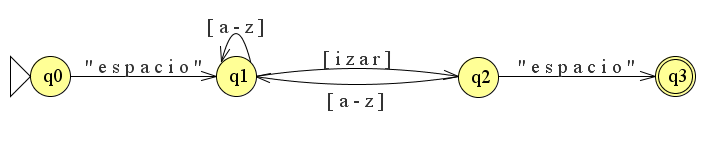
**(a-z)\*(ecer)\b**

****

**-ificar:**

**(a-z)\*(ificar)\b**

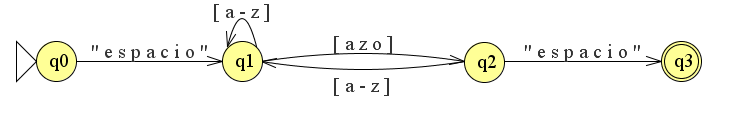
****

**-izar:(a-z)\*(izar)\b**

***Aumentativos:***

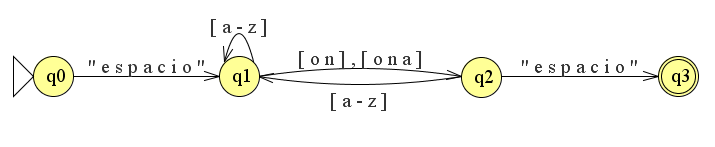
**-azo:**

**(a-z)\*(azo)\b**

****

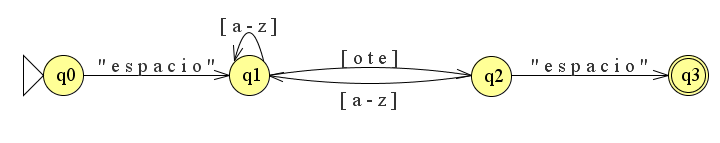
**-on:**

**(a-z)\*(o(n|na))\b**

****

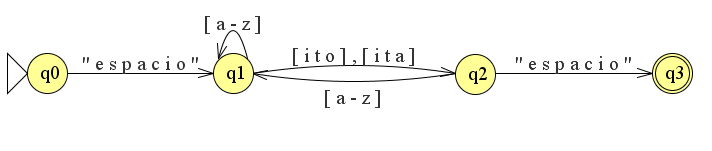
**-ote:**

**(a-z)\*(ote)\b**

****

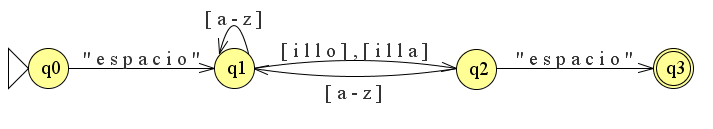
***Diminutivos:***

**-ito:**

**(a-z)\*(it(a|o))\b**

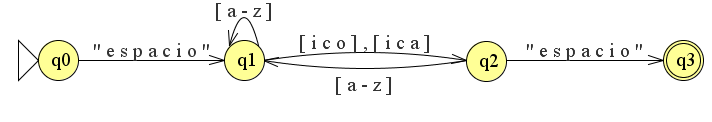
**-illo:**

**(a-z)\*(ill(a|o))\b**

****

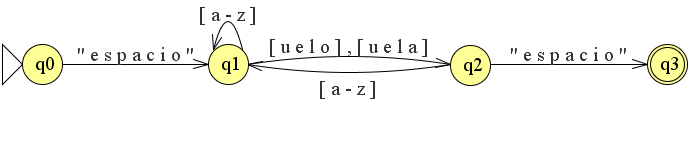
**-ico:**

**(a-z)\*(ic(a|o))\b**

****

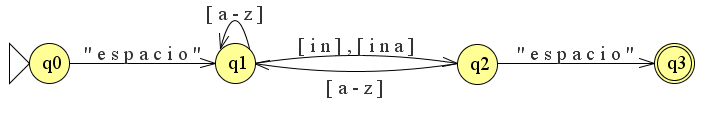
**-uelo:**

**(a-z)\*(uel(a|o))\b**

****

**-in:**

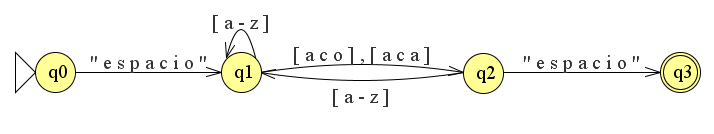
**(a-z)\*(i(n|na))\b**

****

**Despectivos:**

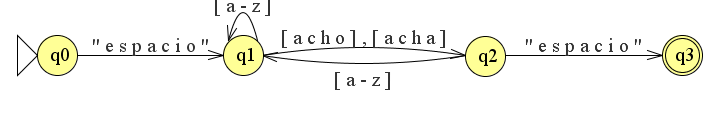
**-aco:**

**(a-z)\*(ac(o|a))\b**

****

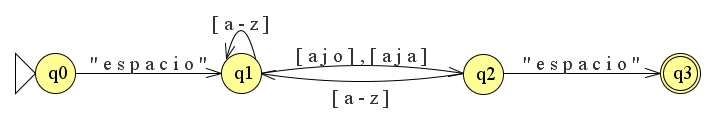
**-acho:**

**(a-z)\*(ach(o|a))\b**

****

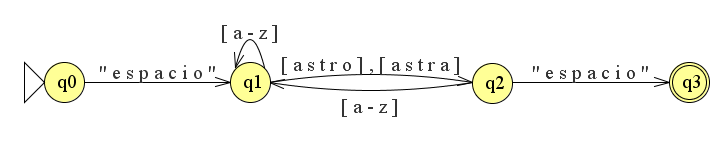
**-ajo:**

**(a-z)\*(aj(o|a))\b**

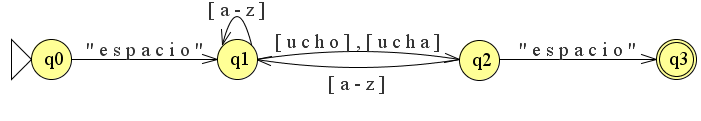
****

**-astro:**

**(a-z)\*(astr(o|a))\b**

****

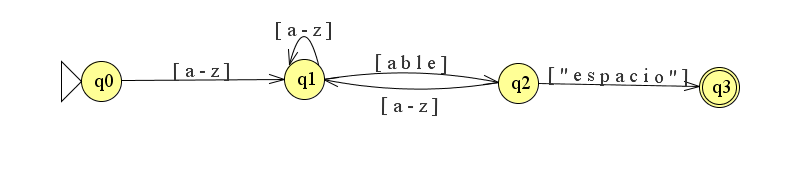
**-ucho:**

**(a-z)\*(uch(o|a))\b**

**Adjetivos:**

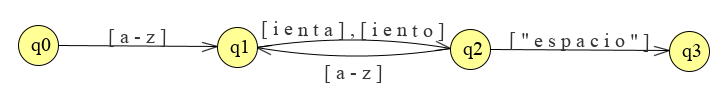
**-Able:**

**(a-z)\*(able)\b**

****

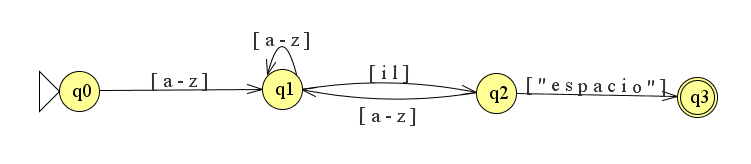
**-iento-ienta:**

**(a-z)\*(ient(o|a))\b**

****

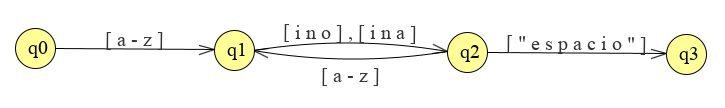
**-il:**

**(a-z)\*(il)\b**

****

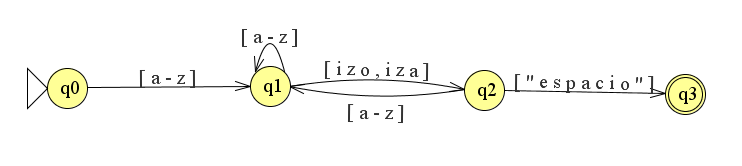
**-Ino-ina:**

**(a-z)\*(in(o|a))\b**

****

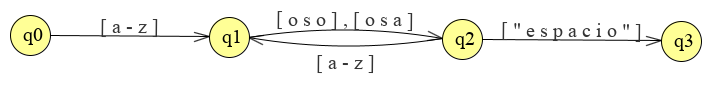
**-Iza-izo:**

**(a-z)\*(iz(o|a))\b**

****

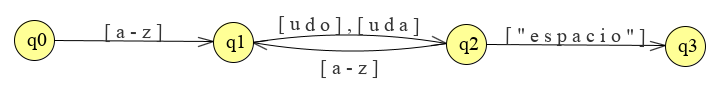
**-oso-osa:**

**(a-z)\*(astr(o|a))\b**

****

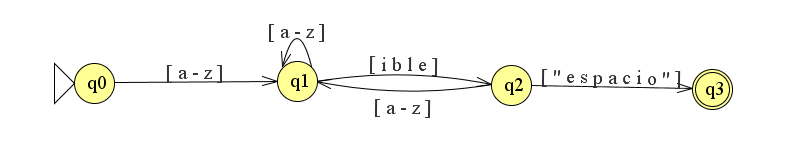
**-udo-uda:**

**(a-z)\*(ud(o|a))\b**

****

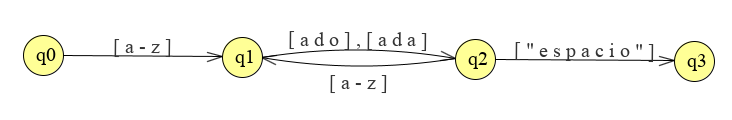
**-ible:**

**(a-z)\*(ible)\b**

****

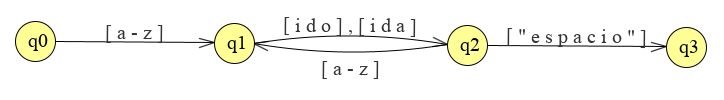
**-ado-ada:**

**(a-z)\*(ad(o|a))\b**

****

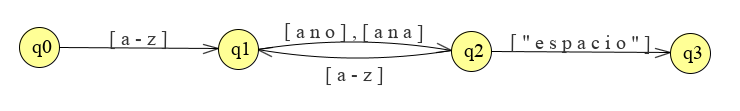
**-ido-ida:**

**(a-z)\*(id(o|a))\b**

****

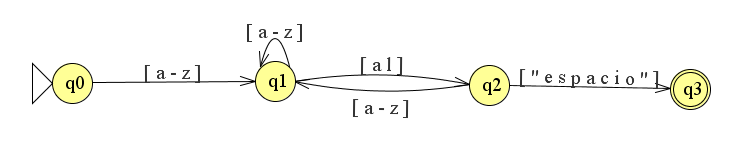
**-ano-ana:**

**(a-z)\*(ano(o|a))\b**

****

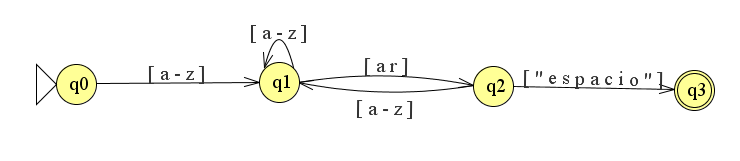
**-al:**

**(a-z)\*(al))\b**

****

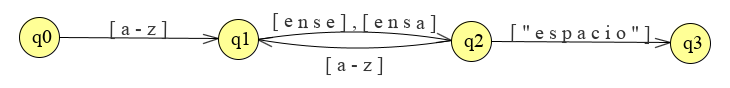
**-ar:**

**(a-z)\*(ar)\b**

****

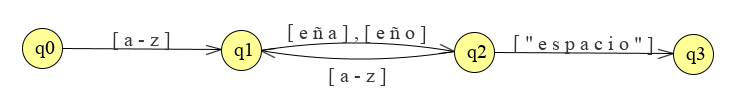
**-ense-ensa:**

**(a-z)\*(ens(e|a))\b**

****

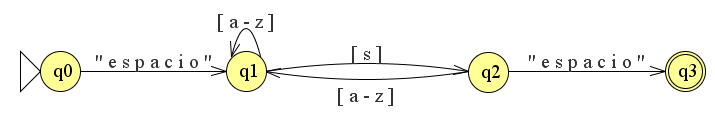
**-eña-eño:**

**(a-z)\*(eñ(o|a))\b**

****

**Eliminar s:**

**[s]\b**

****