TRANSPARENCIA: PROYECTOS DE LEY HACIENDO USO DE TÉCNICAS NLP

TRANSPARENCY: DRAFT LAWS USING NLP TECHNIQUES

Huillca Mozo, Bryan
Universidad Nacional de San Antonio Abad del Cusco,
Ing. Informática y de Sistemas
Cusco, Perú
160329@unsaac.edu.pe

Resumen— *Uno de los problemas que se sigue presentando hasta el dia de hoy es el de la falta de información acerca de los proyectos de ley que se presentan en el congreso, a causa de esto muchos peruanos no sabemos exactamente qué tipos de leyes son propuestas y cuales son aprobadas, además que cabe recalcar que existen también muchas personas que si se informan, pero al momento de leer la ley establecida, no entienden a que se refiere exactamente, puesto que dichas leyes en su mayoría son propuestas haciendo uso de un lenguaje técnico. En el presente proyecto de fin de semestre se propuso realizar una página web la cual contendría información clasificada de las leyes peruanas por los diversos grupos parlamentarios, además de ofrecer un buscador inteligente que filtre las leyes por busqueda ademas que este nos muestre alternativas de sinonimos de la palabra filtrada al igual que su propio significado, esto para que ayude al usuario a poder estar informado y a la vez seguro de si lo que se busca es lo correcto.*

palabras clave— extracción, leyes, datos, buscador

Abstract— *One of the problems that is still present today is the lack of information about the bills that are presented in Congress, because of this many Peruvians do not know exactly what types of laws are proposed and which are approved, and it should be noted that there are also many people who are informed, but when reading the established law, do not understand what exactly is meant, since these laws are mostly proposed using a technical language. In this end of semester project it was proposed to create a web page which would contain classified information of Peruvian laws by the various parliamentary groups, in addition to offering an intelligent search engine that filters the laws by search and also shows us alternatives of synonyms of the filtered word as well as its own meaning, this to help the user to be informed and at the same time sure if what is being searched is correct.*

Keywords—extraction, laws, data, search

I. INTRODUCCIÓN

En los últimos años se ha empezado a tocar mucho sobre el tema de las técnicas de NLP, y como este aporta en el mundo digital, además que cada día la necesidad del acceso a la información aumenta considerablemente.Todo aquello que necesitamos saber lo encontramos en internet como información como blogs, foros, noticias, etc. Esta sobreabundancia de información es uno de los principales componentes de su éxito, es por ello que es fundamental en la vida cotidiana de millones de personas que buscan diferente tipo de información y si es de una manera facil que mejor, pero en muchas ocasiones nosotros al realizar la búsqueda obviamos bastante información, quedándonos al final solo con una parte superficial.

Pero qué pasaría si aprovechamos el uso de las técnicas de procesamiento de lenguaje natural?, pues podríamos manejar a nuestro antojo bastante información, obtenerla y usarla para diferentes fines de propósitos..

II. OBJETIVO PRINCIPAL

El objetivo principal trazado es el de realizar un buscador y mostrar alternativas a la palabra ingresada haciendo uso de extracción de datos, esto en cada tópico de ley según la clasificación por grupos parlamentarios

III. MATERIALES Y MÉTODOS

Para llevar a cabo el levantamiento de dicho proyecto, se usó métodos de extracción de datos seguidamente estos datos se plasmaron en nuestra página web, además de usar también un kit de herramientas que se mencionan a continuación.

A. Kit de herramientas

- **Python.-** Es un lenguaje de programación multiparadigma, ya que este soporta de una manera parcial la programación orientada a objetos, programación imperativa y programación funcional en una menor medida. Haremos uso de este lenguaje de programación por la familiarización y su facilidad de uso.
- **Beautiful soup**.- Esta es una biblioteca del lenguaje python, la cual permite extraer contenido en formato HTML o XML. Se usará beautiful soup para extraer información de la página web Wordreference, la información extraída será la de

los sinónimos y significados de una palabra que recibira como parámetro en nuestro filtro de búsqueda.

- **WordReference.-** Página web referida a un diccionario multilingüe.
- **Pandas. -** Otra biblioteca de software usada para la manipulación de datos. Pandas nos servirá de gran ayuda para poder hacer uso de archivos csv.
- **Word cloud.-** se utiliza para representar datos de texto en los que el tamaño de cada palabra indica su frecuencia o importancia . Esto nos ayudará a resaltar que palabras extraídas resaltan más en nuestros archivos csv de acuerdo al tipo de leyes que se presentaron.

- **Flask.-** Es un mini framework que permite crear aplicaciones web rápidamente y con un mínimo número de líneas de código. Usamos flask por la facilidad de su uso, en sentido a que es casi un lenguaje puro en Python.
- **Mysql.-** Es un sistema de gestor de base de datos, el cual nos ayudó a poder almacenar los datos csv extraídos por nuestro corpus y así subirlas a nuestra página web.
- **Wordnet.-** Es una base de datos léxica del Idioma inglés que agrupa palabras en inglés en conjuntos de sinónimos llamados synsets.
- **Requests.-** Es una biblioteca HTTP cuyo objetivo es hacer que las solicitudes HTTP sean más simples y amigables para los humanos.
- **Selenium.-** Viene a ser un entorno de pruebas de software para aplicaciones basadas en la web

B.Solución Planteada

- En primera instancia se propuso realizar un corpus en el lenguaje de programación python, haciendo uso de las librerías de requests y selenium, esto con el fin de extraer datos de la página web (**https://www.congreso.gob.pe/**).

    1) Se usó la ruta del driver selenium “chromedriver” para poder interactuar con el navegador. Seguidamente le pasamos a este la ruta de la página a extraer datos como parámetro.
    2) Se estableció el número de páginas que se iban a visitar para la extracción de datos.
    3) Recorrimos las tablas que contenían leyes.
    4) Recuperamos los metadata, para ello nos fuimos a la parte inspeccionar la página y ver en qué etiqueta de HTML se encontraba el dato que queríamos recuperar, osea si se encontraba en una clase o cualquier otra etiqueta.
    5) Finalmente se extrajo los datos esperados con beautifulsoup, y estas las guardamos en archivos tipo csv para poder hacer la limpieza correspondiente y más adelante hacer uso de ella.
- Seguidamente se puso en marcha el desarrollo web el cual se vino realizando con el framework flask, y también haciendo uso del sistema de base de datos mysql.
    1) se creó una base de datos "BDLeyes"y una tabla TInformacion, la cual tenía como atributos ("Expediente","Periodo","Legislatura", "Fecha","Propone","Parlamento","Titulo","Objeto").
    2) Se hizo la limpieza de ruido al csv, y se subió los datos a nuestra base de datos.

- En cuanto a la página web desarrollada, esta tiene un contenido de los proyectos de ley clasificados por los

  diferentes grupos parlamentarios que existen.

- Cada página tiene un buscador inteligente, el cual pide como entrada un texto, sea cual sea su tamaño y si existe similitud con la ley a buscar, la muestra, además que también mostrará tanto los sinónimos y el significado de la palabra filtrada.

    1) En la estructura de la página para poder clasificarlos por grupos parlamentarios se les hizo sentencias select.
    2) Y para poder mostrar el significado y sinónimos de la palabra filtrada, se tuvo la idea similar a la del WORDNET .

- La idea para poder mostrar los sinónimos y surge también en la técnica del web scraping haciendo uso con las librerías anteriormente mencionadas, esta consistia en que como parámetros mandamos la url de la pagina wordReference y le añadimos un slash invertido seguidamente con la palabra sinónimo o definición, seguido de un slash, en este último entraría la palabra filtrada y nos redirigirá tanto a su respectivo sinónimo como definición. (ver figura 1)

```
import requests
from bs4 import BeautifulSoup
url='http://www.wordreference.com/sinonimos/'
ur2='http://www.wordreference.com/definicion/'
```

Figura 1: Referencia de URL para extraer información

Seguidamente le pasamos el valor de la palabra que se va a filtrar a buscar, luego hacemos uso de requests y beautiful para extraer los datos y esto finalmente guardarlos como tipo de texto.

El código se muestra a continuación

```
def sinonimos(enlace):

    url='http://www.wordreference.com/sinonimos/'
    buscar=url+enlace
    resp=requests.get(buscar)
    bs=BeautifulSoup(resp.text,'lxml')
    lista = bs.find(class_='trans clickable')
    sinonimos = lista.find("li")
    sino = sinonimos.text

    return sino
```

```
def sinonimos(enlace):

    url2='http://www.wordreference.com/definicion/'
    buscar=url2+enlace
    resp=requests.get(buscar)
    bs=BeautifulSoup(resp.text,'lxml')
    lista = bs.find(class_='trans clickable')
    sinonimos = lista.find("li")
    sino = sinonimos.text

    return sino
```

Donde el parámetro enlace será reemplazado por la palabra de búsqueda de filtrado de la página.

IV. RESULTADOS OBTENIDOS

Al realizar los procedimientos ya mencionados se obtuvieron los siguientes resultados:

1) La página web con los proyectos de ley clasificados por grupos parlamentarios.

2) El motor de búsqueda inteligente mostrando tanto las leyes que se buscan, como también alternativas de búsqueda (sinónimos) y su respectiva definición.

V. CONCLUSIONES

Se consiguió completar el objetivo trazado que era el de mostrar alternativas de búsqueda. Finalmente se concluye que el uso de las técnicas de extracción de datos es una técnica potente que puede ayudar a mejorar en la clasificación de la información de contexto, que apoyadas de otras técnicas de inteligencia artificial será la nueva alternativa a la producción lingüística.

VI. TRABAJOS PARA UN MEJOR COMPLEMENTO

Una idea a completar podría ser el de que cada vez que se haga una búsqueda y no se encuentre, entonces la búsqueda pase a la siguiente alternativa del sinónimo de cada palabra que nos muestre, Esto quiere decir que exista una iteración con cada sinónimo.

REFERENCIAS

[1] Murillo, Saavedra, Huriviades, " Implementación de algoritmo para la extracción de datos estructurados de perfiles en google académico"

[2] Pineda Leal V , "El uso de web scraping para la extraccion de datos". Universidad Nacional Autonoma de Mexico, 2017.

[3] wordReference, https://www.wordreference.com/