Memoria del Proyecto integrador

Hugo Ferrando Seage

Jorge de Castro Cabello

Santiago Gualda Torrijos

Cristian Lopez-Ramos Rivera





Índice

[Índice 1](#_Toc437603955)

[Introducción 2](#_Toc437603956)

[Contexto 2](#_Toc437603957)

[Funcionalidad 2](#_Toc437603958)

[Desarrollo 2](#_Toc437603959)

[Conclusion 3](#_Toc437603960)

# Introducción

El objetivo de esta Proyecto consistía en desarrollar un programa informático para la empresa Everis, cuyo objetivo principal es clasificar CV para una empresa.

# Contexto

En la actualidad, existen muchas empresas que reciben grandes cantidades de CV diariamente y no disponen de los suficientes medios para poder analizar y clasificar cada CV, por ello se ha decidido desarrollar un programa informático que presente las siguientes características:

* Flexibilidad
* Modularidad
* Integración sencilla
* Sencillez
* Escalabilidad

# Funcionalidad

El programa desarrollado no presenta todas las funcionalidades requeridas, ya que aún no se encuentra finalizado debido a que se había planteado para un periodo escolar, sin embargo cumple con estos objetivos:

* Incluir CV de cualquier tipo de formato
* Posibilidad de Clasificación
* Visualización con una interfaz básica

# Desarrollo

El desarrollo del programa se dividió en dos fases que transcurrieron durante el periodo de un mes y medio.

La primera fase consistió en el planteamiento e investigación inicial del programa, ya que se nos permitía usar cualquier tipo de entorno, se decidió usar finalmente Python debido a que el lenguaje presentaba unas mejores condiciones para el desarrollo del objetivo que otros tipos de entorno.

Resumiendo la primera fase, el programa se planteó para clasificar CV en distintos segmentos:

* Información Personal
* Información Académica
* Hobbies
* Idiomas
* Email

Para ello se implementó Tika, donde lo usamos para convertir CV de cualquier tipo de formato en un archivo de Texto y usando expresiones regulares conseguíamos clasificar la información.

Para la segunda parte dicidimos usar una biblioteca de Python llamada NLTK (Natural Language Toolkit). Entrenamos un POS (Piece of Speech) tagger y un chunker con el dataset “Conll2002” y usando Naive Bayes.

En primer lugar usamos un snowball stemmer para normalizar palabras. Luego tokenizamos frases con un tokenizador entrenado en espanol. Para cada seccion anadimos su tag con el POS Tagger (nombre, verbo, adjetivo, etc). Finalmente usamos el chunker para detectar personas, lugares y organizaciones.

Toda esta informacion la serializamos en formato JSON para su posterior uso en otros procesos.

# Conclusion

Creemos que nuestro proceso es un buen metodo para poder extraer informacion de CV. Para mejorar nuestros resultados deberiamos entrenar nuestro chunker con un dataset que se asemeje mas al formato usado en los CV.