UNIVERSIDAD
REY JUAN CARLOS

MÁSTER EN INGENIERÍA EN SISTEMAS DE DECISIÓN

Curso Académico 2018/2019

Trabajo Fin de Máster

Marco de trabajo para evaluar la relevancia de los artículos en el dominio científico

Autor : Adrián Alonso Barriuso

Tutor : Dr. Alberto Fernández Isabel

Dedicado a

mi familia, pareja, amigos y a todos los que me aguantan, en el buen sentido.

Agradecimientos

Resumen

Summary

Índice general

1. Introducción **13**

1.1. Contexto 13

1.1.1. Dominio de aplicación 13

1.2. Objetivos 13

1.2.1. Objetivo General 13

1.2.2. Objetivos específicos 13

1.3. Estructura de la memoria 13

2. Estado del arte **15**

2.1. Algoritmos de reputación 15

2.2. Obtención de relevancias 15

3. Propuesta **17**

3.1. Arquitectura general 17

3.2. Creación del lexicón de relevancias 18

3.3. Creación del la red neuronal 19

3.4. Estimación de relevancias de artículos 19

4. Experimentos y resultados **21**

4.1. Planificación temporal 21

4.2. Experimentos 21

5. Conclusiones **23**

Bibliografía **25**

Índice de figuras

3.1. Arquitectura general 18
3.2. Arquitectura de generación de lexicones 19
3.3. Flujo de trabajo 20

4.1. Diagrama de Gantt del desarrollo 21

Capítulo 1

Introducción

1.1. Contexto

1.1.1. Dominio de aplicación

1.2. Objetivos

1.2.1. Objetivo General

1.2.2. Objetivos específicos

-
-

1.3. Estructura de la memoria

1. Estado del arte:
2. Propuesta:
3. Experimentos y resultados:
4. Conclusiones:

Capítulo 2

Estado del arte

2.1. Algoritmos de reputación

2.2. Obtención de relevancias

Capítulo 3

Propuesta

En este capítulo, se describe la propuesta del sistema completo, definiendo entradas y salidas explicadas a nivel de diseño. Se empieza con una subsección donde se ve la arquitectura y el propósito general y después se entra en detalle para cada uno de los módulos en las subsiguientes secciones.

3.1. Arquitectura general

En la Figura 3.3, se pueden observar el módulo principal: *Relevance estimator module*, que es el encargado de estimar las relevancias de los artículos de entrada, y sus correspondientes submódulos: *Text processor, Reputation calculator y Relevance calculator*. Se cuenta además, con dos fuentes de información precalculadas utilizadas por el submódulo *Relevance calculator*, a saber, *Relevance lexicon* que consiste en cuatro lexicones[1] de relevancias de términos médicos y *Neural network*, que consiste en cuatro modelos entrenados de redes neuronales convolucionales (CNN)[2] por sus siglas en inglés. Por otra parte, se cuenta con un módulo de visualización(*Visualization*), que se utiliza para ver la salida del sistema y para proporcionar la entrada (*Text*). Por último, se utiliza una fuente de información externa en tiempo real (*Web information resources*).

En primer lugar se entra en detalle en cómo se construye Relevance lexicon, después Neural network y, por último, Relevance estimator module, el cuál emplea todo lo anterior para calcular las relevancias de nuevos documentos.

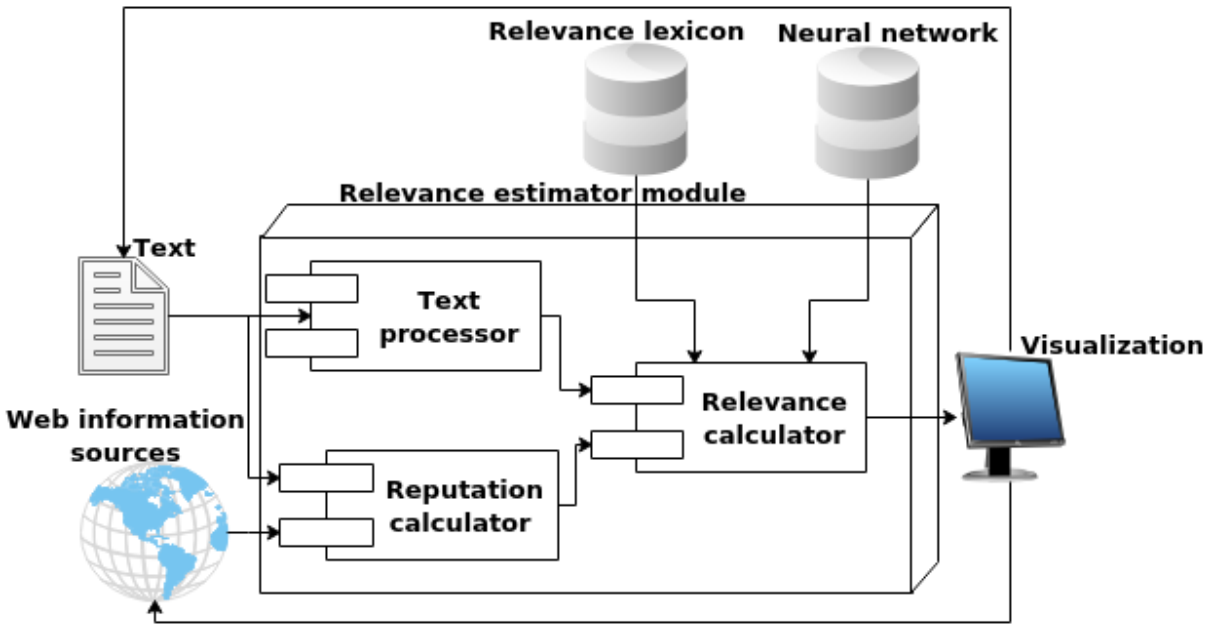

Figura 3.1: Arquitectura general

3.2. Creación del lexicón de relevancias

El lexicón de relevancias tiene un papel fundamental a la hora de estimar la relevancia de los artículos, en concreto, contiene la relevancia de las palabras de el dominio de aplicación utilizado para crear el mismo. Para la creación del lexicón se ha diseñado un marco de trabajo completo, cuya arquitectura puede verse en la Figura **??**. Esta cuenta con un módulo de extracción, transformación y carga (ETL por sus siglas en inglés)[**?**], un módulo de gestión de conomicimiento (Knowledge managment) y uno de visualización. Se cuenta, además, con dos bases de datos, una orientada a documentos[**?**](Document knowledge consolidation) y una ElasticSearch[**?**] para visualización con Kibana [**?**].

El módulo ETL se encarga de la obtención y el preprocesado del corpus de artículos médicos. Se divide a su vez en dos submódulos: Corpus processor y Reputation calculator. Corpus processor obtiene artículos de Pubmed Central[1] utilizando técnicas de web scrapping [**?**]. Estos artículos en formato XML [**?**], son parseados y almacenados en formato JSON [**?**] en Document knowledge consolidation con una estructura de párrafos y frases, eliminando tablas, gráficas u otros artefactos no aprovechables por el sistema. Una vez un documento se almacena en la base de datos, se procede a calcular su resumen automático aplicando el clásico algoritmo Text Rank

[1] `https://www.ncbi.nlm.nih.gov/pmc/`

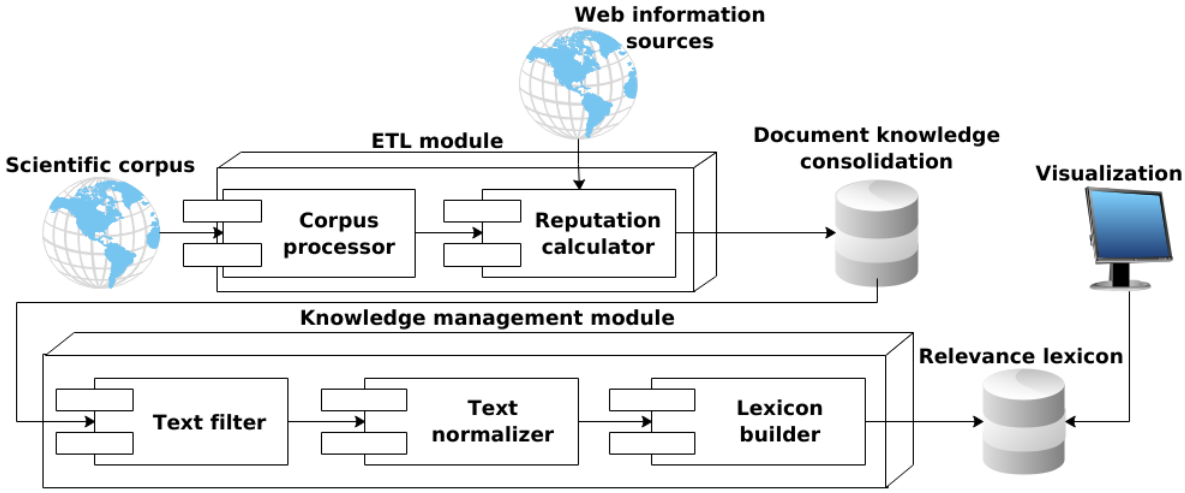

Figura 3.2: Arquitectura de generación de lexicones

[**?**] utilizando una popular implementación en Python [**?**]. Este tipo de resúmenes ofrecen una ordenación de las frases de un documento por relevancia, sirviendo como factor de filtrado de información poco relevante o redundante y a la vez como factor de compresión, haciendo la información más manejable en memoria. Se obtiene un 20 porciento del tamaño original de los artículos y se almacena como un nuevo artibuto del documento en la base de datos.

Una vez preprocesados y almacenados los artículos, Reputation calculator calcula la reputación de los mismos para que esta sea añadida como nuevo atibuto y ser utilizada posteriormente por el módulo Knowledge managment. El algoritmo de reputación empleado es una adaptación del empleado en el artículo [**?**]. La reputación a priori de un artículo viene dada por:

$$rep_p = \alpha \cdot rep\_authors_p + (1 - \alpha) \cdot citations_p\,, \tag{3.1}$$

3.3. Creación del la red neuronal

3.4. Estimación de relevancias de artículos

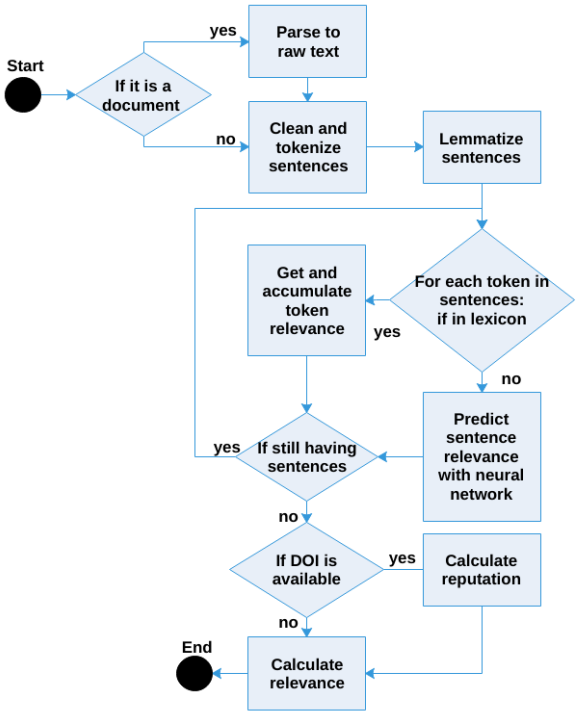

Figura 3.3: Flujo de trabajo

Capítulo 4

Experimentos y resultados

4.1. Planificación temporal

En la Figura 4.1 se puede ver un diagrama de Gantt[3] que refleja el tiempo empleado en cada una de las fases del proyecto.

4.2. Experimentos

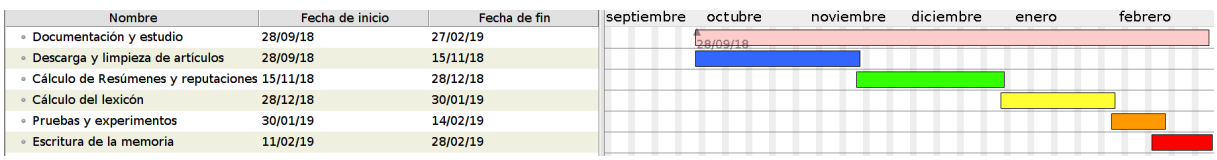

Figura 4.1: Diagrama de Gantt del desarrollo

Capítulo 5

Conclusiones

Bibliografía

[1] James Pustejovsky. The generative lexicon. *Computational linguistics*, 17(4):409–441, 1991.

[2] Soujanya Poria, Erik Cambria, and Alexander Gelbukh. Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2539–2544, 2015.

[3] James M Wilson. Gantt charts: A centenary appreciation. *European Journal of Operational Research*, 149(2):430–437, 2003.